

# AkAbak<sup>®</sup>

You have before you the fruits of years of development work. Expanding the initial idea into a finished product was a process involving extensive study, research work, and tests.

A large number of experts, colleagues and customers contributed invaluable advice and ideas. I would like to take this opportunity to heartily thank all those involved with the project.

## **Contact**

Joerg Panzer: [j.panzer@randteam.de](mailto:j.panzer@randteam.de)

# Table of Contents

<b>INTRODUCTION.....</b>	<b>8</b>
Simulation .....	9
Description of the structure to be simulated.....	9
Sound radiation.....	11
Mechano-acoustical coupling .....	12
Material-specific calculations .....	12
Non-linearities.....	12
System .....	12
Current network .....	12
Filter network.....	16
Script .....	17
Definitions .....	17
Off.....	18
OutPort .....	19
Comment .....	19
Labels.....	20
Parameter, Numbers and Units.....	20
Node numbers .....	21
Numerical value assignment.....	22
Assigning values from a formula system.....	24
Alphanumerical value assignments .....	24
Formula system.....	24
Names .....	25
Keyword.....	25
Formula Parser .....	25
Syntax and format.....	26
Radiation in General .....	30
Experiment.....	30
Observations .....	31
Point radiator.....	32
Diaphragm.....	32
Piston.....	32
Tympanum and plates .....	36
Cone.....	36
Dome.....	38
Baffle .....	38
Reflector .....	40
Radiation in AkAbak .....	41
Using the parameter.....	41
Diaphragm Parameter.....	42
Cross-sectional areas.....	42
Loudspeaker diaphragm.....	43
Positioning of Radiators .....	46
Parameter of the mounting position.....	47
Shifting of the origin.....	48
Radiation Environment .....	49
Parameter of diffraction.....	50
Parameter Reflection.....	52
Parameter NoRad and NoDir .....	52
Parameter Label= .....	53
Radiation Position - Dialog.....	54

Transducer .....	54
Dynamical driver .....	55
List of driver models of AkAbak .....	58
Parameter of transducers.....	59
Removing radiation load (Meas_...)	60
Sub-dialogs for fs, Qms, Qes and Mms.....	61
Modeling a transducer .....	61
Modeling enclosed acoustics .....	64
Example enclosure .....	65
Analyzing the cavity .....	66
Script.....	67
Simulation.....	68
Comparison with other enclosure types .....	71
Compact models.....	72
Helmholtz - Dialog.....	72
<b>FILE, EDIT, SEARCH.....</b>	<b>74</b>
File .....	74
Edit.....	75
Search.....	75
<b>IMPORT .....</b>	<b>76</b>
Data Format Requirements .....	76
Import Script .....	77
Import Script in AkAbak.....	77
Import Script in Tools .....	77
Def_Import .....	79
Parameter .....	79
Ordinate Formula System.....	80
Abscissa Formula System.....	82
File Interpretation Parameter .....	83
Import Display Dialog.....	86
Bode type, output curve .....	86
Diagram settings.....	88
Reload / from Import Script / from Dialog... ..	88
Copy graph.....	88
Close and diagram .....	88
Examples .....	89
<b>DEFINITIONS .....</b>	<b>93</b>
Def_Const .....	93
Def_ListeningPoint .....	94
Parameter .....	95
Def_Reflector .....	95
Parameter .....	97
Def_OpAmp.....	100
Parameter .....	101
Def_Transistor.....	101
Parameter .....	102
Def_Element (DLL) .....	102
Parameter .....	104
Def_Driver.....	104

Parameter .....	105
Def_TwoCoilsDriver .....	106
Parameter .....	107
Def_PiezoDriver .....	108
Parameter .....	109
Def_MeasRadiator .....	110
Parameter .....	111
Def_Speaker .....	111
Parameter .....	112
Def_BassUnit .....	113
Parameter .....	116
Def_BassUnit / Calculator .....	119
<b>GENERAL NETWORK COMPONENTS .....</b>	<b>125</b>
Coupler .....	125
Parameter .....	128
Gyrator .....	129
Parameter .....	130
Impedance .....	130
Parameter .....	131
Potential .....	131
Parameter .....	132
Element (DLL) .....	132
Parameter .....	133
Driver .....	133
Parameter .....	134
MeasRadiator .....	135
Parameter .....	136
Speaker .....	137
Parameter .....	137
Bassunit .....	138
Parameter .....	138
<b>ELECTRICAL NETWORK COMPONENTS .....</b>	<b>140</b>
Capacitor .....	140
Parameter .....	140
Coil .....	141
Parameter .....	141
Resistor .....	141
Parameter .....	142
Transformer .....	142
Parameter .....	143
OpAmp .....	143
Parameter .....	145
Transistor .....	146
Bipolar transistor .....	147
Parameter .....	148
Field-effect transistor .....	149
Valves .....	150

<b>MECHANICAL NETWORK COMPONENTS .....</b>	<b>152</b>
MechCompliance .....	152
Parameter .....	152
MechMass .....	153
Parameter .....	153
MechResistance .....	153
Parameter .....	154
<b>ACOUSTICAL NETWORK COMPONENTS.....</b>	<b>155</b>
AcouCompliance.....	155
Parameter .....	158
AcouMass.....	159
Parameter .....	161
AcouResistance.....	162
Some practical values of acoustic resistances .....	164
Parameter .....	165
Diaphragm.....	165
Parameter .....	166
Radiator .....	167
Parameter .....	169
Enclosure.....	170
Parameter .....	172
Enclosure losses .....	174
Vented cabinet .....	175
Note .....	177
Duct.....	178
Parameter .....	179
Horn.....	183
Horn function .....	184
Radiation cone.....	187
Horn designs .....	188
Parameter .....	191
Waveguide.....	193
Parameter .....	195
<b>FILTER .....</b>	<b>197</b>
Filter Element .....	197
Rational function .....	197
General formula system.....	198
Parameter .....	201
Synthesis of filter circuits.....	202
Filter Dialog.....	202
Group 'Transfer 1' .....	203
Group 'Transfer 2' .....	203
Sub-dialog 'Standard low-passes Functions' .....	203
Sub-dialog 'Bessel all-pass delay'.....	204
Filters for cross overs .....	204
Diagram .....	205
LCR-Synthesis.....	205
Source of the transfer function.....	206
Feasibility .....	206
Low pass, high pass and band pass .....	206

All pass .....	206
Synthesis.....	207
Attenuator .....	209
Compensation network for sealed enclosures .....	209
Inserting the network into the script .....	210
Active Filter Synthesis.....	210
Feedback .....	212
Generating the equations.....	213
Parameter .....	214
Polynomial → Product.....	214
Product → Polynomial.....	216
Polynomial → Poles and Zeros.....	217
<b>SIMULATIONS.....</b>	<b>218</b>
Common Controls of Simulation Control Dialogs .....	219
Distance of the listening point .....	220
Labels.....	221
Sum/ Acoustic Pressure.....	221
Sum/ Acoustical Power + Directivity Factor .....	222
Sum/ Beamwidth.....	223
Sum/ Directivity Pattern .....	223
Sum/ Sum Voltage.....	224
Sum/ Driving Point Parameter.....	225
Impedance .....	225
Power .....	225
Current.....	225
Sum/ Extreme Values .....	225
Maximum values of .....	226
Sum/ Power Density.....	226
Inspect/ Voltage.....	227
Inspect/ Current .....	227
Inspect/ Electrical Power .....	228
Inspect/ Force.....	228
Inspect/ Excursion .....	228
Inspect/ Velocity .....	228
Inspect/ Acceleration.....	229
Inspect/ Pressure.....	229
Inspect/ Volume Velocity .....	229
Inspect/ Network Impedance .....	230
<b>TOOLS.....</b>	<b>231</b>
Create Def_MeasRadiator File... ..	231
Procedure .....	231
Measured data.....	232
Dyn. Driver Parameter.....	234
Procedure .....	234
Measured data.....	235

Piezo Driver Parameter .....	235
Procedure .....	235
Measured data.....	236
Mms/ Cms/ Vas Calculation.....	237
Application of an added mass .....	237
Installation in a sealed enclosure.....	238
Sound-level measurement (Reference Lp).....	239
Impedance Compensation.....	240
Procedure .....	241
Tools/ Impedance Compensation.....	241
<b>APPENDIX .....</b>	<b>243</b>
Installation.....	243
Technical Program Information.....	245
Literature References.....	246
Software License Agreement.....	247

# Introduction

AkAbak<sup>®</sup> is a synonym for 'Acoustic Abakus'. This software provides an integrated development system for analyzing and designing electroacoustic devices such as loudspeaker drivers and systems.

To be able to describe the structure under investigation by means of AkAbak, the structure must be first separated into a set of lumped elements and one-dimensional waveguides. The components are wired with a node-based system, called a network. Based on linear system theory, the analysis is carried out in the frequency domain by solving the node-potential-matrix.

The method to simulate electroacoustical devices by means of lumped elements can be regarded as classical. What is a 'lumped element' and what is the associated simulation method? Without going into details here and roughly speaking, a lumped element represents any device, with respect to its response, when the dimensions are small compared to the wavelength.

Well known examples are the electrical resistor, the capacitor etc. when used in the low frequency range. Here the dimension of the resistor component is much smaller than the electromagnetic wavelength.

The same applies to electroacoustic devices, but that the rate of the velocity of sound is much more smaller. When we analyze audio systems then the medium is air and in many cases the wavelength is at low frequencies much larger than the electroacoustic components like enclosures, diaphragms, obstacles etc. Thus we can use lumped elements representing the effects. At the upper edge of the audio frequency band the ratio of wavelength to component-dimensions can become so small that usually only the three dimensional wave-equation can describe the response sufficiently. In the intermediate frequency range so-called one-dimensional waveguides can be used. This special waveguides lump the acoustical effects of two dimensions and describe the wave-propagation only in one dimension.

A simulation method based only on lumped elements and one-dimensional waveguides implies several points:

1. The modeling bases on understanding of the physics of each part of the system under investigation and can not be automated.
2. The lumped element method remains always a bit abstract in contrast to the real acoustic structure which has precise geometric dimensions.
3. The simulation result can give us only an approximation. It displays only the fundamental effects.

The advantage of the lumped element method is the flexibility and the low number of parameters necessary to describe the system. When applied properly, it clearly displays the fundamental effects and usually leads to a better understanding of the physics of the device under investigation.

AkAbak offers special components and tools which are dedicated to especially loudspeaker design. Beside fundamental electric, mechanic and acoustic elements there are several complete models for transducers, waveguides, enclosures and radiation which can be used to model loudspeaker systems. For the cross over design a large palette of filter elements and related analysis and synthesis tools are available.

Before arranging all of the components into a full system it is necessary to understand how AkAbak handles, for example, the radiation of cone shaped diaphragms, filter, networks, and other computational aspects of the simulator.

The 'Exercises' contain some practically-oriented problems, set out in clear and simple stages. The associated script files are located in the \AkAbak\Scripts\Examples' directory. The detailed tutorial exercise No.1 will be especially useful to beginners (ex\_11.aks, ex\_12.aks,...). This exercise has been found to provide a very quick introduction to the underlying philosophy and use of AkAbak. The individual chapters also give several examples at the appropriate places.

There are two documentation's available: 1. The manual and 2. the AkAbak help system. The content of both is similar but the manual tends to be more informal whereas the help system concentrates more on operating of the program. Therefore press **F1** whenever you need help especially to file, dialog and diagram operating.

People work with simulation programs for various reasons. Some use them for teaching or for studying electroacoustics. Technicians and engineers use them to help them develop ideas. AkAbak is very flexible and



allows you to simulate any possible structures, and even some that are impossible. Try experimenting with it. All the models it uses and their parameter are based on physical principles.

Get to know the elements and how they work. Working with AkAbak will stimulate your imagination. The program will prove a source of any amount of new ideas. At the same time it supports a conceptual structure that will help you to put your ideas into practice more easily.

If you haven't worked with a simulation program before, you will find that AkAbak becomes a new component in your development toolbox. Acoustic conditions are made crystal clear and controllable. Whereas putting an idea into practice was previously a matter of technical feasibility, now it is easier to integrate acoustic laws into the development process, simply because now they can be observed.

Colleagues in high-frequency engineering have already introduced this integration into their development routine. They use a similar computer program for simulation in the HF range. 'At first we found our simulation confusing - nothing was right. After we had learned to integrate our design into the simulation, though, we had a good measure of control over the development. Today we work much more efficiently and know a lot more about the relationships in our circuits.'

These words may help encourage you when you start developing with AkAbak. It really isn't easy at the beginning, but it is worthwhile getting to know the material and elements of the program.

We hope your work with AkAbak is pleasurable and profitable and - in the spirit of Novalis, who said 'a true reader is a continuation of the author' - we look forward to a fertile exchange of ideas.

Jörg W. Panzer

## Simulation

Simulation means reproducing the characteristics of a system. In this case, the system is physical in nature. The characteristics relate to the dynamic transmission behavior from an input to the system to an observation point. The output variable of the transmission is always the impulse response to the input voltage of the entire system. In most cases, the so-called Bode diagram in the frequency range is plotted. The Bode diagram displays the amplitude response, phase response, etc. of the system in the steady state.

The electroacoustic structure to be simulated is reproduced by a mathematical model that takes into account linear system theory. In the system being simulated, only the components relevant to the given observation space are reproduced.

## Description of the structure to be simulated

The structure to be simulated has to meet certain prerequisites if AkAbak is to be able to process it. AkAbak itself has a set of mathematical models that follow the laws of linear system theory. The complexity of the individual models and the display of their transmission are principally oriented to the practical design of loudspeaker systems. In this case simulation has to be accurate enough to represent the main features of the analysis. On the other hand, it is desirable if the computation time is short and the number of parameter is small.

## Structuring

The system to be simulated has to be structured so that the program's models can be used.

To accomplish this, it is necessary to know something about the electrical, mechanical and acoustical components of AkAbak and how they are networked. Then the system is analyzed into its components and an attempt is made to describe them using the parameter of the AkAbak elements. In the next stage, the structure of the system is described using the nodes and branches of the AkAbak networks.

Translating the actual system into the AkAbak structure and estimating which of the parameter that can be entered are important for the analysis are by far the most difficult tasks, and often the biggest source of errors. There is no simple recipe for this. The most successful method is to plan the description carefully and to check it by actual measurements in order to gain experience.

The electrical components are connected as in conventional circuitry. It is less usual to use and connect driver, mechanical and acoustical elements. In the mechanical domain the force is the potential and the velocity is the flow. At the acoustic side the sound pressure corresponds to the potential and the volume velocity corresponds to the flow.

Mechanical, acoustical elements are in parallel in the network if the same force, pressure acts at their connection poles. If, for example, the two drivers radiate into an enclosure, the connection poles at the diaphragm reverse-side are connected in parallel with the driver.

Acoustical elements are in series if the gas particles in them have the same velocity, weighted by a surface area. If, for example, the two aforementioned drivers first radiate into a tubular enclosure that is terminated by a second enclosure of greater volume, the drivers, acoustic ducts and the enclosure are in series.

## Discretization

An important aspect in describing the structure to be simulated is the ratio of the dimensions to the wavelength in the frequency band under consideration. Whereas electrical components are usually considerably smaller than the wavelength of the frequency to be transmitted, this is not the case with acoustic components. In the audio range, the wavelength and the dimensions of the structure are of a similar order of magnitude. The transmission behavior is not only time dependent, but is also space-dependent. Since the networks in AkAbak only accept discrete elements, it is necessary to structure the system so as to produce elements with space-dependent transmission characteristics that are independent of one another.

The acoustic `Duct` element, for example, has a clearly defined form. Although this form can have any uniform cross-section, the length of the duct is the same across the entire cross-section and begins at the same point in the room. If the structure to be simulated does not correspond to this ideal form - for example the duct enlarges in the manner of a funnel at one end - this structure should be subdivided so that the ideal shape of the duct can be recognized and described. The remainder is then described using other AkAbak modules. In this example, the duct is networked with a horn element or waveguide element. Although both elements (`Duct` and `Horn`) have themselves room-dependent transmission characteristics, they act outwardly as discrete elements. When the output of the duct is networked with the input of the horn, the wave propagation automatically continues in the same direction.

## Observation space

It is often not very easy to decide to what extent structures of the system to be simulated have to be taken into account. Three questions may help:

What is the ratio of the dimensions to the wavelength of sound in the frequency band under observation?

- 1) Is the acoustic structure actually in the observation space?
- 2) How sensitive is the overall structure to the parameter of this substructure?

### **regarding 1) What is the ratio of the dimensions to the wavelength of sound in the frequency band under observation?**

As a general rule: if the wavelength is much larger than the dimension under consideration, then the transmission of the element is space-independent.

#### ☒ Some examples

A vent connecting two cavities can be reproduced by an acoustic mass at low frequencies. Above a frequency with a wavelength approximately eight times longer than the vent length, the complete model of the acoustic duct should be used

In radiation from surfaces, a diaphragm acts like a point radiator at low frequencies. From the so-called directivity frequency  $f_D$  upwards (wavelength=diaphragm circumference) the sound energy radiated laterally is, with increasing frequency, more and more attenuated by interference.

In sound diffraction, at low frequencies the sound is diffracted around the enclosure containing the radiator. From the so-called diffraction frequency, the back space is more and more decoupled, so that baffle conditions predominate at high frequencies.

## regarding 2) Is the acoustic structure in the observation space?

The observation space of a transmission path contains only the components that contribute to the system response at the observation point.

### ☒ An example

Assume that two networks are connected to an ideal voltage source. The voltage transmission is measured at any branch in network A. Since the voltage source is ideal, a variation of any kind in network B will have no effect on the potential distribution in network A, and therefore will also not be observable. Network B therefore lies outside the observation space.

This example is, of course, trivial, but illustrates the problem. In acoustics, the exact description of even extremely simple structures, such as the potential distribution in a rectangular, sealed enclosure, rapidly leads to extremely complicated formulae. We simplify these by first attempting an independent description in the three directions in space. Then we investigate how these contribute to the transmission. For example, standing waves in a rectangular enclosure exist in all three directions in space. The radiation impedance exhibited by the enclosure to a built-in diaphragm is described by the characteristic functions in all three directions in space. The greatest effect, however, is exerted by the characteristic functions in the direction of the baffle surface.

The standard element 'enclosure' only takes into account the one-dimensional characteristic function, i.e. the standing waves between the diaphragm and the enclosure wall opposite. Within the given scope of accuracy of AkAbak, therefore, the standing waves between the other walls are located outside the observation space.

## regarding 3) How sensitive is the overall structure to the parameter of this substructure?

This question is closely related to the last point. The parameter of an element do not all act with equal intensity on the observation parameter in the overall system. Taking the example of the enclosure given in point 2, it is clear that the volume of the enclosure influences the observation parameter 'sound pressure outside the enclosure'. The standing waves have only a small influence on this parameter.

If in doubt, change the parameter under consideration and compare the resulting curves, or install additional elements at dubious node points to test how dependent the observation parameter is on them.

## Sound radiation

AkAbak not only simulates network parameter, but also the transmission of the sound to a listening point in the room. It determines the characteristics of the so-called radiating elements and summates the sound pressure according to modulus and phase depending on the position with respect to the listening point. AkAbak simulates the transmission of the sound source. It does not take into account the acoustics of the room. However, reflectors located close to the radiators and the sound diffraction form part of the sound source, and their effects can also be taken into account.

Radiation and sound diffraction are extremely complicated subjects, and are difficult to calculate. A great deal of research work and practical tests have led to a mathematical approach that, while taking into account all the essential points of the radiation and sound diffraction, also provides acceptable computation time.

The radiation-problem of conical and dome type diaphragms is solved by numerical integration over the surface of the diaphragm. The diaphragm form has effects not only on the free-field radiation but also on the reaction of the pressure on the diaphragm area (radiation impedance). The diffraction and reflection of nearby edges and walls are taken into account.

A vibrating diaphragm possesses inertia and a finite inner stiffness. Because this part of the diaphragm is decoupled from controlled movement at higher frequencies. AkAbak cannot calculate the various eigenfrequencies and their effects but takes into account the diminishing diaphragm mass and area with increasing frequency.

AkAbak simulates the loading and the on-axis radiation of even complex horns very well. To have an estimation of the directivity of a horn AkAbak implements a radiation sphere inside the horn with finite rectangular radiation elements. The curvature of this sphere depends on the geometry of the horn.

## Reflection and sound diffraction

The effect on the radiation and on the impedance of maximum three reflecting walls close to the sound source are taken into account.

The diffraction problem of the finite baffle is solved by mirror radiators. This effects astonishing good reproduction. The diffraction at the cabinet edges causes a more or less intense ripple in the frequency range for which the wavelength is comparable to the dimensions of the baffle.

## Mechano-acoustical coupling

A vibrating structure radiates sound into the surrounding medium. The coupling device is called diaphragm. In most cases which are going to be simulated the diaphragm is included in the electro-acoustical transducer. Also the mechanical elements are comprised by the transducer.

But there may be cases where a closer look to the mechano-acoustical structure is necessary. Examples are compression-drivers, microphones etc. In these cases you will use the element Coupler - an ideal transformer. This element couples the acoustical network to the mechanical and - in some cases - also the mechanical to the electrical (piezo or electro-static transducers). With the help of the Coupler the diaphragm can be subdivided in different areas each radiating in the acoustical structure.

## Material-specific calculations

AkAbak does not carry out material-specific calculations, such as those for partial vibrations of the diaphragm, natural vibrations of the enclosure walls, changes of resistance due to heating, etc.

Frequency-independent losses in the acoustic structure are taken into account by means of quality parameter. Frequency dependent losses can be modeled by a so-called runtime-formula (see element Impedance).

As indicated above under 'Sound radiation', AkAbak takes into account the diminishing of the diaphragm area at higher frequencies due to mass inertia.

On this effect are superimposed partial vibrations. They occur in the same frequency range. Partial vibrations are natural vibrations in the diaphragm that propagate as transversal waves with variable intensity and mode, depending on the velocity of sound, damping and shape of the diaphragm. The partial vibrations cause sound to be radiated very diffusely and at very selective frequencies.

## Non-linearities

AkAbak can investigate non-linear characteristics of the components, such as change of resistance due to heating, or the non-linear compliance of springs, by introducing operating points. The formula describing the parameter at the operating point may be entered directly as such.

## System

A `System` comprises a current network and a filter network.

First of all, it must be explained what AkAbak understands by the terms system, network and independent networks.

## Current network

Unless otherwise stated, network is to be understood to mean the current or flow network in this documentation (Fig. 1). There is also the so-called filter network. In the current network, lumped electrical, mechanical and acoustical components as well as electroacoustic drivers and acoustic structures are connected together. Each of these elements connects at least two network nodes to the so-called flow parameter.

## Flow parameter and potential

In the electrical part of the network, the flow parameter is equal to the electrical current  $I$ . In the mechanical part the flow is the velocity  $v$  and in the acoustic part it is equal to the volume velocity of the sound wave  $V$ .

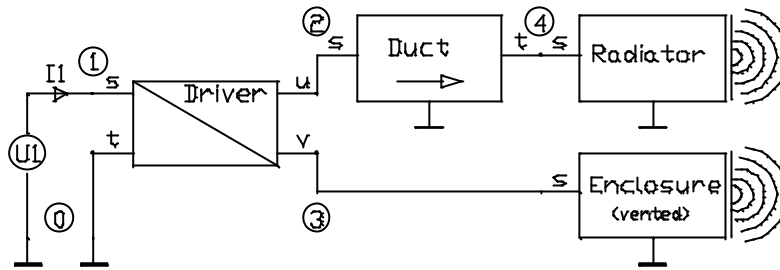


Fig. 1 Example of an acoustic current network

The volume velocity is the velocity  $v$  of the gas particles in a particular area  $S$ :  $V = v \cdot S$ . This unit works out to  $\text{m}^3/\text{s}$ , i.e. a volume per second, from which it derives its name.

The so-called potential is from the network nodes to ground. In the electrical part, the potential is equal to the electrical voltage  $U$ , in the mechanical part the potential is the force  $F$  and, in the acoustic part, equal to the sound pressure  $p$ .

## Topology, nodes

There are various ways of describing the topology of the network. In the method used here, the network is described by the node numbers. The node is a branch of current, at which two or more components meet. There are no special electrical, mechanical or acoustical nodes. The numbers can be in any sequence. The maximum value for a node number is 32000. The maximum number of nodes is 56. To move node numbers, the program makes this easier by providing a tool in the 'Search/Move Nodes' menu.

### Ground nodes

The node with the number 0 is reserved for the ground node. Those element poles that are connected to ground are given the node number 0.

### Input node

Node number 1 is reserved for the input voltage  $U_1$  of the network.  $U_1$  is connected to node and to ground.  $I_1$  is the current flowing into the network at node 1 and away to ground.  $Z_1$  is the impedance that the network presented by the network at nodes 1, 0. These network input parameter can be observed in the simulation (Inspect/... menu).

### Network input voltage

The level of the input voltage of the network  $U_1$  is determined by the output voltage of the filter network of the system. If the system has no *Filter* elements,  $U_1$  is equal to the input voltage  $U_{in}$ , whose rms or peak value is entered into the control dialogs of the simulation.

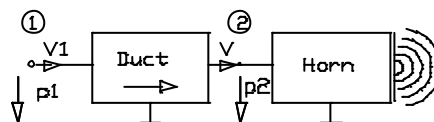


Fig. 2 A purely acoustical network

## Purely mechanical or acoustical networks

If the network consists of purely mechanical or acoustic elements, the source potential  $U_1$  is equal to the force  $F_1$  or to the sound pressure  $p_1$ , respectively. The identifiers in the program cannot be changed, but the

electrical parameter can be inserted in place of the mechanic or acoustic parameter, as Fig. 2 shows,  $p_1$  corresponds to  $U_1$  and  $V_1$  corresponds to  $I_1$ :

voltage	$\leftrightarrow$	force	$\leftrightarrow$	pressure:	1 V	$\equiv$	1 N	$\equiv$	1 Pa
current	$\leftrightarrow$	velocity	$\leftrightarrow$	volume velocity:	1 A	$\equiv$	1 m/s	$\equiv$	1 m <sup>3</sup> /s
impedance	$\leftrightarrow$	mechanical impedance	$\leftrightarrow$	acoustic impedance:	1 $\Omega$	$\equiv$	1 Ns/m	$\equiv$	1 Ns/m <sup>5</sup>

## Independence of current networks

A current network is independent of another current network if no currents or volume velocities can flow from one network into the other. If, therefore, the current circuit to be simulated contains network parts that meet this criterion, the circuit can be divided into two or more systems.

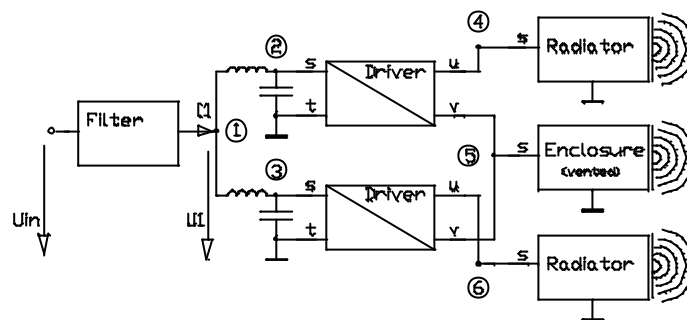


Fig. 3 The upper and lower network parts are connected via node 5

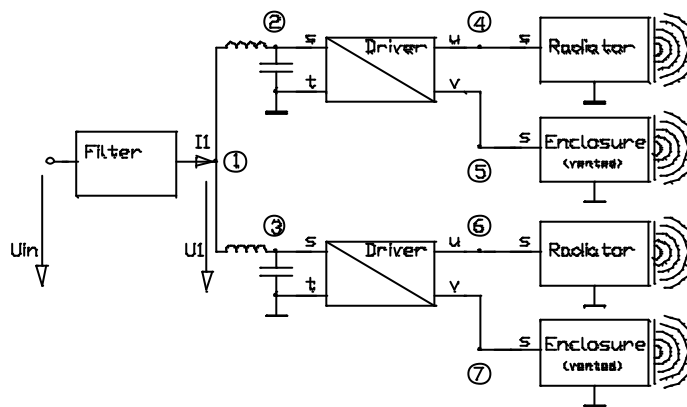


Fig. 4 The upper and lower network parts are independent.

Fig. 3 and Fig. 4 show typical circuits. The input voltage  $U_{in}$  is at a **Filter** element and is weighted with its transfer function. The current network begins at node 1. Two electroacoustic transducers (**Driver**) load, by means of their input impedance across nodes 2,0 and 3,0, a low pass consisting of the coils and the capacitor. At the front part of the driver-diaphragm the radiation element **Radiator** is attached (nodes 4 and 6).

In Fig. 4 the backs of the diaphragms of both drivers are loaded by the impedance of a vented cabinet (**Enclosure**). The upper and lower network parts are connected here. Since both loudspeakers radiate into the same enclosure, they exert an influence on one another. For the circuit from Fig. 3, the script is as follows:

```

Def_Driver 'B1'
  dD=17cm dDl=5cm tDl=5cm fp=1.3kHz |Cone
  fs=30Hz Vas=135.3L
  Qms=1.92 Qes=0.2 Re=5ohm Le=1.1mH ExpoLe=0.6

System 'Fig1'
  Coil Node=1=2 L=1.125mH
  Capacitor Node=2=0 C=22.5uF
  Driver Def='B1' Node=2=0=4=5
  Radiator Def='B1' Node=4
    x=0 y=-10cm z=0 HAngle=0 VAngle=0
  Coil Node=1=3 L=1.125mH
  Capacitor Node=3=0 C=22.5uF
  Driver Def='B1' Node=3=0=6=5
  Radiator Def='B1' Node=6
    x=0 y=10cm z=0 HAngle=0 VAngle=0
  Enclosure Node=5
    Vb=50L Qb/fo=0.1 fb=30Hz Lb=26cm dD=10cm
    x=0 y=-20cm z=-30cm HAngle=180° VAngle=0

Filter fo=30Hz {b2=1; a2=1; a1=sqrt(2); a0=1; }

```

In Fig. 3, each loudspeaker is mounted in its own enclosure. The loudspeakers are acoustically decoupled. The upper and lower parts of the network are independent of one another.

The circuit shown in Fig. 4 is not derived from that in Fig. 3. These are two independent examples.

In fact, there is a further interconnection, namely that via the free radiation space of several radiators. One loudspeaker is always a microphone as well, so that the sound pressure produced by one radiator induces a voltage at the terminals of the other radiator. AkAbak does not take into account this feedback, because its effect is very small. On one hand the sound pressure decreases with distance and the active area of the receiver (diaphragm) is considerably smaller than the area of the sphere at the receiving point. On the other hand, the efficiency of the driver is small, so that only a fraction of the received sound energy reaches the driver terminals. The program neglects this effect, since it is beyond the limits of accuracy of the models of the other models used in the simulation.

We can thus divide the circuit of Fig. 4 into two systems. The `Filter` element is then either found separately in each system, or the `Filter` system is entered in the definition part of the script. In the latter case, its transfer function is weighted by the input voltage  $U_{in}$  of all systems.

The current cannot flow via node 1 from one network into the other, since the filter represents an ideal voltage source with zero ohm generator resistance.

A practical example can be found in conventional passive cross overs, which are fed by a power amplifier. Further active filter circuits may be connected before the power amplifier. If there is a resistance between the power amplifier and the input of the cross over, for example as a result of cable losses or the like, the two parts of the circuit are connected again and can be described in just one system.

If the circuit of Fig. 4 is divided into two systems, the following script results:

```

Def_Driver 'B1'
  dD=17cm dDl=5cm tDl=5cm fp=1.3kHz |Cone
  fs=30Hz Vas=135.3L
  Qms=1.92 Qes=0.2 Re=5ohm Le=1.1mH ExpoLe=0.6

System 'S1'
  Coil Node=1=2 L=1.125mH
  Capacitor Node=2=0 C=22.5uF
  Driver Def='B1' Node=2=0=3=4
  Radiator Def='B1' Node=3
    x=0 y=-10cm z=0 HAngle=0 VAngle=0
  Enclosure Node=4
    Vb=25L Qb/fo=0.1 fb=30Hz Lb=26cm dD=10cm

```

```

x=-10cm y=-20cm z=-30cm HAngle=180° VAngle=0

Filter fo=30Hz {b2=1; a2=1; a1=sqrt(2); a0=1; }

System 'S2'
Coil Node=1=2 L=1.125mH
Capacitor Node=2=0 C=22.5uF
Driver Def='B1' Node=2=0=3=4
Radiator Def='B1' Node=3
x=0 y=10cm z=0 HAngle=0 VAngle=0
Enclosure Node=4
Vb=25L Qb/fo=0.1 fb=30Hz Lb=26cm dD=10cm
x=10cm y=-20cm z=-30cm HAngle=180° VAngle=0

Filter fo=30Hz {b2=1; a2=1; a1=sqrt(2); a0=1; }

```

In addition to the current network, a system also contains the `Filter` elements connected before it. All the `Filter` elements following the `System` keyword form part of this system. The next `System` starts with a new `System` keyword. The `System` keyword may follow a name. The name then appears in the legends of the diagrams and in some selection lists. It is used to make the script easier to read and improve the documentation. The name is given in quotation marks ('...' or '...'), can contain any characters and has a maximum length of 20 characters. For example:

```

...
System 'tweeter'
Speaker Def='K1' Node=1=0
| Linkwitz-Riley HP-4th order
Filter fo=3kHz {b2=1; a2=1; a1=sqrt(2); a0=1; }
Filter fo=3kHz {b2=1; a2=1; a1=sqrt(2); a0=1; }

System 'woofer'
...

```

In this extract from a script, the current network consists of only one `Speaker` element. The two 2nd order Butterworth high pass filters are multiplied together and thus result in a 4th order Linkwitz-Riley filter. The voltage U1 across the nodes 1,0 is applied to the terminals of the `Speaker` element and follows the transfer function of the filter. The next `System`, 'woofer', is now completely independent of the 'tweeter' `System`.

## Filter network

Using the filter network, the abstract, feedback-free `Filter` elements within a system can be networked with one another and with the current network. The topology of the filter network is here entered not by means of the node numbers, but by means of the so-called feedback formula (further details are given in the chapter `Filter/Feedback`).

If no `Feedback` formula is entered, `AkAbak` multiplies the transfer functions of the listed `Filter` elements.

The input parameter of the filter network is the input voltage `Uin`, whose value is entered into the control dialogs of the simulation.

The output voltage of the filter network is the input voltage `U1` of the current network (Fig. 3 and Fig. 4).

If only `Filter` elements are entered, and no network is present, `U1` represents the output voltage of the last `Filter` element or the filter network.



## Script

The input medium for the simulation in AkAbak is the script. It contains all the data of the components, the structure of the networks and the position of the radiation elements. Comments can also be entered anywhere, so that the script can also document the circuit.

The script is simply a text with a certain structure. The data have a fixed format. The script is edited in the script windows provided for this purpose, using the built-in text editor (see Help/File, /Edit and /Search).

When a simulation is started, the script is interpreted and any formulae present are evaluated by the formula parser. The control dialog for the type of simulation then opens, so that further settings can be made. These data are saved along with the script. The calculation is then carried out and the results are displayed graphically in a diagram.

If an error occurs during the interpretation, the process is stopped and a short comment on the nature of the error is displayed in the status line. In the script, the line containing the error is displayed inversely.

Many dialogs also start the interpreter to read in the parameter of the element in which the cursor is currently located. Incorrect entries are not displayed in the dialog.

AkAbak can display several script windows simultaneously. Actions affecting the script always relate to the window that is currently open. With the aid of the clipboard, script text can be copied within a window or from one window to another. In any one window the text must not be longer than 32000 characters.

The font of the script windows is fixed. It is non-proportional, so that the script retains its tabular character. It has been found that the legibility of the script is improved by indenting the parameter of the elements by two spaces.

A script has a specific structure. It consists primarily of one or more so-called systems and, if appropriate, a definition part. Each system forms a framework for one current network and one filter network. Each script can calculate up to 10 independent systems or networks.

The definition part contains elements whose data are available throughout the script.

The input variable for all systems is the voltage  $U_{in}$ . The inputs to the systems are connected in parallel and are always connected to an ideal voltage source, so that they are decoupled at the input side. The magnitude of the input voltage  $U_{in}$  is entered into the control dialogs as a rms or peak value of the particular type of simulation.  $U_{in}$  may possibly be weighted by a *Filter* cascade (see below).

The output parameter is determined by the type of simulation. Physical parameter such as a velocity or a current are displayed as a peak value, regardless the setting of the input form. Levels are normed to the constant 1. The level  $L_p$  of the rms-sound pressure is normed to the threshold of hearing. One group of simulations investigates currents, voltages and velocities and pressure in the branches and at the nodes of the current network. It is described in the chapter 'Inspect'. The other group displays total parameter such as the overall sound pressure, the power, etc. The output parameter from all networks are summated and displayed here. These types of simulations are summarized in the chapter 'Sum'. The output curves can be processed further with the functions comprised in the 'Calc'-menu.

## Definitions

The description of the network elements in the script usually consists of their keyword, the parameter, node numbers and the radiation position. In all electroacoustic transducers, the details of the parameter are listed separately. They are represented by the so-called 'definition'. This definition precedes the first occurrence of the associated network element, usually at the start of the script. You can generally recognize the keywords of the definitions by the prefix `Def_`, for example `Def_Driver`.

Each definition is given an arbitrary, unique name (identifier). The associated network element has a parameter whose identifier is called `Def=`. `Def=` is followed by the name of the definition whose parameter values you intend the network element to use, for example: `Def= 'B1'`.

We have found this method of input useful in practice: Firstly, the network descriptions are often very complex; the extensive list of the driver parameter is therefore given separately to avoid cluttering.

Secondly, any number of network elements can refer to the same definition. In this case they all have the same parameter, but different network and radiation positions.

Furthermore, it is easier to test the characteristic of different drivers in the constellation if they are described centrally. For this purpose you enter the definitions of all drivers to be tested at the start of the script and then only change the names.

## Def\_Const

Another element of the definition part is `Def_Const`. This definition consists of a formula system, which is evaluated while the script is evaluated. The results are assigned to freely selectable identifiers, which you can then use as constants throughout the entire script. You can use this definition to alter parameter values centrally (see chapter Def/Def\_Const).

## Def\_Reflector

`Def_Reflector` is also in the definition part. This definition comprises the parameter of reflectors located close to the loudspeaker, such as the type of reflector, distance and angle of the loudspeaker with respect to the walls. If `Def_Reflector` is specified and a radiator element has the keyword `Reflection`, the effect of the reflectors on its radiation is taken into account. You can switch off `Def_Reflector` simply by prefixing the keyword `Off`.

## Def\_ListeningPoint

`Def_ListeningPoint`, finally, is a definition that shifts the origin of the baffle coordinate system. The details of the listening point in the control dialogs of the simulations and the details in the `Def_Reflector` definition are always relate to this origin. After you have built up a script step by step, you will often find that this reference point is unfavorably located. `Def_ListeningPoint` is a convenient means of changing this without having to modify all the position details.

## Filter elements in the definition part

`Filter` elements may be located in the definition part. All `Filter` elements that are listed before the first `system` keyword are multiplied together. The input voltage `Uin` which can be specified in the control dialogs is weighted with the result. The filter thus has a global effect. The output voltage is the source voltage of the succeeding systems with their networks and filter structures. The filters of the definition part cannot be networked with a `feedback` formula.

## Radiator

The `Radiator` network element is also referenced by the element name. This element is simply a radiator with the radiation impedance of a diaphragm. Its parameter consist of the details of the radiation surface. Since, however, this surface is in most cases the diaphragm area of another element, it is very useful if the `Radiator` element retrieves these details from that element. These are then also coupled by means of the `Def=` parameter. `Def=` is followed by the name of a preceding network element or by a definition (see chapter A-Net/Radiator).

## Off

You use the keyword `Off` to switch off definitions, systems or elements. When the interpreter encounters `Off` it ignores everything that follows, until the next keyword of a definition, a system or an element. You simply enter the word `Off` before the element to be switched off, separated by at least one space, tab or line break. If you enter `Off` before the `System` keyword, you switch off the entire system.

This option is particularly useful for summing simulations (chapter Sum). If, for example, your script describes a loudspeaker combination comprising a woofer and tweeter system, you can switch off one of the systems in order to investigate the radiation characteristic of the other system.

`Off` is also very hand for switching on and off the definition of the `Def_Reflector` reflector.

To control simulations which add the free field sound pressure or power another tool is available which uses Labels (see below chapter 'Labels').

### ☒ Example

The first system, called 'bass' has been switched off. Only the speaker element in the 'tweeter' takes part in the radiation.

```
off System 'bass'
  Bassunit Def='B1'
  Node=1=0
  x=0 y=-15cm z=0 HAngle=0 VAngle=0

System 'tweeter'
  Speaker Def='K1'
  Node=1=0
  x=0 y=0 z=0 HAngle=0 VAngle=0
```

### Note

`Off` is extremely useful. In very extensive scripts, however, it is often simpler to open a second script window and to work with a copy of the system you want to investigate.

## OutPort

The `OutPort` keyword is only used on one type of simulation, namely on the sum of electrical voltages from all networks (chapter `Sum/SumVoltage...`).

This type of simulation is very abstract, but is handy for investigating filter structures for cross overs of loudspeakers. The simulation summates the output voltages of the filter networks of all systems and the voltages via those elements that have the `OutPort` keyword in their parameter list.

### ☒ Example

If you start the 'Sum/ Sum Voltage...' simulation menu, the output voltage of the filter in the 'constant' system and the voltage across the resistor  $R=10\text{ohm}$  in the 'low pass' system are added together.

Since the nodes of this resistor are numbered in the opposite sequence, the inverted voltage is applied across the resistor. The 'low-pass' System is a low pass with inverted output voltage. This is added to the output voltage of the Filter. Since this is constant, the resulting overall transfer function is a high-pass.

```
System 'Constant'
  Filter
  {b0=1;}

System 'low pass'
  Coil Node=1=2 L=2.251mH
  Capacitor Node=2=0 C=11.254uF
  Resistor Node=0=2 R=10ohm OutPort
```

## Comment

The script is not only used for data input, but also documents the circuit that is being simulated. It is therefore important to be able to note additional information and instructions.

The interpreter ignores any text that is between the comment character `|` and the line end or another comment character. The comment character `|` is the vertical line.

On the standard US/British keyboard this is above the backslash (`\`) at the left-hand side of the bottom row.

Within the equation systems, the commentary additionally also ends with the formula separator, the semicolon.

### ☒ Example

Although the comments are not used very methodically in this example, it ought to make clear demonstrate to you how to use them:

```
| 4th order band pass with
| passive components and one filter element
System 'S1'
|Butterworth low pass
Coil |inductance| Node=1=2 L=2.251mH |series
Capacitor          Node=2=0 C=11.254uF |parallel
Resistor           Node=2=0 R=10ohm    |parallel

Filter | Butterworth high pass
fo=1kHz
{b2=1; |numerator
a2=1; a1=1.414214 |=sqrt(2); a0=1; |denominator}
```

## Labels

Labels can control the summation of the sound pressure or sound intensity vectors. For example, when you want to distinguish the total radiation of a vented enclosure and the radiation of the port only. Or, the radiation of a tweeter, the bass channel and the total sum.

Labels must be first specified in the script and can then be selected in the simulation control dialog. System names are also valid labels.

The output is then the sum of all radiators with the same label.

If no label is selected the total sum of all radiators is displayed (`<all>`).

For details see chapter 'Introduction/Radiation Environment/Labels'.

## Parameter, Numbers and Units

Parameter form part of a definition or an element and describe its properties. They follow the keyword of the definition or the element. They apply as far as the next keyword or until the end of the file. Their arrangement is not significant. The parameter can all be in one line or in different lines. Spaces or comments are also permitted between the individual parameter. The only important factor is that each parameter is followed by at least one space, tab stop or line break. For example:

```
Resistor 'R1'
Node=1=2 R=10ohm
```

The name `'R1'`, `Node=1=2` and `R=10ohm` all form part of the `Resistor` element.

Value assignments are always of the form `identifier=value`. Within the parameter there must be no spaces, tab stops or line breaks, for example before or after the equals sign and between the number and unit or text. No distinction is made between upper and lower case in the identifier.

## Node numbers

Node number entries in the current networks have the form:

$$\text{Node} = s = t = u = v = w = x$$

where **s** to **x** are the connection poles of the respective network element. The number of connection poles depends on the type of the element. If the last pole is not entered, it is set to zero.

You can enter any number in the range of 0...32000. The maximum number of nodes is 56. The node numbers need not to be ordered sequentially. Node number one is reserved for the network driving point. Node number zero is the ground.

The sequence of the node numbers in the entry determines the polarity of the potential difference between the nodes, and therefore the direction of the current arrow. If  $U_s$ ,  $U_t$ ,  $U_v$ ,  $U_u$ ,  $U_w$  and  $U_x$  are the potentials at poles **s**, **t**, **u**, **v**, **w** and **x**, the potential differences across the poles of the element are then:  $U_{st} = U_s - U_t$ ,  $U_{uv} = U_u - U_v$ ,  $U_{wx} = U_w - U_x$  etc.

The current arrow points in the direction of the alphabetic sequence of s, t, u, v, w and x. If the element is a two pole the flow points from the first to the second node (Fig. 5). If the element has more than two poles the simulated flow direction points always into the element (seen at the first pole of a port, Fig. 6).

The polarity results from the comparison with the potentials determined by the node numbers. If  $U_1$ ,  $U_2$ ,  $U_3$  etc. are the potentials at nodes 1, 2, 3 etc., then the potential differences from one node to the other are:  $U_{12} = U_1 - U_2$ ,  $U_{13} = U_1 - U_3$ ,  $U_{23} = U_2 - U_3$  etc.

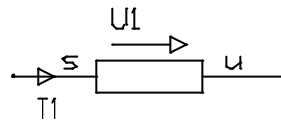


Fig. 5 Two pole flow assignment

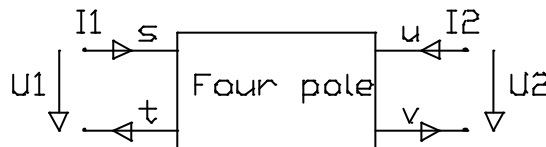


Fig. 6 Four pole flow assignments

### ☑ Example (Fig. 7)

An electroacoustic driver with two voice coils has 6 connection poles. Poles **s** and **t** are the terminals of the first voice coil and poles **u** and **v** are those of the second voice coil. Pole **w** represents the front side of the diaphragm and pole **x** the reverse side of the diaphragm. Pole **s** is at node 1, and is thus the input to the network. The input of the network  $U_1$  is at this node. In Fig. 7,  $U_1$  is represented by the potential difference  $U_{10} = U_1 - U_0$ , where  $U_0 = 0V$  is ground. The voltage  $U_{12}$  across the first voice coil is the difference of the potentials  $U_1$  and  $U_2$ . The current arrow of  $I_1$  points from pole **s** to **t**, i.e. from node 1 to node 2. The current arrow of  $I_2$  points from pole **u** to pole **v**. At the acoustic side, the potential is equal to the sound pressure  $p$ . The potential difference is  $p_{1020} = p_{10} - p_{20}$ . The current arrow of the volume velocity  $V_{10}$  points from pole **w** to pole **x**. Since  $V_{10} = -V_{20}$ , pole **w** represents the front side of the diaphragm and pole **x** the reverse side. The parameter of this network element is thus:  $\text{Node} = 1 = 2 = 2 = 0 = 10 = 20$

If the sequence of the node numbers is exchanged, for example in the manner:  $\text{Node} = 2 = 1 = 2 = 0 = 10 = 20$ , then the voltage  $U_{21} = U_2 - U_1 = -U_{12}$  is at poles **s** and **t** of the first voice coil. The polarity is thus reversed. If the two voice coils and drives of the driver are the same, nothing is transmitted from the driver in this case, since the two drivers cancel each other out.

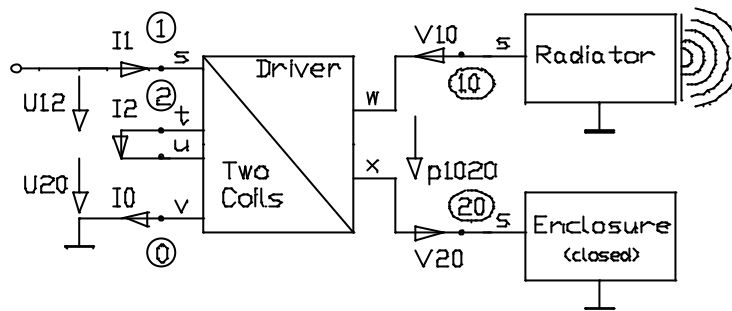


Fig. 7 Example of node numbering

## Numerical value assignment

Identifier=value+unit

### ☑ Example

dD=10cm

Vas=50L

Vb=1ft3

If a unit exists, it has to be entered. **A distinction is made between upper and lower case for the unit.** For example, R=12Ohm generates an error message, since the unit of resistance is ohm.

## Numerical format and range

The numerical values are entered as floating point numbers without thousand separation. The decimal separator may be either a comma or a period. Negative values are preceded by the minus sign.

If the range is not limited by the particular parameter, then the largest number that can be entered is:  $10^{20}$  and the smallest:  $10^{-20}$ . The mantissa is 7 digits long. 19-digit accuracy is used internally.

## Engineering-Format

The ENG-format can be applied for most non-complex units, for example: m, m<sup>3</sup>, V, ohm etc., for example:

**k** for kilo, **m** for milli, **u** for micro etc.

Exceptions are

inch	in, in <sup>2</sup> , in <sup>3</sup>
feet	ft, ft <sup>2</sup> , ft <sup>3</sup>
liter	L, Liter
grad	°, Deg
per cent	%

## Scientific Format

Complex units, such as m/N, Tm, etc. cannot be written in ENG format. You should use SCI format here (for example: 1.26e-6m/N).

For very large or very small numerical values, the computer has its own exponential notation. The lower case **e** is an abbreviation for times ten to the power of, for example:

$$1.26e3 = 1.26 \cdot 10^3$$

**Units and subunits used by AkAbak:**

SI-unit		Description	Subunits	Examples
m	meter	distance	k, c, d, m, u, n, p	dD=2.54cm
in	inch	distance		dD=1in
ft	feet	distance		dD=0.083ft
m <sup>2</sup>	square meter	area	k, c, d, m, u, n, p	SD=5.07cm <sup>2</sup>
in <sup>2</sup>	square inch	area		SD=0.78in <sup>2</sup>
ft <sup>2</sup>	square feet	area		SD=1ft <sup>2</sup>
m <sup>3</sup>	cubic meter	volume	k, c, d, m, u, n, p	Vb=10dm <sup>3</sup>
in <sup>3</sup>	cubic inch	volume		Vb=610.2in <sup>3</sup>
ft <sup>3</sup>	cubic feet	volume		VB=1ft <sup>3</sup>
L	liter	volume		Vb=10L
s	second	time	m, u, n, p	to=1us
Hz	hertz	frequency	T, G, M, k, m	1.6MHz
N	newton	force	G, M, k, m, u, n, p	104.1kN
Pa	pascal	pressure	M, k, m, u, n, p	32.89kPa
W	watt	power	M, k, m, u, n, p	95mW
V	volt	potential	M, k, m, u, n, p	12.32nV
A	ampère	current	k, m, n, p	76.13mA
dB	decibel	power level	k, m	150mdB
%	per cent			100%
Deg	degree	degree (360)		180Deg
°	degree	degree (360)		180°
m/s	meter/second	velocity		343.3m/s
m/s <sup>2</sup>	meter/second <sup>2</sup>	acceleration		55.8e-2m/s <sup>2</sup>
ohm	ohm	resistance	T, G, M, k, m, u, n, p	0.618Mohm
S	siemens	conductance	T, G, M, k, m, u, n, p	1.618uS
F	farad	capacitance	k, m, u, n, p	56pF
H	henry	inductance	k, m, u, n, p	32.6mH
kg	kilogram	mass	m, u, n, p and g	5.6g, 19mg
m/N	meter/newton	mechanical compliance		32.6e-4m/N
Ns/m	Newton seconds/meter	mechanical resistance		8.7Ns/m
kg/m <sup>4</sup>	kilogram/ meter <sup>4</sup>	acoustical mass		123.1kg/m <sup>4</sup>
m <sup>3</sup> /Pa	meter <sup>3</sup> / Pascal	acoustical compliance		360e-11m <sup>3</sup> /Pa
Pas/m <sup>3</sup>	Pascal second/ meter <sup>3</sup>	acoustical resistance		1.265e3Pas/m <sup>3</sup>
N/V	newton/volt	piezoelectric conversion factor		0.073N/V
Tm	tesla meter	magnetic flux density		12.8Tm
kg/m <sup>3</sup>	kilogram/cubic meter	density		1.1877kg/m <sup>3</sup>

## Assigning values from a formula system

Instead of a number, the identifier may also be assigned the result of a formula system:

```
Identifier={ formula system }
```

☑ Example:

```
HAngle={ h = 1e-2; Deg(ArcTan(h/5e-2)); }
```

The formula system is evaluated during the interpretation. The result of the last formula is assigned to the identifier. In this case the conversion of units is not supported. The result of the calculation is indicated by the respective SI unit.

The formula system is entered in curly brackets (braces) {...}. No spaces must be entered between the equals sign and the opening bracket. Within the formula system, the syntax of the formula parser applies (see section formula parser). Results evaluated in one parameter cannot be used in other parameter. Use the `Def_Const` definition if you have to evaluate globally valid values (chapter Def/Def\_Const).

## Runtime-Formula

Some elements are assigned a so-called runtime formula. The name indicates that this formula is calculated during simulation. You are able to access the frequency variable 'f' and  $\omega$  ( $=2\pi f$ ) and in some cases other reserved runtime variables. The syntax is the same as with all other formula systems in AkAbak. (See elements: `Filter`, `Impedance`, `Def_MeasRadiator` and `Coupler`)

## Exception:

For the `Feedback` element, the feedback formula is also written in braces. However, this formula is not evaluated by the formula parser. The syntax of the formula is similar, but the extent of the function is much smaller (see chapter `Filter/Feedback`).

## Alphanumerical value assignments

Parameter that are assigned a text have the following form:

```
Identifier='Text'
```

☑ Example:

```
Def='Any name'
```

```
Filename='Meas1.AKR'
```

The number of characters is limited to 20. The text may contain spaces. The entry must be in quotation marks ('...' or '...', not in accent characters).

## Formula system

This type of parameter entry does not need an identifier and is used by the definition `Def_Const` and the `Filter` element. The same applies as discussed above under 'value assignment from a formula system'; except that there is no identifier (see chapter `Filter/Filter` and chapter `Def/Def_Const`).



## Names

Any definition and any element may be assigned a name. The name has no identifiers:

```
'Type 234A'
```



Example

```
System 'Bass'
  Resistor 'R1' Node=1=10 R=2.2ohm
  Driver 'D1' Def='Type 234A' Node=10=0=100=110
...
```

The number of characters is limited to 20. The text may contain spaces. The entry must be in quotation marks ('...' or '...', not in accent characters). System's and elements which shall take part at the 'Inspect'-simulation should always be named. Otherwise an element is not listed in the inspect-control-dialog (see chapter Inspect).

Definitions or elements to which network elements refer have to be given a name.

In the example above, the `System` is called 'Bass'. The name of the resistor is 'R1'. The `Driver` element has the name 'D1' and refers to the definition with the name 'Type 234A'.

## Keyword

Keywords are used as the identifiers of the definitions, as the introduction of a new system, and as identifiers of an element.

Other keywords are similar to switches. They appear within an element and control the characteristics of the element.



Example

```
Def_Driver 'Type 234A'
  Meas_Baffle1
  dD=5cm dD1=8mm tD1=1.2cm
....
System 'S1'
  Driver Def='Type 234A' Node=1=0=2=0
...
```

Here, the `Def_Driver` keyword introduces the definition. The definition extends as far as the `System` keyword or until the next definition. The keyword `Meas_Baffle1` appears within `Def_Driver`. This tells the program which measuring conditions were used to determine the specified parameter. In this case, the driver is installed in a baffle during the measurement and the reverse side of the diaphragm has been closed with an enclosure (see chapter Def/Def\_Driver).

The keyword `System` is used to create a new system. All the elements following `System` form part of a network. It is valid until the next `System` keyword, or as far as the file end.

`Driver` is the keyword of the network element.

## Formula Parser

The formula parser is a part of the interpreter. It interprets and evaluates mathematical formula. The parameter of the elements can thus be assigned to the result of a formula; if, for example you want the simulation to represent a temperature-dependent value, or if position data depend on a geometrical figure that can be

described using a formula (see below). The `Filter` element always uses the formula parser for its input. The values assigned to the reserved identifiers `b0=`, `b1=`, ... and `a0=`, `a1=`, ... within the formula system are the coefficients of the rational transfer function of the filter (see chapter `Filter`). Since the parser can calculate with complex numbers, the filter coefficients can be expressed by, for example, the poles and zeroes of the transfer function.

AkAbak's formula parser can evaluate individual formulae or entire formula systems. Results of formulae calculated in the `Def_Const` definition even apply throughout the script and can be either assigned directly to the identifiers or used in further formula systems. The formulae are evaluated during the interpretation of the script, shortly before the simulation itself is carried out.

Some elements are assigned a so-called runtime formula. The name indicates that this formula is calculated during simulation. You are able to access the frequency variable '`f`' and '`ω`' ( $=2\pi f$ ) and in some cases other reserved runtime variables. The syntax is the same as with all other formula systems in AkAbak. (See elements: `Filter`, `Impedance`, `Def_MeasRadiator` and `Coupler`)

In the script, the formula system is always written within curly brackets (braces). Outside the brackets, script format applies, and within them the formula parser format.

#### ☒ Example:

1. First the constants `b`, `h`, `z`, `x` and `HAng` are evaluated (see also example in chapter `Def/Def_Const`). These identifiers are freely selected. Since they appear in the formula system of `Def_Const`, they can be accessed throughout the script. In this example the two speaker elements use the constants for positioning.

```
Def_Const
{  b=30e-2;  h=10e-2;  z=h/2;    x=b/4;
  HAng=Deg(arctan(x/z));
}
...
System  's1'

Speaker  'left'  Def='K1'
Node=1=0
x={x}    z={z}    HAngle={HAng}

Speaker  'right'  Def='K1'
Node=1=0
x={-x}   z={z}    HAngle={-HAng}
```

2. Example for a runtime-formula. '`ω`' is the angular frequency of the abscissa:

```
Impedance  'I1'  Node=100=110
Z={
  wo=2*pi*1000;  z1=1;  z2=-1;  p1=-1+j;  p2=-1-j;
  wn=w/wo;      |runtime variable of abscissa
  Num=(jwn - z1)*(jwn - z2);  Denom=(jwn - p1)*(jwn - p2);
  Z=Num/Denom;
}
```

## Syntax and format

The format differs from the rest of the script in two points:

- 1) Spaces, tabs and line breaks are permitted, even within the individual formulae.
- 2) The individual formulae are always separated by a semicolon - even if a line break follows (this can be omitted from the last formula).

This modification is used not only to make formulae easier to read, but also so that very long formulae can extend over several lines.

A formula system consists of one or more formulae. Each of these formulae has an identifier and the formula itself. The identifier is to the left of the equal sign, the formula to the right thereof. Any identifier can only be used once. The parser evaluates the formulae in the order in which they have been entered. Then the identifiers are constants and represent a numerical value. The identifiers of formulae already evaluated can be used as constants in subsequent formulae.

The last formula of a formula system does not necessarily require an identifier. If the formula system is assigned to the parameter of an element or of a definition, the result of the last formula of the formula system is assigned to it.

The maximum length of a formula system is 4000 characters.

## Numerical constants

The numerical values are entered as floating point numbers without thousand separation. The decimal separator may be a comma or a period.

Negative values are preceded by the minus sign (without space).

The imaginary unit is identified by the letter 'j' ( $j^2 = -1$ ). If 'j' comes before a number (for example between the minus sign and the value), before the name of a constant or function, or before a bracket, this is equal to multiplying the following term by the imaginary unit. The syntax is thus similar to that of a minus sign.

### ☒ Example

```
{ h = -jSin(j - -j5); k = -j(5 + j^(Sin(3))) + h }
```

For very large or very small numerical values, the computer has its own exponential notation. The lower case 'e' is an abbreviation for 'times ten to the power of', for example:

$$126\text{e}-6 \equiv 126 \cdot 10^{-6} \text{ (Scientific Format SCI).}$$

The largest and smallest numbers that can be entered are:

$$1\text{e}20 = 10^{20} \text{ or } 1\text{e}-20 = 10^{-20}$$

The mantissa is 7 digits long. 19-digit accuracy is used internally.

### ☒ Example

```
{ a=-110.34 + -j1.3e-2; }
```

The identifier for the constant  $\pi=3.14159...$  is called `Pi`. The growth constant  $e=2.718...$  is entered via the function `Exp(1)`.

### ☒ Example

```
{ SD=pi*sqr(dD)/4; e=exp(1); }
```

## Identifier

The identifiers come to the left of the equal sign and after the calculation constants, and therefore represent a numerical value. The length of the identifier is limited to 20 characters. Any characters may be used except for the operator characters:

`+, -, *, /, ^, (, ), {, }, _` etc.,  
point, comma, and the comment character `|`.

The name must not start with 'j', but can start with 'J'. The name should not be the same as a function name or the constant Pi.

## Brackets

Round brackets ( ) may be used within the formulae and nested according to the rules of arithmetic.

## Operators

There are five binary operators

+	addition of complex or real numbers.
-	subtraction of complex or real numbers.
*	multiplication of complex or real values.
/	division of complex or real values (divisor not equal to zero).
^	exponential base is a complex or real number, exponent is a real number

## Runtime-formula

This kind of formula is calculated during simulation. You are able to access the frequency variable 'f' and  $\omega$  ( $=2\pi f$ ) and in some cases other reserved runtime variables. The syntax is the same as with all other formula systems in AkAbak. (See elements: *Filter*, *Impedance*, *Def\_MeasRadiator* and *Coupler*)

## Functions

Functions always have the form:

`name(argument)`

Name is the function name. No distinction is made between lower and upper case. You must not use function names as identifiers of the formulae. The name may be preceded by unitary operators - such as the minus sign and the imaginary unit 'j'. The argument is another expression and is written in brackets. If the function evaluates complex arguments, a 'z' is entered as the argument name in the following list instead of an 'x'.

<code>abs(z)</code>	absolute value: $\text{abs}(z) =  z $
<code>sgn(x)</code>	sign function: $\text{sgn}(x) = \begin{cases} +1 & \text{for } x > 0 \\ 0 & \text{for } x = 0 \\ -1 & \text{for } x < 0 \end{cases}$
<code>exp(z)</code>	exponential function: $\text{exp}(z) = e^z$
<code>ln(z)</code>	natural logarithm
<code>log(x)</code>	logarithm to base ten
<code>si(z)</code>	$\text{Si}(z) = \frac{\text{Sin}(z)}{z}$
<code>cos(z)</code>	cosine
<code>sin(z)</code>	sine
<code>tan(z)</code>	tangent
<code>cosh(z)</code>	hyperbolic cosine
<code>sinh(z)</code>	hyperbolic sine
<code>tanh(z)</code>	hyperbolic tangent

<code>arctan(x)</code>	arc tangent
<code>arcsin(x)</code>	arc sine
<code>arccos(x)</code>	arc cosine
<code>sqr(z)</code>	square of z: $\text{Sqr}(z) = z^2$
<code>sqrt(z)</code>	square root of z: $\text{Sqrt}(z) = \sqrt{z}$
<code>rad(z)</code>	radian of z: $\text{Rad}(z) = \frac{\pi}{180} \cdot z$ (for converting angles from degrees to radians).
<code>deg(z)</code>	degree of z: $\text{Deg}(z) = \frac{180}{\pi} \cdot z$ (for converting angles from radians to degrees)
<code>re(z)</code>	real component of z
<code>im(z)</code>	imaginary component of z
<code>APtoRI(z)</code>	Amplitude+phase is converted to real+imaginary
<code>LPtoRI(z)</code>	Level+phase is converted to real+imaginary
<code>LpPtoRI(z)</code>	Sound pressure level+phase is converted to real+imaginary

The latter three function are mostly used for data import.

## Integration

The integration statement is not a formula, but a command and has the form:

```
identifier=integral(constant, final value, number);
```

`Identifier` is the name of the formula again and, after the integral has been calculated, is a constant that represents the result of the integration and can be used again subsequently.

`Integral` is the keyword of the operation. It cannot be used within a formula. The parameter of the statement are written in brackets. `Integral` forms the numeric integral according to the '3/8 method' of the result of the last formula that occurs directly before the `Integral` command.

`Constant` is the name of the identifier of a previously defined formula. It is the integration variable and contains the lower limit of the integration interval.

`Final value` is the upper limit of the integration interval. `Final value` may be a numerical value or the name of an identifier of a previously defined formula.

`Number` defines the number of summands of the numerical integration. It does not need to be specified. If it is omitted, the parser sets it to `number=22` by default. If the number of summands is increased, the accuracy of the integration may be improved.

Integrations cannot be nested., i.e. there must be no further integrals between the integration variables (`Constant`) and `Integral`.



Example:

The integral within a formula system in the `Def_Const` definition is to be evaluated:

$$I = \int_0^2 (x+1)dx = 4$$

For the test, we use the Abakus program or evaluate the formula analytically. The integration variable is  $x$ .  $x$  is first assigned the value zero. That is the lower limit of the integration interval.  $y=x+1$  is the integral. The upper limit of the integration interval is given by the second parameter in the `Integral` statement and is  $x=2$ . Since the third parameter is not specified, Abakus sets `number` to 22. The constant `I` contains the value 4 after the integral has been evaluated and can be subsequently used for further calculations.

This formula system could also appear before any parameter. The parameter is then assigned the value of the constant  $h=5.5$ .

```
Def_Const
{ a = 1;
  x = a - 1;
  y = x + 1;
  I = Integral(x, 2);
  h = I + 1.5;
}
```

## Radiation in General

Radiation is the term used for the propagation of sound waves into a room. A wave is generated by the sound source and measured at the listening point. The sound source consists of the vibrating diaphragm, the baffle, nearby refraction edges and reflectors. The listening point is located in the free space. At the listening point, the acoustic pressure of the radiators is summated according to modulus and phase.

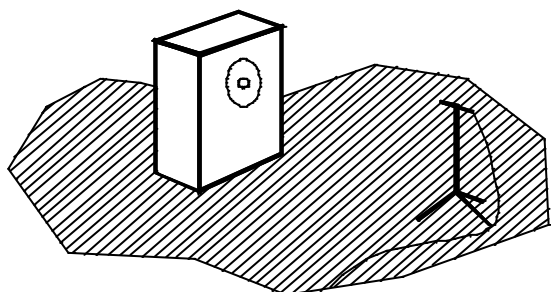
The calculation of the radiation is especially difficult to understand and calculate if the wavelength is of the order of magnitude of the dimensions of the sound source. This is true in general and very likely true for all problems that AkAbak will be used to solve. The physics of radiation is generally very highly advanced. A brief look at the relevant literature shows how extremely complex the material is. Radiation cannot be calculated to any desired numerical accuracy, especially not under the complicated circumstances found, for example, in loudspeaker technology or ultrasonics.

One of the most important simulations in AkAbak relates to radiation. To provide a good compromise between speed and precision of calculation and effort, the program 'only' processes the most important and basic characteristics of radiation. Since it is clear that the simulated curves are always incomplete in some respect, it is worth asking what the purpose of the simulation is. The intention is almost always to display the effect of one characteristic or another - whether to test a design or an idea or to gain a clear idea about how a particular element functions. Certainly no-one would produce simulated designs blindly, in full confidence that the spectral curve of the actual system will be identical to that of the simulated curve.

## Experiment

For simulation to be meaningful, it is critically important for the user to correctly understand the radiation - at least to the point where he is capable of distinguishing between certain important phenomena. As an approach to the problem, it is helpful to carry out a simple experiment:

A loudspeaker enclosure is located on the floor, but far removed from any walls. (In acoustics, adjectives like 'far' and 'large' are always relative to the wavelength). The enclosure contains a conventional loudspeaker. A microphone measures the sound level at a certain distance. The microphone is positioned on an extension of the diaphragm axis. By 'axis' it is meant the perpendicular to the diaphragm surface of the loudspeaker.



The spectrum of the sound level is measured. The measured frequency range is selected wide enough for the biggest wavelength to be considerably larger than the dimensions of the box and, naturally, of the diaphragm, and for the wavelength of the highest frequencies to be considerably smaller than the depth of the diaphragm cone. For example, in the range from 20Hz to 20kHz, the wavelengths are approx. 17m to 1.7cm.

## Observations

- What does the sound-level spectrum from such a system indicate?
- How do the radiation elements affect the curve?

The radiation elements acting here are: The diaphragm cone, the baffle with the sound-diffracting edges and the floor as reflector.

## Diaphragm cone

If the wavelength is much larger than the diaphragm, the reproduction curve is equal to that of the flat, rigid diaphragm. The sound-pressure curve is proportional to the diaphragm acceleration. By contrast with the flat, rigid diaphragm, however, the reproduction at high frequencies is more or less rippled, and after moderate amplification, the level drops strongly. Selective amplification and damping are close together in this frequency range.

## Baffle

The baffle represents the extension of the diaphragm. It is acoustically rigid, flat, and does not vibrate. The ideal baffle reflects the acoustic energy, so that a hemispherical wave propagates. The acoustic pressure increases to twice the value of the free surround radiation. The ideal case is the so-called 'infinite baffle', i.e. the edges of the baffle are, relative to the wavelength, far enough away from the radiator. In practice, on the other hand, one deals with a finite baffle, as in the example here. At very low frequencies, the average sound level is 6dB lower than at very high frequencies. In the frequency range in which the wavelength is comparable with the dimensions of the baffle, sound refraction occurs at the edges of the baffle - manifested by an 'up and down' effect in the sound-pressure curve.

## Reflector

The reflector has similar characteristics to the baffle. The surface of the reflector, however, is located at a certain distance from the diaphragm center. At very low frequencies, at which the wavelength is much greater than this distance, the reflector has the same characteristics as an infinite baffle. The sound level is amplified by 6dB. With increasing frequency, however, this amplification decreases and, after initial ripple in the sound pressure curve, disappears entirely.

## Various listening angles

It is also valuable to measure the sound level from various listening angles. If you rotate the loudspeaker enclosure through, for example, 180°, so that the loudspeaker radiates backwards, you can recognize the acoustic shadow at high frequencies. The shorter the wavelength becomes, relative to the housing dimensions, the less sound is diffracted around the edge of the baffle. At very high frequencies, virtually no sound pressure can be measured. The reason for this, however, lies not only with the diffraction effects of the baffle edges, but also with the diaphragm. To illustrate this, the sound pressure curve can also be measured at a listening angle of 90°. The level curve is similar to that measured on-axis, except that in the upper frequency range the sound pressure decreases with strong ripple. It is obvious that no diffraction effects can occur if no sound is radiated laterally.

Another interesting feature is the diagram of directivity characteristics. To illustrate this, with the generator, set third-octave random noise in the upper frequency range and measure the sound level at different listening angles. This illustrates that the level fluctuates and decreases all the more as the listening angle increases.

These observations are a useful introduction. It is advisable and extremely informative to carry out a few experiments yourself here to gain a clear idea of the relationships. Extend the experiment and investigate the sound pressure level, for example in the presence of various reflectors. What happens when the box is set up in a corner of the room? What does the spectrum look like if the loudspeaker enclosure is rotated through 180° in this case?.

How do the sound level and the directivity diagram vary when two radiators are present? Vary the position of the radiators or reverse the poles of the terminals.

Or: Insert the loudspeaker backwards into the baffle so that it radiates through a small vent. What effect does this have on the sound pressure variation? In this experiment it is also very interesting to see how the response is changed when the aperture is partly covered. The electrical driving-point impedance of the driver can also be observed.

## Point radiator

The point radiator radiates the sound uniformly in all directions. The radiation directivity is independent of frequency. Point radiators form the basis of geometrical acoustics, in which diaphragm forms are 'scanned' by point radiators.

The outgoing wave generated by a point source is spherical. The pressure gradient is only in one dimension, so that the one dimensional wave-equation can be applied.

## Diaphragm

In acoustics, a diaphragm means a vibrating surface that radiates sound. This surface may have any shape and may be made of any material. For example, the cross-sectional surface at the end of a vent or a horn is a diaphragm. The diaphragm surface can vibrate in various ways. The best known is the diaphragm that vibrates in a piston-like manner. In addition, there is also a 'breathing' sphere, as at the output of a horn, the linear radiator (side) and the point radiator. The surface generally does not vibrate as a whole, but is divided into sub-vibrations. These natural vibrations of the diaphragm depend on its shape, on the material, the clamping, suspension, etc. Radiating cross-sectional areas of vents and horns can also have a location-dependent distribution of the diaphragm movement, which is principally produced by dispersion during wave propagation.

Acoustic radiation is always associated with mechanical vibration of the diaphragm. It is immaterial whether the diaphragm surface vibrates coherently or fragments into transversal waves. Sound radiation exerts a force at each point in the room, i.e. not only at the listening point but on the diaphragm surface itself. The quotient of this force and the diaphragm velocity is called radiation impedance.

Radiation impedance is a frequency-dependent parameter. It is intermeshed with the mechanical elements of the driver and thus exerts a critical influence on the transfer characteristics of the radiator. For example, the acoustic power gain of the radiator is proportional to the real component of the radiation impedance. At low frequencies the imaginary component behaves as a mass and constitutes several per cent of the effectively vibrating mass  $M_{ms}$  of the diaphragm.

The acoustic waves in the vicinity of the diaphragm surface follow the mechanical movement. With respect to the listening point, they are summated according to modulus and phase. This transmission is generally characterized by strong interference, especially at relatively high frequencies.

The calculation is performed by subdivision of the diaphragm into small subdiaphragms and subsequent summation or integration of the partial sound pressures. The subdiaphragms formed are considered as flat, rigid ideal diaphragms (finite element) moving in the manner of a piston or as a 'point' on the surface (point radiator). In this case, finite elements are typically interconnected and point radiators are considered as independent units.

## Piston

A diaphragm is called piston-shaped when its surface is flat and it moves coherently in one direction. There are thus no natural vibrations of the diaphragm. In many investigations, a diaphragm that is otherwise complicated in shape can be regarded as a piston if the wavelength is considerably greater than the dimensions determining the shape. Partial vibrations can only occur from a specific frequency upwards. Below this frequency, the diaphragm vibrates in phase. Only inertia effects due to the finite stiffness of the diaphragm material lead to some parts of the diaphragm vibrating more intensively than others. This effect, which depends mainly on the position at which the driving force acts, is known as mass reduction.



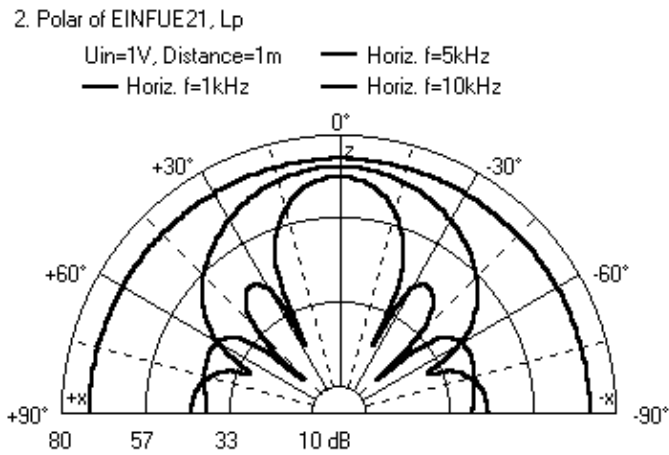


Fig. 8 Directivity characteristics of a flat, circular diaphragm at 1kHz, 5kHz and 10kHz

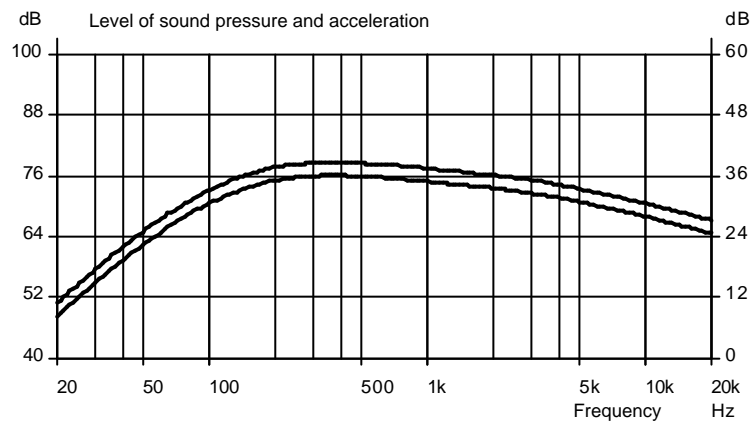


Fig. 9 On-axis sound pressure level of flat diaphragm and diaphragm acceleration (as level)

The directivity characteristics of a piston diaphragm are illustrated in a typical diagram (Fig. 8). Slight variations are caused by outer diaphragm shape (rectangular/circular) and by any holes present in the diaphragm (ring radiators). Since the surface is flat, there are no interferences in the direction perpendicular to the plane - i.e. on-axis - or only at extremely high frequencies, if the listening point is far enough away from the diaphragm (far field).

The on-axis acoustic pressure transmission of a rigid piston is similar to the transmission of the diaphragm acceleration (Fig. 9).

Since the travel time of all points on the diaphragm is the same, there is also no temporal distortion. The shape of a pulse is therefore retained (Fig. 10).

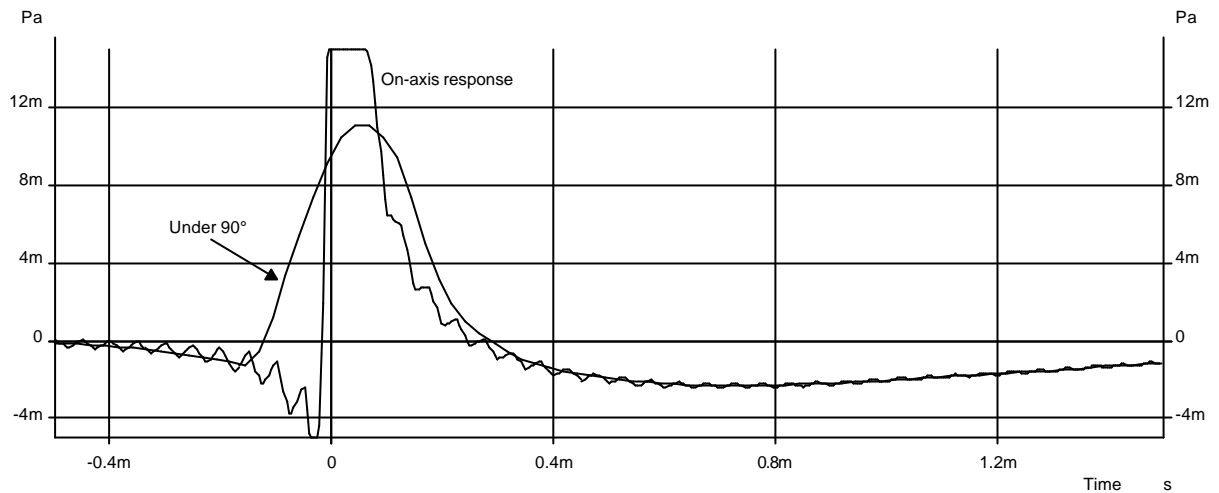


Fig. 10 Pulse response of a flat diaphragm, on-axis and at 90°

If the listening point is not on-axis, the picture is completely different (Fig. 11). At low frequencies, the transmission is the same as that on-axis. At very high frequencies, on the other hand, interferences ensure there is less radiation laterally. Intense rippling with partial complete extinction can be observed in this case. It appears as if the diaphragm focuses the sound.

As shown in Fig. 10 the time-impulse response under 90° is smeared. Some parts of the diaphragm are located more near the listening point whereas others are more distant.

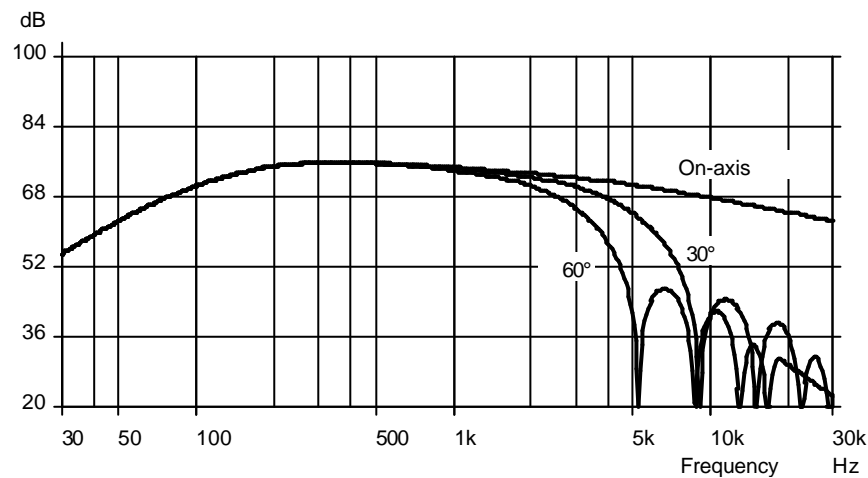


Fig. 11 On-axis sound pressure level, 30° and 60°; flat diaphragm

The term 'focusing' is not entirely correct in this context, since the acoustic power decreases overall at high frequencies. A typical frequency, known as the directivity frequency, is defined in this case, from which the directivity effect of a diaphragm leads to loss in power:

$$f_D = \frac{c}{\pi \cdot dD} \approx \frac{110}{dD}$$

where  $c$ : velocity of sound,  $dD$ : outer diaphragm diameter

The -3dB cut-off frequency for the acoustic power is for most loudspeaker diaphragms about one octave above  $f_D$ , depending on the diaphragm shape, mass reduction and other parameter.

The graphs of the directivity characteristics are obtained by integrating across the diaphragm surface. For standard diaphragms, there are closed solutions: The directivity characteristics of the ideal rectangular piston-shaped diaphragm is described by the si function. If the diaphragm is circular, the analysis results in a Bessel function. The curves of the si and Bessel functions are very similar. The different travel times of the subdiaphragms at the listening point lead to a distortion in the time domain, so that the form of an impulse is lost.

One variant of the piston-shaped diaphragm is the ring radiator. The diaphragm shape of an ring radiator may be outwardly rectangular or circular. In the center is a hole. The directivity characteristic of the circular ring radiator is a combination of Bessel functions.

If parts of the diaphragm can no longer follow the drive force because of inertia and the finite stiffness, the effect of mass or area reduction is observed. By this is meant that, with respect to the drive, a smaller mass acts and the radiation takes place over a modified surface. In this case it is important to distinguish the location of the point of action of the force. If the force acts in the center of the diaphragm, for example, first the edge regions of the diaphragm are decoupled. If the drive is at the diaphragm edge, there is a hole in the diaphragm center. In the first case, the directivity characteristic will naturally spread. The directivity frequency  $f_D$  is shifted to higher frequencies. The power bandwidth increases. In the second case,  $f_D$  does not change. In loudspeaker technology, this effect is well known, and designers attempt to use the mass reduction effect to advantage.

The latter effect is an integral result of the transversal vibration of the membrane. As soon the driving frequency is equal one of the eigen-frequencies of the diaphragm strong excitation is performed by the material. Naturally this vibration pattern is very difficult to describe mathematically. Hand in hand with this vibration a more or less diffuse radiation takes place.

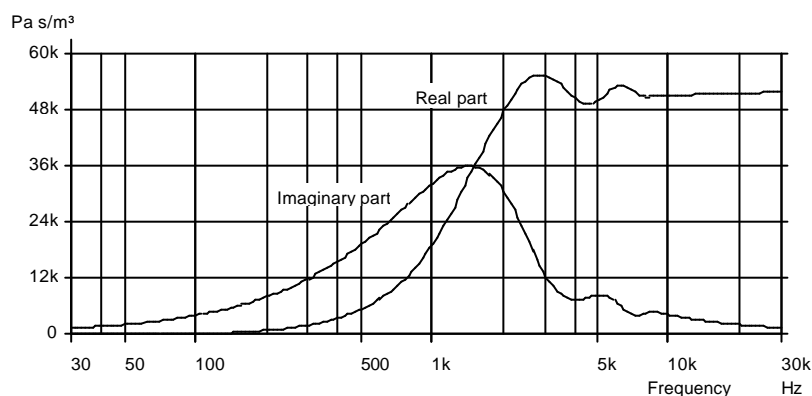


Fig. 12 Radiation impedance of flat diaphragm ( $dD=10\text{cm}$ )

As already mentioned, the acoustic pressure reacts on its own diaphragm, so that force divided by diaphragm velocity forms an impedance. This so-called radiation impedance acts together with the mechanical elements of the driver. With piston-shaped driver, the curve of radiation impedance can be calculated by a closed function in series form (Fig. 12). There is little difference between curves of the functions for round and rectangular diaphragms if the ratio of the sides of the rectangle is not too extreme. Other diaphragm shapes, such as the cone or dome, behave in a similar way.

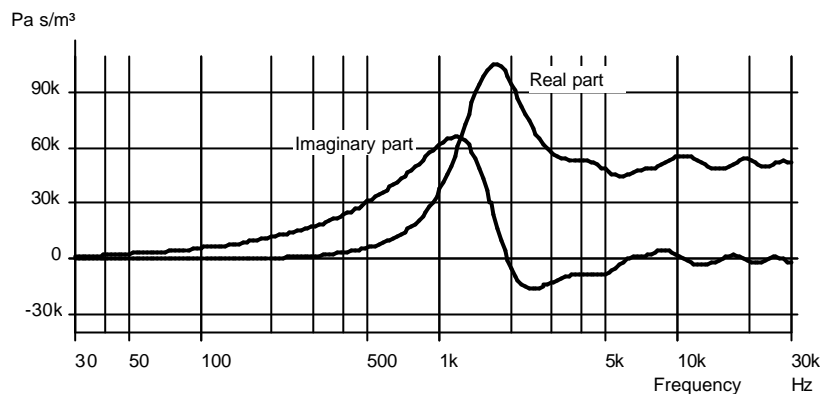


Fig. 13 Radiation impedance flat diaphragm ( $dD=11\text{cm}$ ) sunk by  $t_1=2\text{cm}$ ; top: real component and bottom: imaginary component

The greatest deviation is found when the diaphragm radiates across a hollow space: if, for example, the diaphragm itself is concave or the diaphragm is, for example, recessed into the baffle (Fig. 13).

The real component of the radiation impedance is constant above the directivity frequency  $f_D$ . That means that in this range there is hardly any interaction with the subdiaphragms. At low frequencies, the resistance increases with frequency. In the vicinity of  $f_D$ , the response is most sensitive to the diaphragm shape and the radiation environment.

The imaginary component of this impedance has, at low frequencies, the properties of a mass. In the vicinity of  $f_D$ , the reactance reaches its maximum and at high frequencies approaches zero. For radiation across hollow spaces, the reactance in the vicinity of  $f_D$  may assume the character of a compliance.

## Tympanum and plates

Strictly speaking, any diaphragm has more or less the properties of a plate or a tympanum. In reality there is no ideal case. The conditions of the vibrating plate or of a tympanum therefore also apply to any other diaphragm shape.

If the diaphragm is not broken up into natural vibrations, the surface of the tympanum vibrates like a 'breathing' sphere. Otherwise a variety of vibration patterns are produced, which depend on the tension, diaphragm shape, action of force, etc..

The vibration pattern of bars, plates and shells is most complex since there are more inherent forces at work as for example at the stretched membran.

The description of the natural mechanical vibrations in closed mathematical form can only be given in certain special cases. Otherwise it can be determined with the aid of the finite element method (FEM). One often attempts to convert the descriptive parameter into equivalent parameter of the diaphragm oscillating in a piston-like manner. Loudspeaker and microphone technologists attempt to damp the partial vibrations, which are associated with the mass reduction. In the sound pressure curve, the radiation of the partial vibrations shows a typical profile of closely spaced resonances and antiresonances with high qualities. There are, however, cases in which the natural vibrations are distributed precisely so as to cancel out the radiated energy. This effect is unfortunately limited to a few selected frequencies.

If strong natural vibrations occur, radiation is more or less diffuse since each transversal wave generates sound whose main direction is perpendicular to the wave movement. But there exists also some other types of wave propagation which is the result of the interaction between the air and the vibrating surface.

## Cone

The conical diaphragm is the typical loudspeaker diaphragm. With dynamic drivers, the voice coil exerts its force at the tip of the cone. The mechanical vibration characteristics of the conical diaphragm of a loudspeaker is in practice optimized so that the cone oscillates as a piston in the direction of the cone axis. Natural

vibrations are suppressed by skillful shaping and damping in the diaphragm material. If parts of the diaphragm can no longer follow the high accelerations at high frequencies, the effective cone diameter becomes smaller. If the diaphragms have been well designed, the energy liberated is partly converted into movement and partly into frictional losses in the diaphragm material.

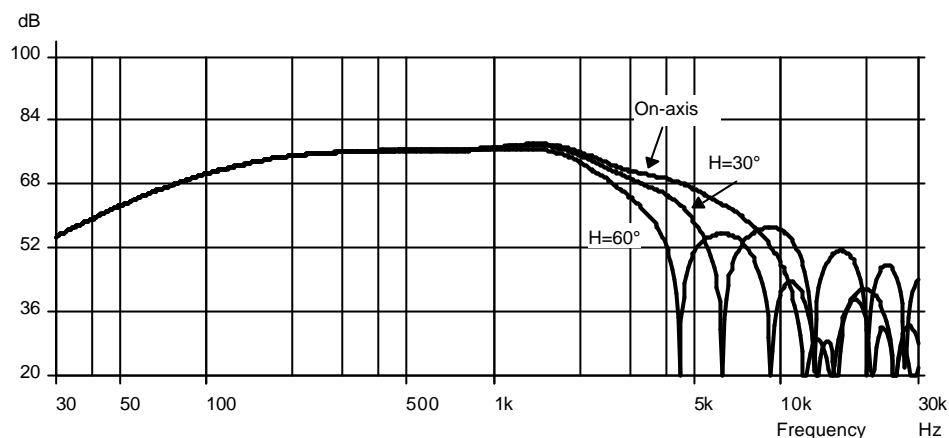


Fig. 14 On-axis sound pressure level, 30° and 60°; conical diaphragm

The radiation of the cone differs considerably from that of the flat diaphragm (Fig. 14). At high frequencies, in particular, interferences also occur on axis. A typical curve is the comb-filter-like curve of the sound-level spectrum at high frequencies.

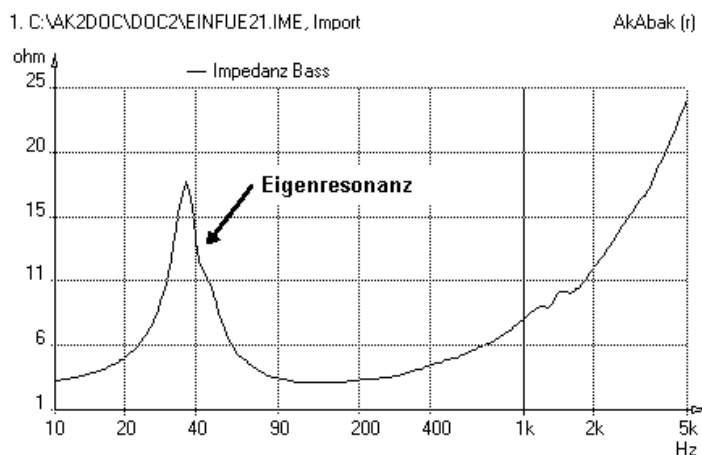


Fig. 15 Natural resonance of the diaphragm parts in the curve of the electrical driving point impedance (measured)

In addition to natural vibrations of the diaphragm surface itself, resonances often occur, which may be caused by the joining of the various part of the conical diaphragm. For example, the voice-coil support, the voice coil, the connection wires, the centering suspension and the dust cap all resonate with the cone. At low frequencies, such resonances are in the vicinity of the basic resonance  $f_s$  of the driver and are, for example, clearly visible in the frequency response of the electrical input impedance of the driver (Fig. 15).

Tweeters with a conical diaphragm, in particular, often lack the flexible suspension at the outer rim of the diaphragm. These can be regarded as constrained conical diaphragms. Loudspeakers with this design may have a pronounced 'smooth' sound-level curve and a broad radiation characteristic up to very high frequencies. However, appropriate measurements manifests the intensive natural vibrations of the diaphragm.

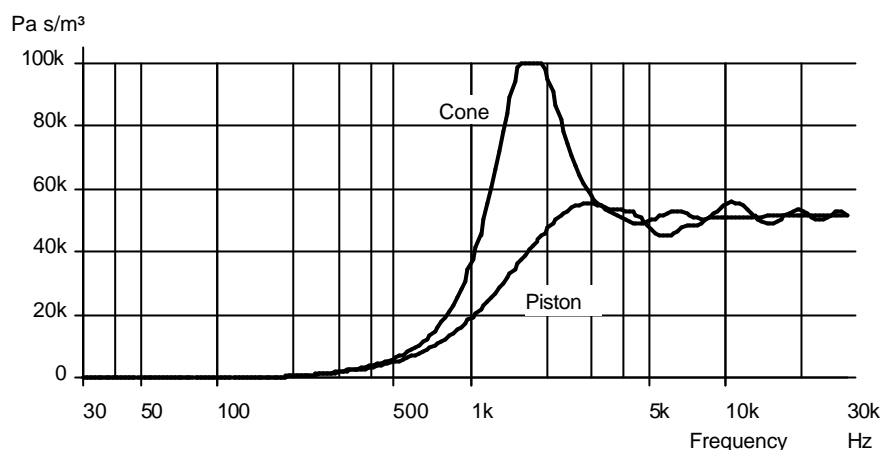


Fig. 16 Curves of radiation resistance: Concave (peak) and flat diaphragm shapes.

Concave diaphragm shapes such as conical diaphragms usually have a regular gain in the upper frequency range. At even higher frequencies, the level then decreases sharply with frequency. The reason is a selective increase of the radiation resistance in the vicinity of the directivity frequency  $f_D$ . The deeper the cone is shaped, the more pronounced this gain (Fig. 16).

## Dome

A dome is a spherical diaphragm with either convex or concave curvature. The voice coil exerts its force on the outer rim of the diaphragm. Dome diaphragms are used predominantly for medium range speakers and tweeters. The statics of the dome shape provide a high internal stiffness, so that relatively lightweight diaphragms can be produced. If, in the dome, parts of the diaphragm can no longer withstand the acceleration, an ring radiator is the result. In addition to partial vibrations, an annulus, decreasing in size as the frequency increases, then radiates.

The curve of radiation impedance of the convex dome is similar to that of the flat diaphragm. The cavity of the concave dome generates a curve similar to that of the cone. In the vicinity of the directivity frequency, the sound pressure is amplified.

## Baffle

The baffle is one of the most important acoustic elements, and also the one that is most difficult to describe mathematically. A baffle is always present if the propagation medium for the advancing sound wave changes. The sound wave is refracted. One part of the energy continues to pass through the medium behind the baffle; the rest is reflected. A baffle forms when, for example, air layers of different temperature lie one on top of the other. If the density and stiffness of adjacent media are very different, the baffle is known as sound reflecting or acoustically hard. An acoustically hard wall reflects sound totally. The angle of incidence is equal to the angle of reflection. This fact is analogous to that in optics. In acoustics, however, the problem is that the wavelength is not significantly smaller than the dimensions of the baffle. In this case there are complicated diffraction effects at the edges of the baffle.

If the baffle absorbs sound or vibrates itself, reflection is no longer total. The angle of reflection and the sound pressure amplitude and phase are distorted.

In loudspeaker technology the baffle is used for two reasons. Firstly, the baffle increases the travel time between sound pressure at the front and reverse sides of the diaphragm, and by this increasing the bass reproduction range. The baffle is usually formed to an enclosure to suppress the rear radiation. Secondly, a large baffle doubles the sound pressure by reflecting the sound. If the dimensions are not large enough with respect to the wavelength, only a part is reflected. The rest radiates into the room behind. At very low frequencies it is as if the baffle were no longer present. The level is 6dB lower than at very high frequencies.

If there is no sound diffraction in the frequency band under observation, the baffle is known as 'infinite'. The infinite baffle is an especially good basis for theoretical analysis. But in design, too, it is usually logical to start out from this type of radiation environment, since sound diffraction and reflections appear rather as fluctuations and the curves of the infinite baffle by and large represent a good 'average'.

When the baffle amplifies, it does not conjure up energy from nowhere, but one half of the room does not receive radiation. If you imagine a large sphere around the sound source, then a point radiator radiates into all quadrants of the sphere. This is characterized as a  $4\pi$ -sr room. If a baffle divides the sphere into two equal halves and the point radiator is located on the baffle, then there is twice the sound pressure in one half and silence in the other half. This radiation environment is called a  $2\pi$ -sr room (steradians).

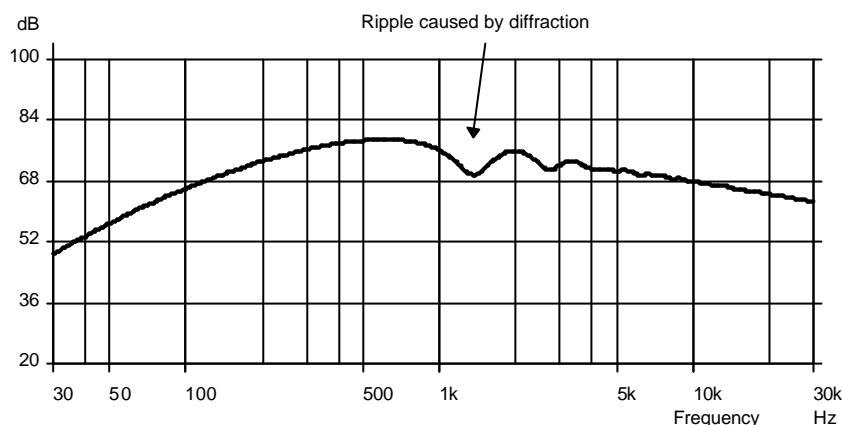


Fig. 17 Flat diaphragm in a circular finite baffle

With increasing frequency, the sound diffraction at the edges of a finite baffle influences the transition from full room to the half room. If the sound source is located on the acoustically hard surface of a sphere, a gently rising sound-level curve is obtained. The 3dB point is at the frequency at which the wavelength is equal to the length of the equator of the sphere (diffraction frequency).

If the baffle is flat and all edge points are equidistant from the radiator, there is first a strong accentuation compared with the sound level in the  $4\pi$ -sr room (+9dB). With increasing frequency, when the wavelength is equal to the distance from the edge, a trough appears in the sound-level curve. The ripple propagates and is damped mainly by the radiation characteristics of the radiator. Fig. 17 shows the typical curve. The strong increase in the frequency range between 1 kHz and 2 kHz is dampened by the effect of the voice-coil induction  $L_e$ .

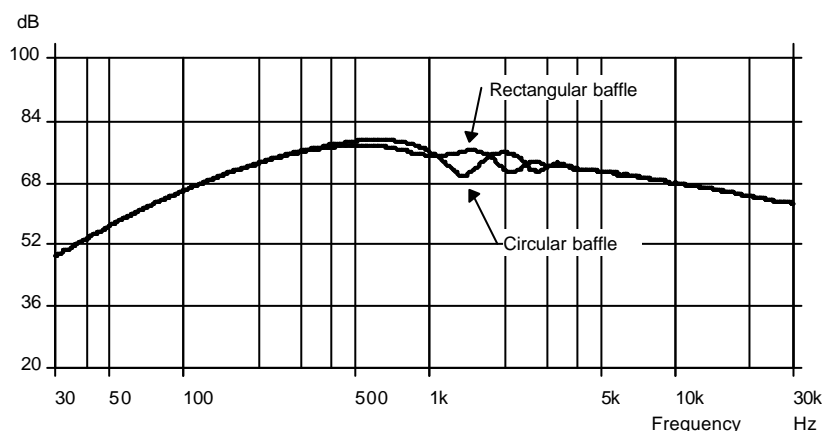


Fig. 18 Flat diaphragm in a rectangular baffle in comparison to the circular baffle

A circular baffle is rare. The baffle is usually rectangular and the radiator is asymmetrical with respect to the edges. The peaks and troughs in the sound pressure curve are then not so strongly pronounced. The region with ripple, however, is distributed over a wider frequency range (Fig. 18).

The sound diffraction is caused by the pressure loss behind the edge. Each pressure loss generates a 'radiator' whose 'radiation direction' is characterized by the direction of the pressure gradient. At the measurement point, the newly created 'radiators' behind the edges of the baffle interfere with the sound radiated directly. Some of the energy travels back to the diaphragm, and there interacts with the diaphragm movement. The rest of the acoustic energy is radiated from the reverse side of the baffle. With increasing frequency, a sound shadow appears here, so that at very high frequencies, the sound is only radiated into the front half of the room. The intensity is then twice as high. The complete description of the sound diffraction is extremely complicated. The influence of the sound diffraction depends not only on the distance from the edges to the radiator, but also on the angle between the edge and the listening point, and also on the texture of the edges.

## Reflector

For reflectors, the same applies as for the baffle, with the difference that the radiator is not in the same plane as the wall. The difference in travel time between the reflected and directly emitted sound wave causes interferences (Fig. 19). There are also diffraction effects at the edges of the reflectors.

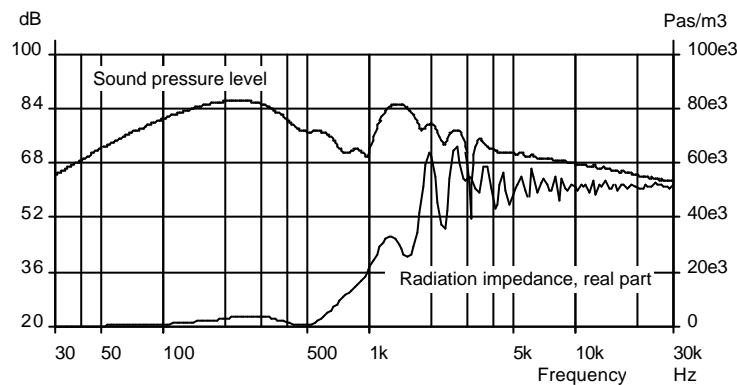


Fig. 19 Sound pressure curve and radiation resistance, including reflections from three walls

A large variety of reflectors can be distinguished, each with very distinctive characteristics. Many of them belong to the field of room acoustics. A criterion is introduced here that helps to distinguish whether a reflector belongs more to the room or more to the sound source. This criterion is known as the diffuse-field distance [Zwi]. The diffuse-field distance is the distance from the sound source at which the sound level in the free field is equal to the average sound level in the room. Reflectors located outside the diffuse-field distance belong to room acoustics. Those within it are assigned to the radiator.

If the sound wavelength is much smaller than the dimensions of the reflector wall, the laws of geometrical optics can be applied.

To gain an idea of this, imagine the reflectors as a mirror and the sound source as a candle. If you place the candle in a corner of a room with three mirrors perpendicular to one another, you see a total of eight candles. The same is true in acoustics. If, for example, you place a loudspeaker in a corner of a room, at very low frequencies the measured sound level is 18dB greater than if the speaker radiates freely. Since the speaker is usually located in a closed room, the standing waves of the room have to be additionally superimposed. With increasing frequency, the waves radiated directly and reflected interfere with one another and the value sinks to the level of free radiation. The process has superimposed on it the directivity of the baffle and the diaphragm.

An important dimension in acoustics is the level of the acoustic output power. To calculate this, the sound pressure over an envelope is integrated. For the practical calculation, this is only integrated over the room into which radiation takes place. A wall doubles the acoustic power in the rest of the room. A second wall, perpendicular to the first, quadruples the power. A room corner raises the level by 9dB. The values naturally only apply to acoustically hard, infinitely large surfaces and at very low frequencies.



## Radiation in AkAbak

AkAbak cannot calculate all the above-described characteristics of radiating loudspeakers. Because of the complicated relationships, the simulation concentrates on those characteristics that have the most significant and fundamental effects. The models used are in all cases based on physical principles, so that the effects on the output are linked to the particular parameter.

The missing characteristics may be manifested by, for example, too low or too high a level of the amplitude response of the simulation. It is also possible that the measured directivity diagram appears broader than that simulated. Many of the causes, but not all, can be attributed to natural vibrations. Diaphragm shapes and materials also play a large part. For the sound refraction at the edge, AkAbak shows the general trend. Additional cavities, the edge shape, etc. are not taken into account. Reflections are usually exaggerated in the simulation, even when absorption is specified, since room modes and scattering effects are usually present.

AkAbak calculates radiation from diaphragms and refraction edges in the far field. 'Far field' means that the distance from the diaphragm is much larger than the dimensions of the diaphragm. In this case the subdiaphragm can be regarded as equidistant from the listening point, so that the interferences that otherwise occur are negligible.

By contrast, the sum of all radiators in the near field is calculated (even if the listening point is very remote). This indicates that the travel times of the sound waves of the individual radiators depend on the listening point of all three coordinates.

## Using the parameter

The parameter that affect the radiation are the diaphragm shape, the mounting position, the baffle dimension and the reflectors.

The parameter of the diaphragm are obtained by measuring its geometrical form. It is best to determine the driver parameter using the program item 'Tools/Dyn. Driver parameter' or 'Tools/Piezo Driver parameter'.

When using these parameter, it is in many cases useful to get used to thinking in terms of a black box. With a black box, one only knows the input and output parameter but not what is in the black box. In the simulation of drivers with acoustic radiation, the black box can be equated with the radiation. Although we have a mathematical model of the radiation, we know that it does not extend to all the conditions.

If, for example, a driver is used in a large system which it is important to configure, the prerequisite is that the driver, with its radiation, has the correct transmission even though one parameter or another may not correspond to the correct geometrical dimensions.

In another example, certain properties of a driver are modified. In this case, too, the prerequisite is that the transmission of the driver is correctly simulated. It is easy to estimate the parameter because of their linear characteristic.

If the parameter of a driver are determined for the simulation, their values should be set so that both the input and output parameter of the simulation correspond to the measured data. It is of secondary importance that the parameter values correspond exactly to the actual geometry.

For example: We want to analyze a conventional loudspeaker with a conical diaphragm in a large system with filters, enclosures etc. First the curves of the driving point impedance and the sound pressure level are recorded under test conditions at various listening angles. We determine the motor-parameter and the diaphragm dimensions. In the second stage, we simulate this loudspeaker under the test conditions and vary the parameter until we have found an optimum agreement of **all** curves. By this post-processing we can then assume that, if the curve of the electrical driving point impedance and of the acoustic output have been well simulated, that the driver will also behave correctly when combined with other electrical and acoustic elements.

It is incorrect, for example, to compensate a level drop at higher frequencies by reducing the value of the voice-coil inductance  $L_e$ . This modifies the input impedance of the driver, which in turn has consequences for the rest of the network. It is better in this case to reduce either the mass reduction frequency  $f_p$  or the cone depth  $td_1$ .

## Diaphragm Parameter

Since the acoustic modeling is in many cases done with the help of one-dimensional waveguides the specification of cross sections is common practice in AkAbak. Therefore a large subset of parameters is available which are called diaphragm parameter. Diaphragms are also the means of radiation which are responsible for the mechano-acoustical coupling. There are radiating cross sections and loudspeaker diaphragms. The first category is used for radiating holes, vents and waveguides. The latter copes the more sophisticated mechanics of loudspeaker cones and domes. The specification of the diaphragm parameter is usually given at the actual component to which an additional Radiator element refers which is responsible for the radiation into free space.

### Cross-sectional areas

Cross-sectional areas are flat, piston-like vibrating circular or rectangular radiation surfaces. The cross-sectional areas have no function for mass reduction or recessing into the baffle. Holes in the diaphragm area are also not taken into account here (except the `Coupler`). The reference point for mounting is in the center of the cross-sectional area.

#### Duct, Enclosure (vent)

<code>dD= . . . m</code>		diameter of the duct cross-section. Unit: meter [m].
<code>SD= . . . m<sup>2</sup></code>	(alternative)	duct cross-sectional area. Unit: square meter [m <sup>2</sup> ].
<code>WD= . . . m</code>	(alternative)	width and
<code>HD= . . . m</code>		height of the rectangular duct. Unit: meter [m].

#### Horn, Waveguide

<code>dTh= . . . m</code>		diameter of horn throat.
<code>STh= . . . m<sup>2</sup></code>	(alternative)	cross-sectional area of horn throat.
<code>WTh= . . . m</code>	(alternative)	width and
<code>HTh= . . . m</code>		height of horn throat with rectangular cross-sectional area
<code>dMo= . . . m</code>		diameter of horn mouth.
<code>SMo= . . . m<sup>2</sup></code>	(alternative)	cross-sectional area of horn mouth
<code>WMo= . . . m</code>	(alternative)	width and
<code>HMo= . . . m</code>		height of horn mouth with rectangular cross-sectional area

#### Coupler

(if used as diaphragm or part of it)

<code>dD= . . . m</code>		diameter of cross-section.
<code>SD= . . . m<sup>2</sup></code>	(alternative)	cross-sectional area.
<code>WD= . . . m</code>	(alternative)	width and
<code>HD= . . . m</code>		height of the rectangular cross section.
<code>dD1= . . . m</code>		diameter of a circular hole in the middle.

$SD1 = \dots m2$  (alternative) cross-sectional area of a circular hole in the middle.

## Loudspeaker diaphragm

The loudspeaker diaphragm is regarded as an inherently rigid diaphragm that vibrates in the manner of a piston. The outer shape may be rectangular or circular. With flat or conical diaphragms, a circular hole may be cut in the center ( $dD1 < 0$ ). The diaphragm area of conical and dome-shaped diaphragms is frequency dependent if the parameter  $f_p =$  has been entered. Eigen-vibration of the diaphragm causes an additional sound field with a complex directivity pattern which appears at high frequencies more or less diffuse. Since in some cases the directivity is broadened by this radiation. Use than the  $Diffuse =$  parameter to control the directivity calculation.

The two parameter  $tD1 =$  and  $t1 =$  indicate the depth of a cavity. This provides the eigenfrequencies and interferences that are apparent in the sound pressure level curve. In the upper frequency band, the simulation curve of the sound pressure level has a comb-filter shape. The greater the values of  $tD1 =$  and  $t1 =$ , the lower the frequencies with these troughs. In the measurement curve for the sound level, at least the first trough can be clearly identified, despite strong natural vibrations of the diaphragm in this frequency range.  $tD1 =$  and  $t1 =$  should then be post-processed until the first trough in the simulation curve is congruent with that in the measurement curve.

With circular conical and domed diaphragms, calculation is done by numerical integration across the diaphragm surface. With rectangular diaphragms, the program weights the directivity characteristic of the rectangular diaphragm (si-function) with the integral of the radiation on-axis over the cone depth. This last-mentioned method is not quite as precise, but nevertheless provides a good approximation to the directivity characteristic of this fairly rare diaphragm shape (rectangular cone).

With flat diaphragms, the program calculates the directivity characteristics by means of the Bessel series (circular) or the si-function (rectangular).

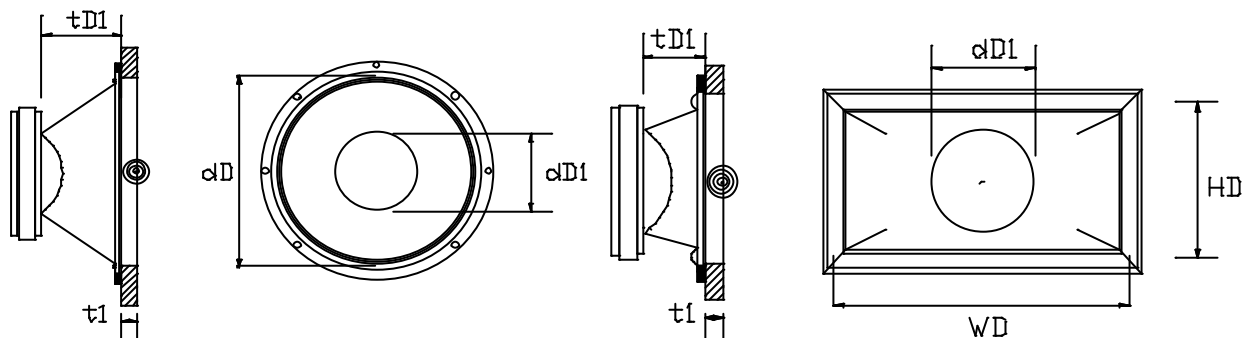


Fig. 20 Circular and rectangular diaphragm (The dot indicates the reference point for mounting)

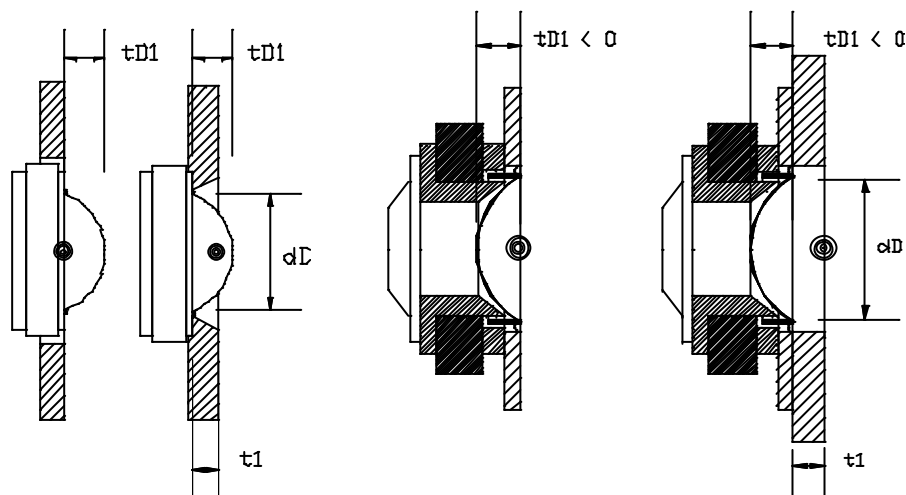


Fig. 21 Convex and concave diaphragm

$dD = \dots m$		<p>External diameter of the circular diaphragm. <math>dD</math> is measured from the center-line of the suspension. If the loudspeaker does not have a suspension, but the diaphragm is clamped as with capacitor microphones or many tweeters or the like, the area equivalent to the diaphragm vibrating in a piston-like manner should be used.</p> <p>Unit: meter [m].</p>
$SD = \dots m^2$	(alternative)	<p>Cross-sectional area of the circular diaphragm. The program internally converts <math>SD</math> to the external diameter <math>dD</math>.</p> <p>Unit square meter [<math>m^2</math>].</p>
$WD = \dots m$ $HD = \dots m$	(alternative)	<p>Width and height of the outer diaphragm. If <math>WD</math> and <math>HD</math> are entered, the program interprets the outer diaphragm shape as rectangular. <math>WD</math> and <math>HD</math> are measured from the center of the diaphragm suspension.</p> <p>Unit meter [m].</p>
$dD1 = \dots m$		<p>Diameter:</p> <ul style="list-style-type: none"> <li>- inner diaphragm (<math>dD1 &gt; 0</math>) or</li> <li>- hole (<math>dD1 &lt; 0</math>).</li> </ul> <p>With flat diaphragms:</p> <ul style="list-style-type: none"> <li>- only hole (<math>dD1 &lt; 0</math>)</li> </ul> <p>With cones:</p> <ul style="list-style-type: none"> <li>- dust cap (<math>dD1 &gt; 0</math>)</li> <li>- hole (<math>dD1 &lt; 0</math>)</li> </ul> <p>With domes:</p> <ul style="list-style-type: none"> <li>- is characterized by <math>dD1=0</math></li> <li>or <math>dD1</math> not entered.</li> </ul> <p>The range of the parameter <math>dD1</math> therefore controls important characteristics of the diaphragm shape.</p> <p>If <math>dD1=0</math>, the diaphragm is dome-shaped.</p> <p>If <math>dD1&gt;0</math>, the diaphragm cone has a dust cap. For flat diaphragms, it has no meaning to enter <math>dD1&gt;0</math>, and an error message is given.</p> <p>If <math>dD1&lt;0</math>, in the calculation of the radiation, radiation impedance etc. for flat and conical diaphragms, a round hole of diameter <math>dD1</math> is formed in the center or at the cone tip.</p> <p>The outer diaphragm shape is immaterial.</p> <p>The inner surface of <math>dD1</math> is always circular.</p> <p>Unit meter [m].</p>
$SD1 = \dots m^2$	(alternative)	<p>Cross-sectional area of the inner diaphragm.</p> <p>Internally, the program converts <math>SD1</math> to the diameter <math>dD1</math>. The same applies as described for <math>dD1</math>.</p> <p>Unit: square meter [<math>m^2</math>].</p>
$tD1 = \dots m$		<p>Cone:</p> <p>depth of the cone. Measured from the suspension to the dust cap.</p> <p>Concave dome:</p> <p>depth of the dome sphere measured from the suspension to the lowest point.</p> <p>Convex dome:</p> <p>height of the dome sphere measured from the suspension to the tip.</p> <p>Flat diaphragm:</p> <p>characterized by <math>tD1=0</math></p> <p>or <math>tD1</math> not entered.</p>

Note, that the response is very sensitive to  $tD1$ . You clearly can recognize the effect to the SPL at high frequencies. In most cases the value read from geometrical measure is a bit to large.

Unit meter [m].

$t1 = \dots m$

Diaphragm recessed in the baffle.

$t1$  is measured from the diaphragm suspension to the surface of the baffle.

For convex domes, the dome may well project above the baffle surface.

Transmission is very sensitive to this parameter. If the vent is not cylindrical but, for example, flared, the value of  $t1$  should be adapted.

$t1$  should be less than the diaphragm radius. Otherwise the acoustic structure modeled with the aid of the element `Duct`.

The parameter  $t1$  may be entered at two points in the script. Firstly, as here, in the diaphragm dimensions and secondly in the position data of the radiator. `AkAbak` adds the two dimensions. Together they must be smaller than the diaphragm radius.

This option is very useful, since the driver diaphragm itself is often already offset or provided with a small horn.

In addition, offset installation in the baffle can be taken into account without the `Duct` element having to be used.

Unit: meter [m]

$f_p = \dots Hz$

Frequency for controlling the mass or area reduction.

Not for flat diaphragms ( $tD1=0$ ).

The effective diaphragm diameter of conical diaphragms is at  $f_p$ :

$dD(f_p) = (dD + dD1)/2$  plus the inner diaphragm area, if present. The inner diaphragm is not reduced.

When calculating dome-shaped diaphragms, the program calculates in the diaphragm center a hole that increases in size with frequency. At  $f_p$ , the diameter of this hole is 2/5 of the outer diameter.

At frequencies above  $f_p$ , the diaphragm area decreases constantly. At frequencies below  $f_p$ , the effective area rapidly converges to that entered.

The reduction of the radiation area is followed by the mass reduction. The driver parameter are changed correspondingly. The parameter  $f_p$  therefore acts even if the diaphragm does not radiate directly but into an acoustic structure, such as into a duct (`duct`) or into a horn (`Horn`, `Waveguide`).  $f_p$  is an integral parameter that is used to express several different mechanical and acoustical diaphragm characteristics.

This is generally successful. Unfortunately, it is impossible to find a measuring method for  $f_p$ . A rough rule of thumb is that the frequency  $f_p$  is equal to the frequency range in which the first partial vibrations occur. A good starting value for post-processing is twice the directivity frequency which is approximately  $220/dD$ .

If  $f_p=0$  or if  $f_p$  has not been entered, the mass reduction is disabled.

Unit: hertz [Hz].

`Diffuse = ...%`

At high frequencies the directivity pattern is in many cases wider in range than calculated by the program. To have the possibility to control this effect to some degree a diffuse factor diminishes the effect of the calculated directivity pattern calculation. The range of controlling is:

`Diffuse=0%` Directivity pattern as calculated

....

`Diffuse=100%` No Directivity, radiates like a point source.

## Reference point for mounting

The reference point for mounting is with all loudspeaker diaphragms in the center, at the level of the suspension of the diaphragm.

If the diaphragm is recessed in the baffle ( $t_1 > 0$ ), the mounting point is in the center of the outer cross-sectional area (marked by the dot in Fig. 20 and Fig. 21).

Entries for loudspeaker diaphragms can be found in the definitions:

```
Def_Driver, Def_TwoCoilsDriver, Def_PiezoDriver,
Def_Speaker, Def_BassUnit
```

and in the elements:

```
Radiator, Diaphragm.
```

The acoustic network element `Radiator` is only a radiator, without mechanical elements, but with acoustical impedance. It takes the parameter for the loudspeaker from the associated driver or acoustic element. If no reference (`Def=`) has been entered, the parameter of the loudspeaker diaphragm can also be used for the `Radiator` element.

Another network element that accepts the parameter for the loudspeaker diaphragm is the `Diaphragm` element.

The `Coupler` element can be used to connect the mechanical domain to the acoustical domain. Because of its importance you can assign diaphragm parameter directly. Instead to use entries as, for example: `Ratio={SD=pi*sqr(0.15); 1/SD }`, you would specify: `dD=30cm`. To be able to specify also ring-shaped diaphragm, it is possible here to specify the parameter `dD1=` or `SD1=`.

## Positioning of Radiators

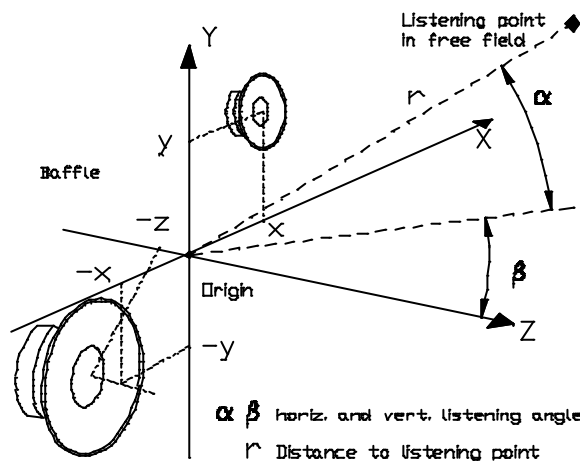


Fig. 22 Baffle coordinate system

The program calculates current, voltage, sound pressure and volume velocity in the networks of the individual systems. To be able to determine the sound pressure at a specific point in the listening room, it is necessary to have position data.

At this listening point, the sound pressure of the individual radiators are summated according to modulus and phase. The sound pressure at the listening point is determined by the distance from the radiator as well as by the radiation characteristics of the particular radiator. This results in a mixed near-field and far-field analysis. Far-field reproduction is used as basis for the individual radiators and near-field analysis for the entire system of existing radiators.

The distance and listening angle from the origin of the baffle coordinate system are entered in the control dialog of the simulation (menu: Sum/...). As shown in Fig. 22, the distance  $r$  relates to the origin of the baffle coordinate system.

The vertical listening angle  $\alpha$  relates to the xz plane of this coordinate system. By contrast with mathematical convention, angles greater than zero are upwards and angles greater than zero downwards.

The horizontal listening angle  $\beta$  relates to the yz plane. It is positive for angles towards the positive x axis and negative in the other direction.

## Parameter of the mounting position

The reference point of the radiation, to which these coordinates relate, is similar for all elements. With loudspeaker diaphragms, the reference point is always in the center of the diaphragm surface at the height of the suspension - irrespective of whether the loudspeaker is of cone or dome type. If the radiator is the aperture of an acoustic duct (Duct or vent), the reference point is in the center of the cross-sectional area. With horns, it is in the center of the horn mouth flange (see Horn).

If the parameter  $\tau_1$  for the recessing is entered, the reference point for mounting is in the center of the outer cross-sectional area of the vent thus produced.

With conical diaphragms, the acoustic center is displaced towards the inner diaphragm with frequency. That means that at very high frequency virtually only the diaphragm, which is offset slightly backwards, radiates sound. The displacement of the acoustic center with frequency is calculated by AkAbak.

$x = \dots m, y = \dots m, z = \dots m$

For each radiator, the position data x, y and z in the baffle coordinate system are entered. If the radiator is left of the origin, the x-coordinate is positive, otherwise negative. If it above, then y is greater than zero, if it is below, then the value of y is less than zero. Positive values of the z coordinates place the radiator in front of the baffle, if they are less than zero, then it is behind the baffle.

If one of the parameter x, y, or z is not given in the script for a radiator, its value is set to zero by default.

Unit: meter [m] or inch [in].

$HAngle = \dots^\circ, VAngle = \dots^\circ$

In addition to the position data x, y, z, the mounting angles, HAngle and VAngle can also be entered. They are not drawn in Fig. 22.

HAngle and VAngle rotate the axis of the aperture about the reference point of mounting. This axis is perpendicular to the radiating area and passes through the reference point of the radiator.

HAngle describes the rotation of this axis in the horizontal (xz) plane. The value of HAngle varies between  $-180^\circ$  and  $+180^\circ$ . Values greater than zero rotate the radiator towards the positive x axis. If, for example, a loudspeaker is located on the left-hand side of a rectangular enclosure, then you should enter  $HAngle = 90^\circ$ . If it is on the rear side of the speaker, then enter

$HAngle = 180^\circ$ .

VAngle rotates the radiate in the vertical (xy) plane. In this case its range is limited to  $-90^\circ \leq VAngle \leq +90^\circ$ . Positives values rotate the radiator upwards, negative values downwards. Although from a mathematical point of view this direction is the wrong way round, it is more intuitive: positive upwards, negative downwards. If, for example, a loudspeaker in a rectangular enclosure radiates towards the base - it is therefore on the underside - then

$VAngle = -90^\circ$

With the angles HAngle and VAngle, the radiator can be rotated in any direction. If one of the parameter HAngle or VAngle is not entered, then AkAbak sets its value to zero degrees by default.

Unit: degree [ $^\circ$ , Deg]

$\tau_1 = \dots m$

Recessing of a loudspeaker diaphragm in the baffle.  $\tau_1$  is measured from the diaphragm suspension to the baffle surface. With convex domes, the dome may well project above the baffle surface.

The transmission responds very sensitively to this parameter. It is particularly

when the vent is not tubular that the value of  $\tau_{1=}$  should be modified..

$\tau_{1=}$  should be smaller than the diaphragm radius. Otherwise the acoustic structure should be modeled by means of the `Duct` element.

The parameter  $\tau_{1=}$  can be entered at two points in the script. First here, as with the positional data and secondly at the diaphragm dimensions of the radiator. AkAbak adds the two dimensions together. Their sum must be smaller than the diaphragm radius. This option is very useful, since the driver diaphragm is often already offset or provided with a small horn. In addition, the offset installation in the baffle can be taken into account with the `Duct` element having to be used.

Unit: meter [m]

## Shifting of the origin

The origin of the baffle coordinate system can also be shifted retroactively without changing the individual position data. To do this insert the definition `Def_ListeningPoint` at the start of the script (see chapter Def).

### ☒ Example

The form for entering the data is best illustrated using an example (Fig. 23). In this case the origin of the baffle coordinate system is at the upper rim of the medium-range speaker, which is installed set back by 4cm. The two tweeters are turned outwards slightly. A bass reflex vent ends at the head of the enclosure.



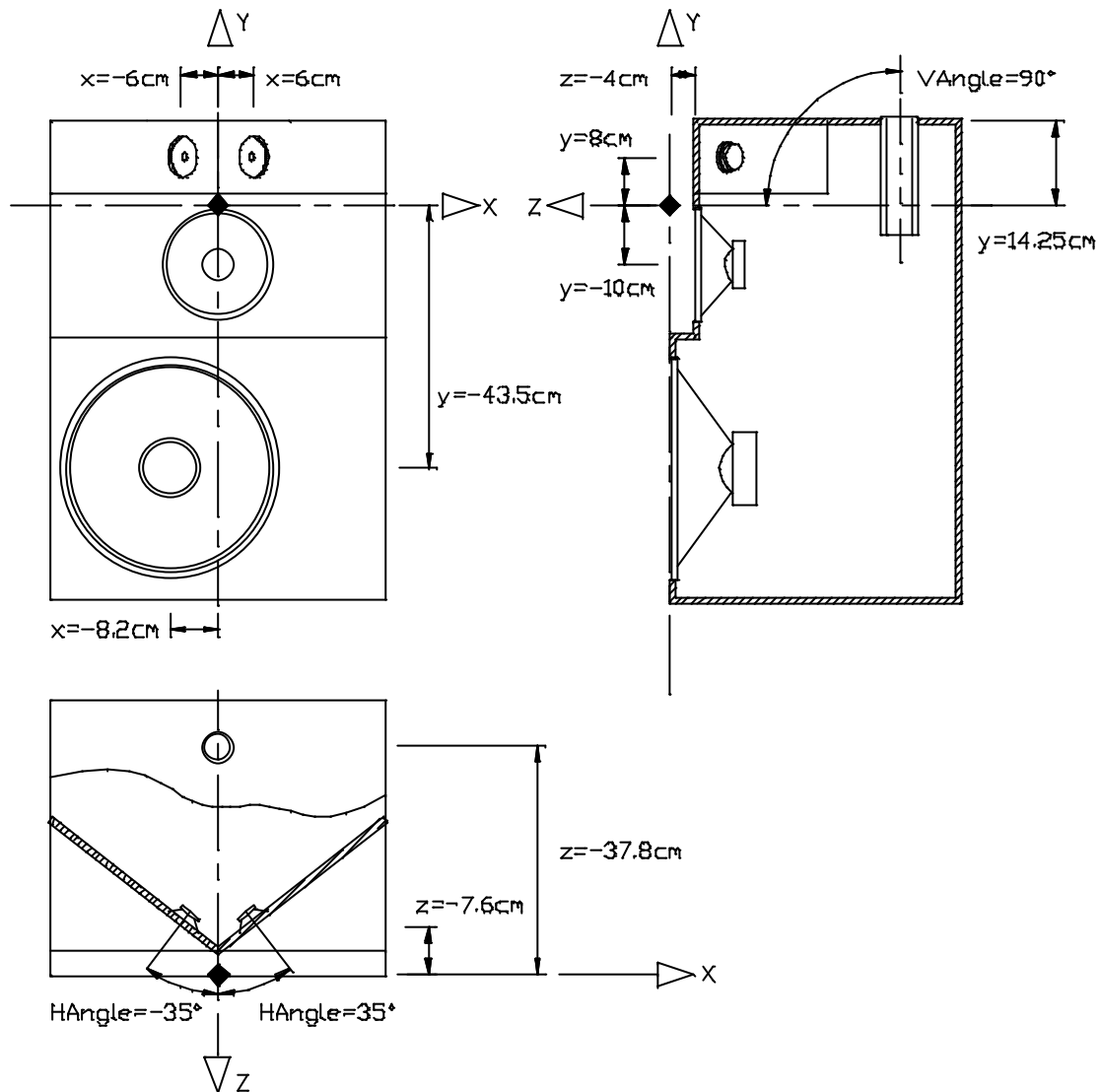


Fig. 23 Example of positioning

```

...
Radiator 'bass' ...   x=-8.2cm y=-43.5cm z=0      HAngle=0    VAngle=0
Radiator 'Mid' ...   x=0      y=-10cm  z=-4cm    HAngle=0    VAngle=0
Speaker 'high right' x=6cm    y=8cm    z=-7.6cm  HAngle=35°  VAngle=0
Speaker 'high left'  x=-6cm   y=8cm    z=-7.6cm  HAngle=-35° VAngle=0
Enclosure 'vent' ... x=0      y=14.25cm z=-37.8cm HAngle=0    VAngle=90°

```

(This example omits some parameter for the sake of clarity.)

## Radiation Environment

It is particularly simple to simulate the radiation if the radiator is located in a surface which is very large compared with the sound wavelength. This surface reflects the sound completely, is inherently rigid, and has no obstructions. Such a radiation environment is called an 'infinite baffle' or a ' $2\pi$ -sr room' (sr: steradians). The sound pressure of such an embedded sound source is twice as high as so-called 'free radiation' or radiation in the ' $4\pi$ -sr room'<sup>1</sup>.

<sup>1</sup>The expressions ' $2\pi$ -sr room' and ' $4\pi$ -sr room' originally come from the calculation of acoustic power, in which the sound pressures are summated on an envelope surface around the source. If the radiator is mounted in an infinite baffle, the sound

In practice, of course, this case is rarely found. Usually the transition from free radiation conditions ( $4\pi$ -sr room) to the acoustic hemisphere ( $2\pi$ -sr room) takes place in the middle of the transmission band of a loudspeaker. The sound pressure curve is, on average, 6dB lower at low frequencies than at high frequencies, at the level of the sound power it is 3dB. In the frequency band between these, the level increases with more or less ripple. The ripple is least when the radiator is located in a spherical housing.

This effect is called sound diffraction. The mathematical description of the sound diffraction is extremely complex, especially in the case of angular housings. The program approximates the sound diffraction in a manner that, by and large, illustrates the effect within acceptable computation times.

Diffraction at the baffle edges is calculated in the far field and affects the radiation and the radiation impedance. The diffraction calculation uses the mirror radiator model. Imagine a radiator in the center of a circular baffle. After the wave has traveled across the baffle and reached the edge, a pressure gradient is produced beyond the baffle. It is possible, to some degree of approximation, to lump this outer 'diaphragm' into a mirror radiator which is radiating the same signal but time delayed. The mirror radiator is fed by the original radiator under a 'listening angle' of  $90^\circ$  to take into account the directivity of the 'source-radiator'.

With the specification of diffraction edges the loudspeaker radiates omni-directional at very low frequencies, i.e. when the wavelength is much longer than the dimensions of the baffle. If the wavelength becomes comparable to the dimensions of the baffle interference happens and the transmission shows rippling. At very high frequencies the directivity of the original radiator suppresses the diffraction effects and no sound energy is radiated to the rearward space.

In AkAbak, a radiator in the center of a circular baffle is the fundamental model. The parameter of the circular baffle is `dEdge=` which is the diameter of the baffle. In the range of simulation accuracy of the program this model on the one hand gives good results and on the other hand a fast calculation speed.

When the baffle is rectangular and/or the radiator is mounted asymmetrically then two mirror radiators are installed, each standing for the distance to one of the two edges. The parameters of the rectangular baffle are `WEdge=` and `HEdge=` which are the width and height of the baffle. The asymmetry leads to a smearing effect in the transition frequency range. Please keep in mind that the model is an approximation. The specification of `WEdge` and `HEdge` should be more or less estimated by using mean values.

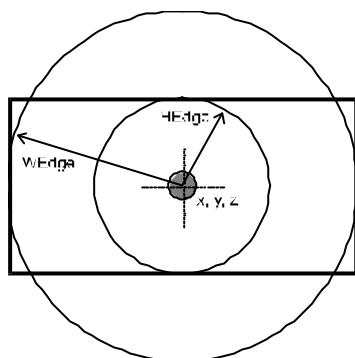
The diffracting baffle is attached to the mounting position of the radiator i.e. there are no absolute position parameter for the baffle. When the position of a radiator is modified the diffraction parameter may be also adjust.

## Parameter of diffraction

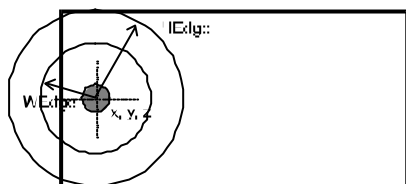
<code>WEdge= . . . m</code>	Width and.
<code>HEdge= . . . m</code>	height of the rectangular baffle. The radiator is assumed to be located in the center of the baffle. If the baffle has a different outline shape or is installed asymmetrically, use average values or adjust them accordingly. Unit: meter [m]
<code>dEdge= . . . m</code> (alternative)	Diameter of a circular baffle. The radiator is assumed to be located in the center of the baffle. Unit meter [m].

### ☒ Examples:

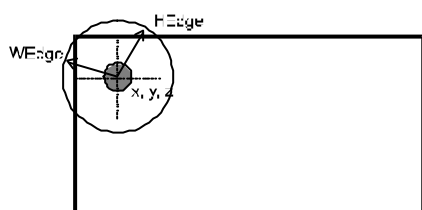
Take a baffle of dimensions 30cm x 60cm



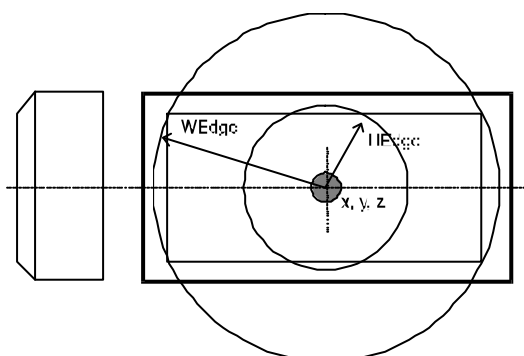
The radiator is mounted in the center  $W_{Edge}=30\text{cm}$   $H_{Edge}=60\text{cm}$



The radiator is mounted 10cm from the edge  $W_{Edge}=20\text{cm}$   $H_{Edge}=60\text{cm}$



The radiator is mounted in a corner 10cm from the edge  $W_{Edge}=20\text{cm}$   $H_{Edge}=20\text{cm}$



The radiator is mounted in the center but the edges are bent  $W_{Edge}=40\text{cm}$   $H_{Edge}=70\text{cm}$

As this examples shall demonstrate, in most cases an appropriate estimation has to be applied.

## Default setting

If none of these parameter has been entered or their values have been set to zero, baffle conditions are assumed for all frequencies (infinite baffle). There is no radiation calculated on the rearward side of the baffle.

## Sound diffraction and reflectors

The parameter for sound diffraction are usually optional. However, there is an exception, namely when the radiation element participates in room reflections. In this case the radiator must not be located in or behind the reflecting wall. It is thus mounted not in an infinite baffle but in a finite baffle with diffraction effects. Whenever the `Reflection` keyword has been entered, therefore, one of the aforementioned parameter must also be present.

## The baffle is as big as the radiation surface

If the baffle is so small that there is just enough space for the radiator, you can 'fool' the program by entering an extremely small value for  $W_{Edge}$ ,  $H_{Edge}$  or  $d_{Edge}$ , for example:  $d_{Edge}=1\text{mm}$ . AkAbak tests whether the distance from the baffle edge is greater than the diaphragm radius. If it is smaller, the parameter for the edges

are replaced by the dimensions of the diaphragm. In this case, the baffle is equal to the diaphragm area. Such a configuration is known as 'radiation at the end of a long tube'.

☒ Example:

```
Radiator  Def='B1'
Node=2
x=0  y=0  z=0  HAngle=0  VAngle=0
WEdge=50cm  HEdge=30cm
```

## Parameter Reflection

Reflection

Keyword in the description of a network element with radiation. If `Reflection` is entered, the particular radiator participates in the reflection. The prerequisite is that a definition `Def_Reflector` exists and is not switched off by `Off`. A further condition is that the radiator is not located in an infinite baffle, i.e. one of the sound diffraction parameter (`WEdge=`, `HEdge=` or `dEdge=`) must be entered.

☒ Example:

```
Radiator  Def='B1'
Node=2
x=0  y=0  z=0  HAngle=0  VAngle=0
WEdge=50cm  HEdge=30cm  Reflection
```

The position of the loudspeaker with respect to the reflectors is given in the `Def_Reflector` definition (see Chapter 'Def'). When the keyword `Reflection` is entered, the routines for calculating the reflection of the particular radiator is switched on. When the `Def_Reflector` definition is present, the `Reflection` keyword does not necessarily have to be entered for each radiator.

If `Def_Reflector` is switched off with `Off`, or is not present, the entry for `Reflection` is ignored.

The program first calculates the position of the individual radiator with respect to the reflectors and then calculates the radiation resistance and the radiation.

The radiation reactance is not affected by the reflectors in the program, since in this case the interaction is much smaller than for the resistance. Eigenfrequencies of the listening room can have a much greater effect. This type of room acoustics cannot yet be calculated by AkAbak.

## Parameter NoRad and NoDir

NoRad

In some cases it is desirable to be able to switch off the radiation, but to keep on working the radiation impedance.

NoDir

This keyword switches off the directivity-calculation of the diaphragm (Point source). Diffraction and reflection effects are still working.

☒ Example:

```
Radiator  Def='B1'  Node=2
NoRad
```

```

Radiator  Def='B1'
Node=2
x=0  y=0  z=0  HAngle=0  VAngle=0
WEdge=50cm  HEdge=30cm  Reflection
NoDir

```

## Parameter Label=

Labels can control the summation of the sound pressure or sound intensity vectors. For example, when you want to distinguish the total radiation of a vented enclosure and the radiation of the port only. Or, the radiation of a tweeter, the bass channel and the total sum.

Labels must be first specified in the script and can then be selected in the simulation control dialog (Fig. 24). System names are also valid labels.

The output is then the sum of all radiators with the same label.

If no label is selected the total sum of all radiators is displayed (<all>).

### Specifying labels

Enter a list of labels in that particular radiator declaration which shall be sensitive to the labeling. The syntax of a label is:

```
Label=x1=x2=x3=...
```

x1, x2,... are numbers out of the set [1...200].

#### ☒ Example

```

System  'Bass'

Radiator  Def='D1'  Node=2
x=0  y=0  z=0  HAngle=0  VAngle=0
WEdge=20cm  HEdge=40cm  Label=10
...

Enclosure  'E1'  Node=200
Vb=30L  Lb=20cm
fb=50Hz  dD=10cm
x=0  y=-20cm  z=0  HAngle=0  VAngle=0
WEdge=20cm  HEdge=40cm  Label=10=20

```

This example could be part of a loudspeaker system with a vented enclosure. The Radiator is the implementation of the diaphragm radiation and Enclosure has a radiating vent. Label '10' is part of both elements and label '20' is part only of the Enclosure element. In the simulation the sum-sound pressure or power of both can be displayed as well as the radiation of the vent only.

### System-names as labels

When a System is assigned a name then this name is also a label. When this name is selected in the simulation control dialog all radiators of this System are selected and summed up. This feature is practical in cases you want to investigate the radiation of, for example, a tweeter and a bass-system separately.

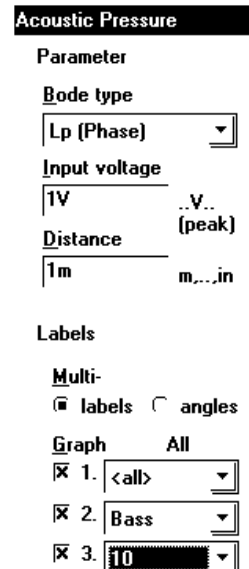


Fig. 24 Control dialog of simulation. Select here the labels you have specified in the script

## Radiation Position - Dialog

The entries for positioning and the radiation environment may be edited using the modal 'Radiation position' dialog.

If you activate the 'Def/Radiation Position...' menu command, the input boxes are filled with the position data of the element in which the script cursor was located before the dialog was called up.

For details press the **F1**-key when the dialog is active.

By clicking on the 'OK' button, you close the dialog and the data are stored, in formatted form, in the clipboard. They can then be inserted anywhere in the script (Menu: Edit/Paste or **Ins** key).

The dialog 'Radiation position' is also part of the component dialogs and acts there as a sub-dialog. The data entered appear in a display of the dialog that called it up.

## Transducer

An electro-acoustic transducer is a network element. The driver converts the electrical power at the input of the element first into mechanical power and then into acoustic power. The electrical, mechanical and acoustic parts of the network are equivalent:

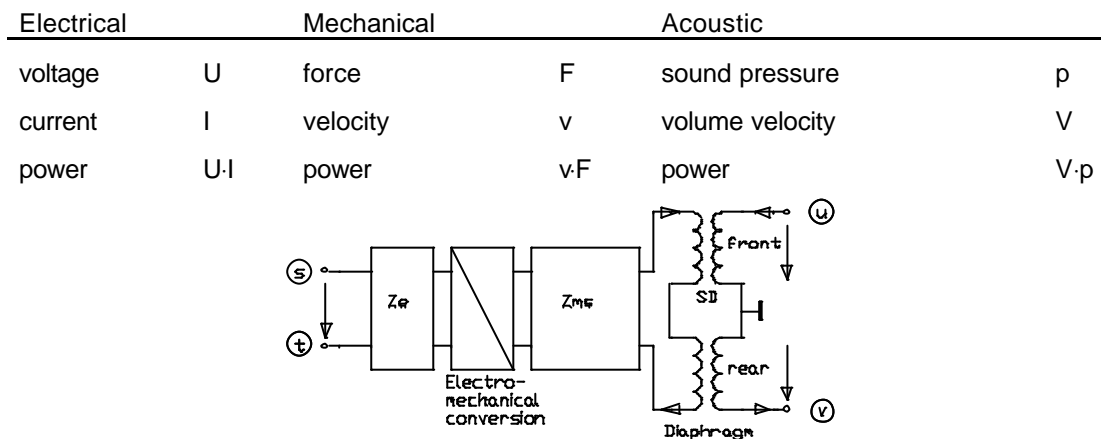


Fig. 25 General schematic of electro-acoustical transducer

Fig. 25 shows a general schematic of a transducer. It might be any driver: dynamical, piezo etc.

$Z_e$  is any electrical impedance as for instance the voice coil impedance of the dynamical driver or the piezo capacitance of the piezo driver. It follows the electro-mechanical converter or transducer-motor.  $Z_{ms}$  is the impedance of the vibrating mechanical assembly. The output stage is the diaphragm and eventually some acoustical elements. The two ideal transformer form the front and rear-side of the diaphragm. At node  $u$  all acoustical elements are wired which are connected with the front diaphragm. Node  $v$  is the reverse side of the diaphragm. More couplers are necessary if parts of the diaphragm radiate in different acoustical paths.

In most cases you will use the default standard driver-models of AkAbak. The input terminals are at the electrical side and the output poles at the acoustic side. That means all mechanical elements and the couplers are encapsulated within the transducer-model. Since all drivers possess a diaphragm, one output pole corresponds to the front side of the diaphragm and the other represents the rear side of the diaphragm.

In AkAbak the model and its parameter are designed in such a way that the 'exactness' is scaleable. It is possible to give only a handful common parameter for a rough evaluation. On the other hand when you specify all possible parameter, the most exact analysis can be obtained. Furthermore the basic parameter follow the common known syntax as given among others by Small [Sma]. Take as an example the voice coil inductance  $L_e$ . The lumped element  $L_e$  without modification leads to a wrong simulation in the upper frequency band. There is probably no speaker which follows the ' $Re + j\omega L_e$ ' model. But if your attention is focused just to the low frequency part this simple model may be sufficient and you can left apart the more 'difficult' parameter of the voice coil model.

It is also possible to create your own transducer models. Elements like the `Coupler`, `Impedance` and the mechanical elements are powerful means to form sophisticated electro-mechano-acoustic structures (see chapter `Net/Coupler` and exercises).

The radiation of microphones can only be simulated in reverse state. The enclosed acoustics of a receiver can be analyzed in the usual way. The input voltage  $U_{in}$  is then the input force or pressure, respectively.

## Dynamical driver

The dynamical motor is the most common for loudspeakers. Some fundamental principles of the model shall be given here. The most of it, excluding the electrical side, is also valid for any other transducer-principle.

### Electrical impedance

The electrical impedance is made first by the voice coil impedance and second by the gyrator transform of the mechanic-acoustical part  $Z_e = Bl^2/Z_m$ .

The voice coil impedance is usually non-linear, temperature dependent and with a complicate frequency response.

The non-linearity can not be solved directly by the current version of AkAbak. We have to introduce operating points and assume that the parameter are constant.

The temperature dependence can be implemented with the help of the formula assignment to parameter.

To describe the impedance frequency response, different models can be applied. Without going into details, we have decided to use transcendental functions to modify the basic model of the series connection of  $R_e$  and  $L_e$ .

#### Voice coil resistance $R_e(f)$

$R_e$  is the d.c. resistance of the voice coil and should be measured with a d.c.-meter.

As  $R_e$  increases, so the quality  $Q_{es}$  also increases. The attenuation of the oscillating circuit comprising  $M_{ms}$  and  $C_{ms}$  is thus reduced<sup>2</sup>.

In practice, further resistances are often in series with  $R_e$ , such as loss resistances in a cross over, cable losses, etc. They all increase the value of  $Q_{es}$  and thus affect the transmission characteristic at the resonance frequency.

$R_e$  increases with increasing voice coil temperature, so that the driver has a reduced efficiency and a different transmission behavior.

Eddy currents inside the pole-piece and other effects cause a rise of  $R_e$  at very high frequencies. The frequency  $f_{re}$  and the exponent  $ExpoRe$  can be specified in all dynamical driver models of AkAbak:

$$R_e(f) = R_e \cdot (1 + f / f_{re})^{ExpoRe}$$

Typically  $f_{re} = 2\text{kHz} \dots 20\text{kHz}$  and  $ExpoRe \approx 1$ . The value of  $f_{re}$  is much larger then the fundamental resonance of the driver:  $f_{re} \gg f_s$ . In many cases you can neglect these parameter. The tool 'Tools/Dyn. driver parameter' is able to determine  $f_{re}$  and  $ExpoLe$  from measurement.

#### Voice coil reactance $X_e(f)$

The equivalent circuit diagram for the voice coil impedance, which consists of the series connection of  $R_e$  and  $L_e$  is oversimplified for a full-range analysis. With most dynamic loudspeakers, the voice coil reactance  $X_e$  has a shallower graph at high frequencies ( $R_e \ll \omega L_e$ ) than  $X_e = \omega L_e$ .

The formula AkAbak uses to calculate the reactance of the voice coil impedance is:

<sup>2</sup>This effect can be easily demonstrated. Knock against the diaphragm of a loudspeaker. If the terminals are short circuited, you hear a short dull sound and if the terminals are open you hear a fuller sound. In the second case,  $R_e$  is infinitely large and the oscillating circuit is only attenuated by the mechanical loss resistance  $R_{ms}$ .

$$X_e = (\omega \cdot L_e)^r \quad \text{with} \quad r = \frac{1 + (\omega \cdot L_e / R_e)^{2 \cdot g}}{1 + (\omega \cdot L_e / R_e)^2}$$

This formula has been determined empirically and simulates the impedance graph of the voice coil surprisingly well. In the transmission range of the loudspeaker, between the resonance frequency and the frequency at which  $R_e = \omega L_e$  applies,  $X_e \approx \omega L_e$ . Only at frequencies for which  $R_e < \omega L_e$  applies does  $X_e$  continuously approach the above formula.

In the script the exponent 'g' is called `ExpoLe` which can be in the range of `ExpoLe=0 . . . 3`. In most cases a good fit is attained with `ExpoLe=0.5 . . . 0.7`. If you do not specify the parameter `ExpoLe` then `AkAbak` sets the default value `ExpoLe=0.618` (Golden Cut).

Please note, that although  $L_e$  is given in Henry,  $L_e$  is just a constant with no relation to a real inductivity-parameter unless `ExpoLe=1` in which case  $X_e = \omega L_e$ .

Using the values for voice coil inductance obtained from the data sheets

Since we can assume that these data are subject to a considerable degree of tolerance, they rather provide a rough estimate. We enter the value quoted for  $L_e$  just as it is specified in the datasheet. For an accurate simulation we require the measured driver parameter.

The effect of the modification formulae is that the value of `AkAbak`'s  $L_e$  is much larger than those usually given in datasheets ( $L_e$  @ 1kHz for example).

#### Simple measuring of $L_e$

It is best to use the tool for determining the voice coil parameter in the 'Tools/Dyn. Driver parameter...' menu. The value of the voice coil inductance is then accurately matched to the simulation method.

Otherwise measure the modulus of the electrical impedance of the driver  $|Z_e|$  at very high frequencies ( $R_e \ll \omega L_e$ ). In this frequency range the modulus function is simplified to:

$$|Z_e| = \sqrt{R_e^2 + (\omega \cdot L_e)^{2 \cdot 0.618}} \quad \left| \begin{array}{l} \text{ExpoLe} = 0.618 \\ R_e \ll \omega \cdot L_e \end{array} \right.$$

From this, it follows for  $L_e$ :

$$L_e = \frac{1}{\omega} \cdot \left( |Z_e|^2 - R_e^2 \right)^{0.5 \cdot 1.618} \quad \text{with} \quad \text{ExpoLe} = 0.618$$

## Electro-mechanic coupling

The mechanical part is coupled to the electrical via the transducer motor (Fig. 25).

In the electrodynamic transducer, the drive is specified by the equation  $F = B \cdot l \cdot I$ ,  $B$  being the magnetic flux density in the air gap of the permanent magnet,  $l$  the length of the voice coil wire in this gap,  $I$  the current through the voice coil and  $F$  the force with which the vibrating system is driven.

$Bl$  is called the force-factor. The unit is Tesla·Meter [Tm]. The force factor is one of the basic parameter of the dynamical motor. The appropriate equivalent circuit diagram of this drive is the gyrator (see also chapter Net/Gyrator). In principal  $Bl$  is easy to measure. Because  $F = Bl \cdot i$ , apply a d.c.-current  $i$  and measure the force of the diaphragm excursion. The more usual way to measure  $Bl$  is to calculate the moving mass  $M_{ms}$ , the electrical quality  $Q_{es}$  and the d.c. resistance of voice coil  $R_e$  (see Tools/Mms, Cms, Vas evaluation).

The drive transforms the impedance of the electrical side into the mechanical part:  $Z_m = (B \cdot l)^2 / Z_e$ . Together with the resistance of the voice coil  $R_e$ , this produces the electrical quality which is relative easy to measure from the electrical impedance curve:

$$Q_{es} = \frac{R_e}{(Bl)^2} \cdot \sqrt{\frac{M_{ms}}{C_{ms}}}$$



## Mechanical components

In the mechanical part, the common variable is the velocity  $v$  and the potential is the force  $F$ . The mechanical domain consists of the vibrating system. If the mechanics of the driver can be described by linear, discrete elements, the series circuit comprising the mass  $M_{ms}$ , the compliance of a spring  $C_{ms}$  and the resistance due to friction  $R_{ms}$  represents a simple basic model:

$M_{ms}$  and  $C_{ms}$  resonate at the resonance frequency:

$$f_s = \frac{1}{2\pi \cdot \sqrt{M_{ms} \cdot C_{ms}}}$$

The mechanical quality factor of this series oscillating circuit is:

$$Q_{ms} = \frac{1}{R_{ms}} \cdot \sqrt{\frac{M_{ms}}{C_{ms}}}$$

## Mass $M_{ms}$

The mass  $M_{ms}$  of the vibrating system comprises the mass of the voice coil, of the diaphragm and air load. The effective mass fluctuates with frequency, since the radiation reactance is frequency dependent and the diaphragm breaks up into partial vibrations at very high frequencies.

If the parameter  $f_p$  is entered, AkAbak operates with a frequency-dependent diaphragm area and mass (see Section 'Diaphragm parameter' in this chapter).

## Compliance $C_{ms}$

The compliance of the diaphragm suspension  $C_{ms} = \partial x / \partial F$  is the constant in a linear relationship between displacement  $x$  and spring force  $F$  and is equal to the reciprocal of spring stiffness. The quality of a loudspeaker depends, among other factors, on how well the designer has succeeded in holding  $C_{ms}$  constant, even at high diaphragm excursions.

## Mechanical resistance $R_{ms}$

$R_{ms}$  represents the mechanical losses in the diaphragm suspension.  $R_{ms}$  is usually considerably greater than the (mechanical) radiation resistance.  $R_{ms}$  attenuates the oscillating circuit comprising  $M_{ms}$  and  $C_{ms}$ . The greater  $R_{ms}$ , the smaller the value of the mechanical quality.

## Quality factors

The electrical quality factor  $Q_{es}$  and the mechanical quality factor  $Q_{ms}$  determine the transmission characteristics of the loudspeaker in the vicinity of the resonance  $f_s$ .

If we form the overall quality  $1/Q_{ts} = 1/Q_{es} + 1/Q_{ms}$ , we can estimate the transmission behavior using  $Q_{ts}$ . The ratio of the sound pressure at resonance to the average sound pressure in the transmission band is equal to  $Q_{ts}$ .

If the  $Q_{ts}=0.707$ , we get the maximum flat Butterworth characteristic of a high-pass filter. Higher values of  $Q_{ts}$  result in a magnification of the resonance. Lower values flatten out the transmission graph.

This applies for the loudspeaker in a sufficiently large baffle. If the driver is installed in an enclosure, the qualities and the resonance frequency vary. The aim of design is to find a satisfactory compromise between bandwidth, load-carrying capacity, efficiency and any further boundary conditions. You will find design aids for closed and reflex enclosures in the dialog for defining `Def_Bassunit` and in the literature.

## Mechano-acoustical coupling

The mechanical part is coupled to the acoustical part by the diaphragm surface. The equivalent circuit diaphragm for this coupling is the ideal transformer, as shown in Fig. 25 (see chapter Net/Coupler).

At least two couplers has to taken into account because each diaphragm has two sides. In some more complicate devices the moving diaphragm must divided into multiple sub-diaphragms.

The common design for midrange compression drivers for example has at least three diaphragm areas. The first radiates to the rear chamber. The inner front side of the concave dome is loaded with the compression chamber and the horn. The rest of the front diaphragm area forms a ring which radiates into several cavities which end in the compression chamber via the voice coil path (see exercises).

### The equivalent air volume Vas

The equivalent air volume to the compliance of the diaphragm suspension is:

$$Vas = Cms \cdot SD^2 \cdot \rho \cdot c^2$$

where

SD,  $\rho$ , c: diaphragm area, air density, velocity of sound

The parameter Vas is proportional to the compliance of the diaphragm suspension. Vas is purely a design parameter. Vas describes a theoretical volume of air whose compliance is proportional to the value of Cms. The parameter Vas originated from the design of bass loudspeaker enclosures. There, together with the volume of the enclosure, it represents a ratio that is easy to calculate.

In the program, you can specify the parameter Mms or Cms as well as Vas.

Please note, that the parameter Vas is highly dependent on other parameter and values of the surrounding medium. It is recommended to preferably use Mms or Cms in the specification. For instance if you enter Vas and modify the diaphragm area, AkAbak would calculate a fault spring force of the suspension.

The medium parameter c and  $\rho$  are fixed at this place and will not be modified when you change the system parameter ( $c=343.3\text{m/s}$ ,  $\rho=1.187\text{kg/m}^3$ , air @ 20°, see chapter 'File/Preferences/ Physical System Constants').

## List of driver models of AkAbak

AkAbak provides a range of compact driver models for the loudspeakers that are used most often in practice:

	Definition	Network element
Electrodynamic driver with one voice coil	Def_Driver	Driver
Electrodynamic driver with two voice coils	Def_TwoCoilsDriver	Driver
Piezo driver	Def_PiezoDriver	Driver
Loudspeaker in an enclosure	Def_Bassunit	Bassunit
Loudspeaker with enclosure	Def_Speaker	Speaker
Loudspeaker whose transmission characteristics are described by a table with measured values of sound pressure and impedance.	Def_MeasRad	MeasRad

Other driver types can be built up by supplementing or modifying the predefined drivers or by using lumped electrical, mechanical and acoustical elements. With all custom-designed driver types, the drive can be simulated either by means of an ideal transformer (Coupler), by a gyrator (Gyrator) or both. The mechanical

part is most easily specified with the help of the Impedance-element (see chapter 'Introduction/Transducer/Modeling a transducer').

## Parameter of transducers

$f_s = \dots \text{Hz}$		Resonance frequency of driver, Unit: Hertz [Hz].
$M_{ms} = \dots \text{kg}$		Mass of diaphragm assembly, including air load on both sides. Mms together with Cms or Vas can replace the resonance $f_s$ . Alternatively, the mass Mms with the resonance $f_s$ can replace the compliance factors Cms and Vas. With the four-pole elements, such as Def_Driver, etc., depending on which Meas_ keyword is entered, the program internally subtracts the imaginary component of the radiation impedance at $f_s$ from Mms (air load). Unit kilogram [kg].
$C_{ms} = \dots \text{m/N}$	(alternative)	Mechanical compliance of diaphragm suspension. Unit meter/newton [m/N].
$V_{as} = \dots \text{m}^3$	(alternative)	Equivalent volume of air to the suspension-compliance Cms. Note that Vas depends on the size of the diaphragm area and on the medium constants $\rho$ and $c$ . Unit cubic meter [ $\text{m}^3$ ], cubic inch [ $\text{in}^3$ ] or liter [Liter, L].
$Q_{ms} = \dots$		Mechanical quality factor of driver.
$R_{ms} = \dots \text{Ns/m}$	(alternative)	Mechanical resistance of suspension. With the four-pole elements, such as Def_Driver, etc., depending on which Meas_ keyword is entered, the program internally deducts the real component of the radiation impedance at $f_s$ from Rms. Unit newton-second/meter [Ns/m] or mechanical ohm [mks].
$Q_{es} = \dots$		Electrical quality factor of driver. Note that Qes depends inter alia on the d.c. resistance of the voice coil.
$Bl = \dots \text{Tm}$	(alternative)	Force factor of the voice coil. Describes the magnetic flux density in the magnet coil multiplied by the wire length of the voice coil. Only the length actually in the magnetic field is taken into account. Bl is the conversion factor of the electrodynamic transducer. Unit Tesla-meter [Tm].
$R_e = \dots \text{ohm}$		d.c.-Resistance of voice coil. Unit Ohm [ohm].
$f_{re} = \dots \text{Hz}$		Optional. Makes $R_e$ frequency dependent to simulate the influence of eddy currents at high frequencies. $f_{re}$ is the frequency, where $R_e(f)$ is doubled ( $\text{ExpoRe}=1$ ). Unit Hertz [Hz].
$\text{ExpoRe} = \dots$		Optional. Controls the resistance of the voice coil. Typical value: $\text{ExpoRe}=1$ .
$L_e = \dots \text{H}$		Optional. Reactance constant of the voice coil. If $\text{ExpoLe}=1$ then $L_e$ is the inductivity factor. Unit Henry [H].

ExpoLe=...

Optional. Controls the reactance of the voice coil.

Typical values: ExpoLe=0.5...0.7, default: ExpoLe=0.618,  
max. range: ExpoLe=0...3.

## Removing radiation load (Meas\_...)

Since a driver element does not radiate, the entered parameters, such as fs, Mms, Qms, Rms have to be freed from the real and imaginary parts of the radiation impedance, because in the simulation the radiation impedance is given by the elements connected to the front and rear side of the diaphragm, such as Radiator, Enclosure etc.

### Default

The program assumes that the parameters have been measured under conditions of dipole radiation (free radiation at both sides, no baffle).

### Other conditions

If the parameters have been measured under other conditions, you have to indicate that by means of a keyword:

Meas_Dipole	Default (i.e. applicable unless otherwise indicated) Free radiation at both sides without a baffle or enclosure. e.g.: bass loudspeaker $RmS' = 273.4 \cdot 10^{-9} \cdot fs^4 \cdot SD^3$ $MmS' = 0.565 \cdot SD^{3/2}$
Meas_TubeEnd	Free radiation at one side, no baffle, but driver mounted in a small enclosure. e.g.: dome tweeter $RmS' = 0.0108 \cdot fs^2 \cdot SD^2$ $MmS' = 0.408 \cdot SD^{3/2}$
Meas_Baffle1	Baffle radiates at one side. The rear side is closed by means of, for example, an enclosure. e.g.: dome tweeter installed in a cabinet or baffle $RmS' = 0.0215 \cdot fs^2 \cdot SD^2$ $MmS' = 0.565 \cdot SD^{3/2}$
Meas_Baffle2	Baffle and radiates at both sides (no enclosure). e.g.: loudspeaker in a baffle $RmS' = 0.043 \cdot fs^2 \cdot SD^2$ $MmS' = 1.13 \cdot SD^{3/2}$
Meas_DoNotModify	The parameters are not modified. If, for example, the parameters are already freed of the radiation impedance.

The above formulae are approximations for the radiation impedance in the frequency range far below the directivity frequency ( $ka \ll 1$ ):

$$ZmS' = RmS' + j\omega MmS'$$

fs is the fundamental resonance of the driver, SD is the diaphragm area. The program subtract RmS' and MmS' from Rms and Mms.

### ☒ Example

```
Def_Driver      'dome'
Meas_TubeEnd
dD=2.5cm  tD1=0.7cm  t1=3mm  fp=8000Hz  |Dome
fs=2.5kHz  Mms=1.1g
Qms=1.5  Qes=1  Re=5ohm
```

### Note

The components `Def_Bassunit` and `Def_Speaker` uses a much more simple means to handle the radiation load. Only the mass is effected by the factor  $mb=$  (see `Def/Def_Bassunit`).

## Sub-dialogs for $f_s$ , $Q_{ms}$ , $Q_{es}$ and $M_{ms}$

The parameter of resonance  $f_s$ , qualities  $Q_{ms}$  and  $Q_{es}$  and the vibrating mass  $M_{ms}$  can be replaced by alternative parameter: Compliance  $C_{ms}$ , equivalent volume  $V_{as}$ , mechanical resistance  $R_{ms}$  and transducer conversion factor  $Bl$ .

The alternative to choose for a particular case is a matter of expediency. For example, for tweeters, the diaphragm mass  $M_{ms}$  is entered instead of the equivalent volume  $V_{as}$ . For bass loudspeakers,  $V_{as}$  is used, since this volume has a simple relationship with the housing volume. If you want to include formulas for temperature dependency you have to use the basic parameter  $C_{ms}$ ,  $M_{ms}$ ,  $R_{ms}$  and  $Bl$ .

The `AkAbak` interpreter always calculates the basic parameter  $M_{ms}$ ,  $C_{ms}$  and  $R_{ms}$  from the entered parameter. If for example, the values for  $f_s$  and  $C_{ms}$  are entered, the mass  $M_{ms}$  is calculated from them, etc.

To convert the equivalent Volume  $V_{as}$  to  $C_{ms}$  the medium density  $\rho$  and the sound velocity  $c$  has to be employed. `AkAbak` uses constant values here and not those which can be entered in 'File/Preferences/Phys. parameter' ( $c=343.3\text{m/s}$ ,  $\rho=1.187\text{kg/m}^3$ , air @  $20^\circ$ ).

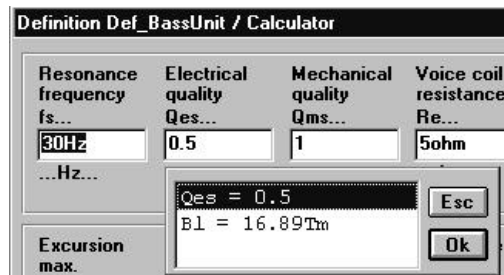
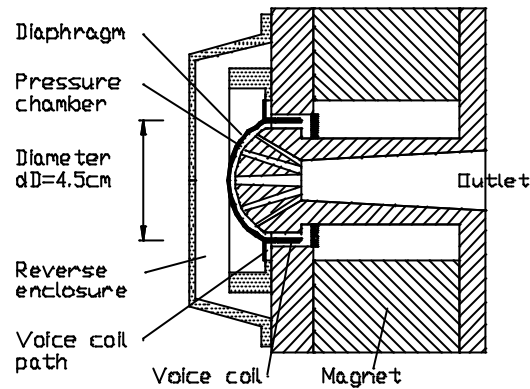


Fig. 26 Sub-dialog for entering the alternative parameter

The dialogs for assisting the entry of driver parameter have a small sub-dialog, which can be used for selecting the alternatives. You can see that this function is available by the three dots after the box identifier. The sub-dialog is opened with the **Alt+cursor keys** - `~` or with the right-hand mouse key. In the list, the desired alternative parameter are selected and inserted by means of the 'OK' button. If the type of parameter has been changed, it can be changed by again with the 'undo' function (Ctrl+Z).

## Modeling a transducer

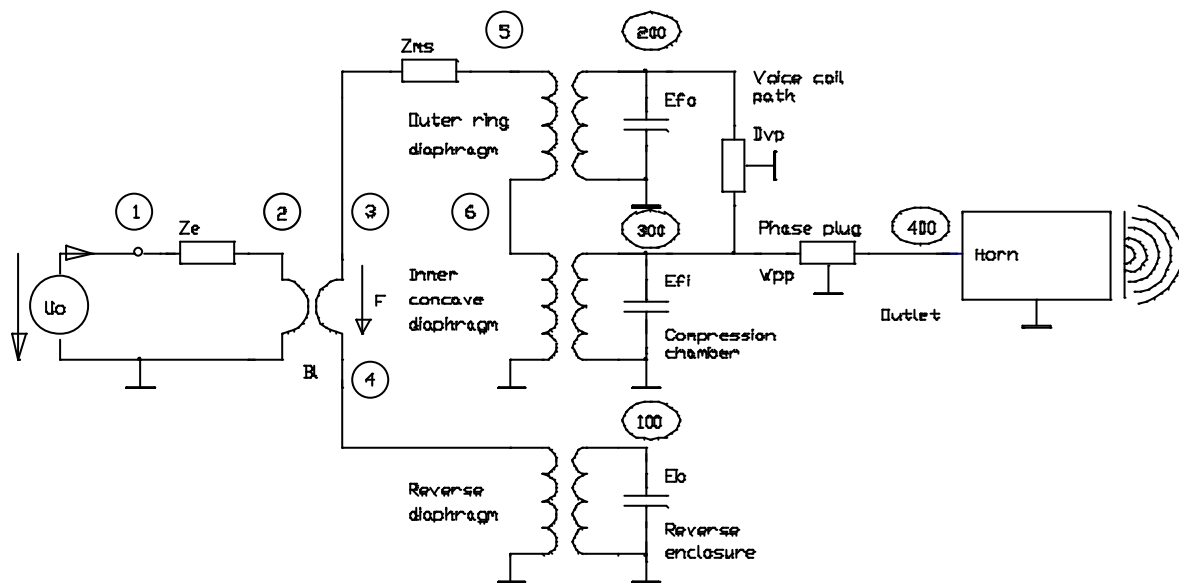
Let us, for an example, investigate a conventional mid-range compression driver as sketched out in Fig. 27. This class of transducers lead to a high sensibility of the response to the way the voice coil is embedded into the magnet structure. Usually there are small path ways through the magnet slit at a point within the total construction where a high pressure is generated. In fact, the diaphragm has to be split into at least two domains to be able to model this kind of driver.



## Modelling

The concave diaphragm of our compression driver radiates via a thin layer of air into a duct system at whose other end the horn is mounted (Fig. 27). Due to the diminishing cross section, air particles are accelerated to enhance the efficiency of the transducer.

The force of the voice coil drives the diaphragm at the outer rim. At the junction of diaphragm rim and voice coil former is fixed the suspension. The suspension of most compression drivers with a concave dome functions itself as a diaphragm radiating into cavities which are acoustically connected to the compression chamber. These cavities along the voice coil former are called the voice coil path and form an impedance, to which the system response is highly sensitive. The reverse side of the diaphragm is loaded by a small closed enclosure.



To model the driver we need components from the electrical, mechanical and acoustical domain. Fig. 28 is the equivalent circuit and below the script is shown (see file \Scripts\Examples\HDrv.aks). The centre-part of the schematic comprises lumped elements of the transducer. To the left we have the driving voltage source and to the right the four radiation devices which can be mounted at the acoustical output of the driver.

```
| AkAbak example script - Compression driver
```

```

Def_Const
{
    Ree=13;      |Voice coil resistance [ohm]
    Le=0.5e-3;   |Voice coil inductance [H]
    Bl=8;        |Motor conversion factor F=Bl*i [Tm]
    Mms=1.1e-3;  |Mechanical mass [kg]
    Rms=0.4;     |Mechanical resistance [Ns/m]
    Cms=140e-6;  |Mechanical compliance [N/m]
}

```

```

dDi=4.5e-2; |Diameter centre diaphragm [m]
dDa=5.3e-2; |Diameter outer diaphragm [m]
Hi=1e-3;    |Distance between diaphragm and phase plug
Ha=13e-3;   |Eff. height of outer chamber under the ring
ds=0.2e-3;  |Slit between voice coil and magnet
ls=15e-3; } |Length of voice coil path

System 'Windermere'
|Voice coil (frequency-non-linear resistance and reactance)
Impedance 'Ze' Node=1=2 Z={ Re*(1 + f/50e3) + j*(w*Le)^0.6; }
|Motor
Gyrator 'Gyl' Node=2=0=3=4 B1={B1}

|Reverse side with enclosure
Coupler 'Cplb' Node=4=0=100 dD={dDa}
Enclosure 'Eb' Node=100 Vb=120cm3 Qb/fo=0.1

|Mechanical part (Mms frequency depending, cut-off at 5kHz)
Impedance 'Zms' Node=3=5
Z={ wo=2*pi*5000; Zms=Rms + j*(w*Mms/(1 + w/wo) - 1/(w*Cms)) }

|Outer diaphragm (ring) with cavity
Coupler 'Cplo' Node=5=6=200 SD={ pi*(sqr(dDa/2)-sqr(dDi/2)) }
Enclosure 'Efo' Node=200 Vb={ SD=pi*(sqr(dDa/2)-sqr(dDi/2)); Vb=SD*Ha }

|Centre diaphragm with compression chamber
Coupler 'Cpli' Node=6=0=300 dD={dDi}
Enclosure 'Efi' Node=300 Vb={ SD=pi*sqr(dDi/2); Vb=SD*Hi }

|Voice coil tunnel between outer ring cavity and compression chamber
Duct 'Dvp' Node=200=300 SD={ U=pi*dDi; SD=U*ds } Len={ ls } QD/fo=0.01

|Horn inside compression driver
Waveguide 'Wpp' Node=300=400 dTh=1.5cm dMo=2.24cm Len=4cm Conical

|Radiation via radial horn
Horn 'H1' Node=400
T=1 dTh=1in WMo=53.5cm HMo=23cm Len=47cm
HArc=11cm LenTh=10cm
WEdge=59cm HEdge=27cm

```

The force generated by the magnetic fields of the voice coil and the permanent magnet is symbolised by the gyrator element 'Gyl'. The voice coil impedance 'Ze' is resistive at low frequencies. At high frequencies not only the imaginary part of Ze becomes inductive but also the real part rises due to eddy currents in the magnet pole piece.

The mechanical part is dominated by the fundamental resonance frequency formed by the suspension compliance and the mass of the vibrating assembly. Further there are more or less strong eigen-frequencies of the mechanical structure. It is possible to model some of them with AkAbak but for the sake of clarity this is omitted here. We would like our mechanical impedance 'Zms' to include the so-called mass reduction of the diaphragm. At high frequencies the acceleration is so strong that only parts of the diaphragm are able to follow the imprinted force.

The transformation from the mechanical to the acoustical domain is performed by the diaphragm, symbolised by the Coupler element. The front side of the diaphragm has to be split into two areas. The concave side of the dome radiates directly into the compression chamber (Coupler 'Cpli'). Part of the suspension ring radiates into the voice coil path (Coupler 'Cpli'). The reverse side of the diaphragm is coupled to the reverse enclosure (Coupler 'Cplb').

In the acoustical domain the cavities are modeled by acoustical compliances (Enclosure element 'Eb', 'Efo', 'Efi'). Passages where standing waves are expected are formed by waveguides. For the sake of clarity we are only including the most important modules in the script. We do not include resistive losses or dissipative Helmholtz resonators, for example. It turns out that the response is highly sensible to the dimensions of the path through the voice coil tunnel which is modeled using the Duct element 'Dvp'.

Here the phase plug model, which is in reality a sophisticated duct system, is formed by a simple Waveguide element 'Wpp' with variable cross section. The sound wave present at the outlet of the Waveguide element is also the output-port of the driver (node 400). At this port any acoustic device can be connected (Fig. 29).

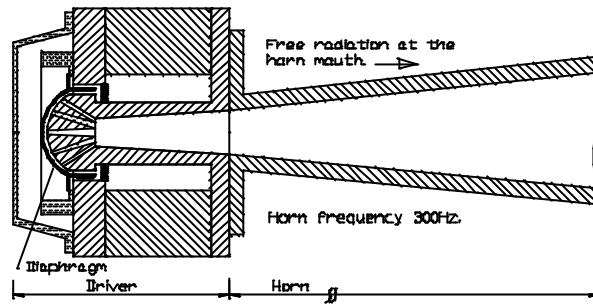


Fig. 29 Horn radiation

Fig. 30 displays the response of the driving point impedance of the driver as well as the transfer function when the compression driver is loaded with a radial mid-range horn.

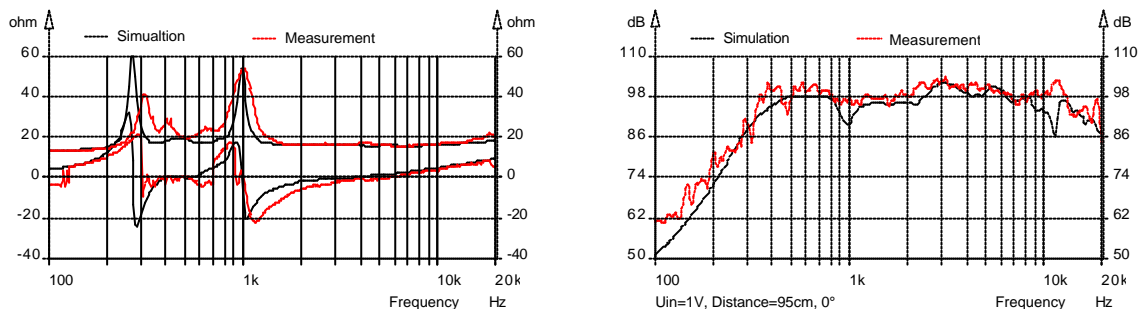


Fig. 30 Horn radiation: Electrical impedance, real and imaginary and sound pressure level

For example one of the most surprising details we can learn from our compression driver model is the high sensitivity of the response to the dimensions of the voice coil path. Fig. 31 demonstrates the effect of doubling the width of the slit between the voice coil former and the magnet.

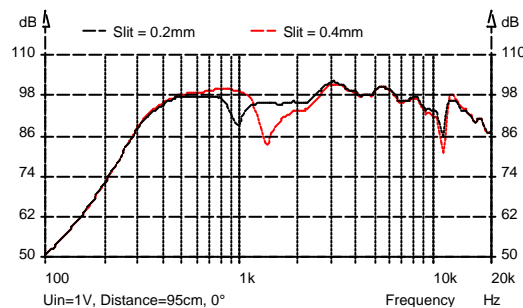


Fig. 31 Sound pressure level with horn radiation. Doubling the width of the voice coil path

## Modeling enclosed acoustics

Due to the method of simulation the propagation of waves in cavities is described exclusively by lumped elements and one-dimensional waveguides. Thus the existing world is of only one dimension. There are only points (lumped elements) and lines (waveguides). Constrained to these dimensions properties as 'up/down' or 'left/right' are unknown. Just 'forward/backward' are the directions the wave is allowed to propagate. Naturally, physical relations are much more easier to describe than in the three-dimensional world. In the case that there



is a pressure gradient into other dimensions we have to describe this effect with the help of the available set of elements.

As an example let us imagine a long duct bent at the center. The wave travels forward and backward with respect to the boundary conditions. One of the boundary condition is set up by the bend. In the line-world this bend is usually modeled with the help of an acoustic mass.

In an initial step a rough model has to be worked out. A rough model is made in such a way that especially fundamental resonances are taken into account. In a further step the sensibility of the analysis is tested with respect to details. As an example take the previously mentioned acoustic mass representing a bend in a duct. If there are only small or no effects due to the implementation of this mass between the ducts, this element can be neglected. On the other hand, if a high sensibility of the system-response against this point is observed a further detailed modeling is necessary.

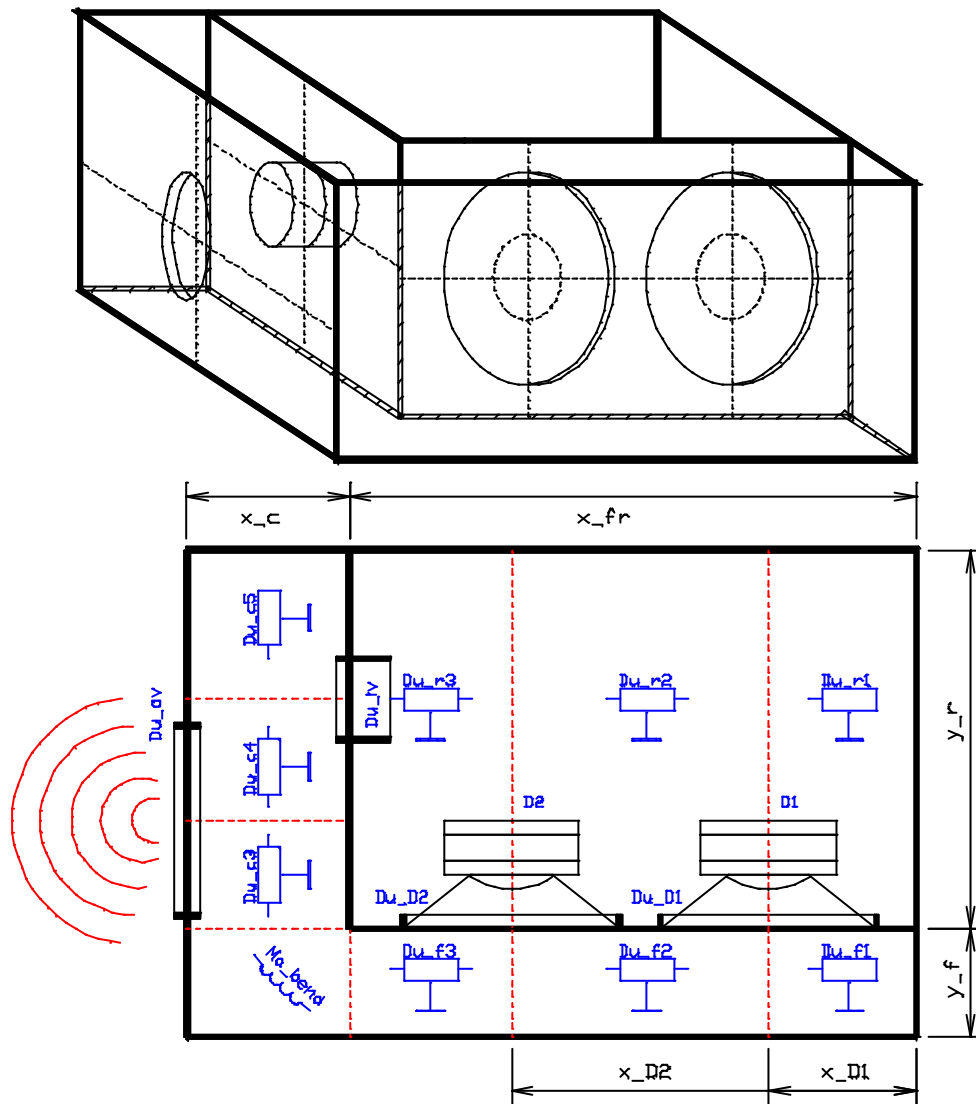


Fig. 32 Sketch of band-pass enclosure

## Example enclosure

The modeling of enclosed acoustic structures should be explained with the help of an example of a multi-chamber band pass enclosure (Fig. 32).

In this example there are two drivers. The front and rearward diaphragm sides are separated by an inner baffle. The sound wave which is radiated from the front side travels through a duct system, and after passing the output tunnel on the left hand side the sound is radiated into the free space. On the rearward side of the diaphragms the sound wave also moves through a duct system which is connected to the front system by the inner tunnel  $Du_{iv}$ .

## Analyzing the cavity

At very low frequencies, when the wavelength is much greater than the enclosure-dimensions, it is possible to describe this enclosure only with lumped elements such as the acoustic mass, compliance and resistance. But the bandwidth of simulation can be extended when we use waveguides instead.

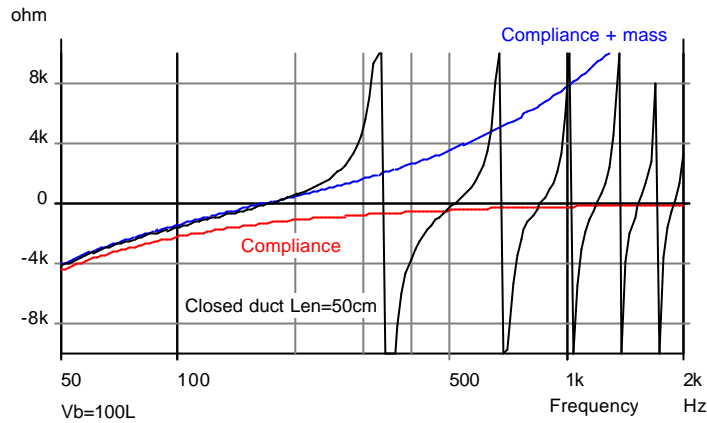


Fig. 33 Comparison of reactances.

One-dimensional waveguides are four-poles with impedance functions which describe the modi of vibration in one direction only. The analysis, for instance, of a closed enclosure with the help of a waveguide reveals at very low frequencies an impedance function similar to that of an acoustic compliance (Fig. 33). But with rising frequency the magnitude of reactance diminishes. Beyond the first eigen-frequency the reactance is of an acoustic mass. In Fig. 33 the reactance curve of a series combination of compliance and mass is added for comparison.

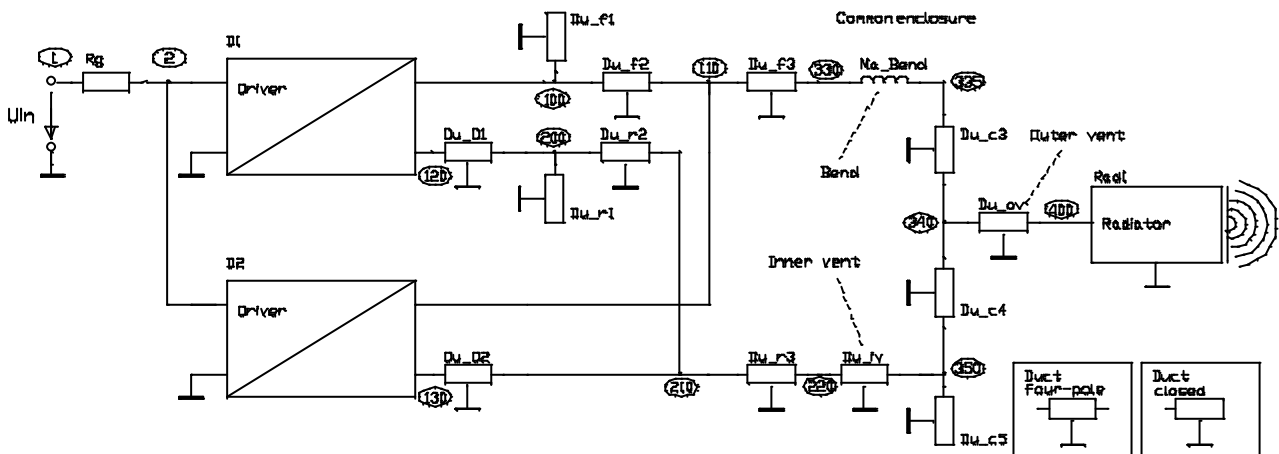


Fig. 34 Equivalent circuit with one-dimensional waveguides

Fig. 34 displays the equivalent circuit belonging to the sample loudspeaker-enclosure. Two electrically in parallel connected transducers are fed by the source voltage  $U_{in}$  through a generator resistance  $R_g$ . More electrical elements could easily be connected in front of the drivers but are omitted here for the sake of clarity.

The two poles on the right hand side of the driver symbols represent the front and reverse side of the diaphragm. The reverse sides of the diaphragms (lower pole) radiate each through a short duct with a ring formed cross section ( $Du_{D1}$ ,  $Du_{D2}$ ) into the rearward enclosure. This elements model the mounting way at which the sound is led through a displacement in the baffle. At this place an acoustic mass could do the same job.

The cavities in front of and at the rear side of the baffle are modeled by a chain of ducts. The frequency dependent wave-expansion is calculated from the right hand side to the left side of Fig. 32. The duct-chain starts with closed ducts ( $Du_{f1}$ ,  $Du_{r1}$ ). Note that closed ducts are painted un-grounded in the equivalent circuit because the flow, i.e. the volume velocity, is zero at the wall.

The distance from diaphragm 1 to diaphragm 2 is simulated by duct 'Du\_f2' and 'Du\_r2'.

In the front-cavity 'Du\_f3' is between diaphragm 2 and the bend. In the rearward cavity 'Du\_r3' connects the diaphragm 2 with the inner vent 'Du\_iv'.

The bend connects the front enclosure with the common duct-chain and the inner vent connects the rearward enclosure to the common enclosure.

After [Mor] a 90°-bend in a waveguide with constant cross section creates a reactance equivalent to an acoustic mass of approximately  $Ma=1.85/WD$ . WD is the width of the waveguide or, as in our example, the enclosure height  $z$ . Using this formula the bend has an acoustic mass of  $Ma=4.62\text{kg/m}^4$ .

Between the rear and the common enclosure there is the inner vent. 'Du\_iv' has a diameter of  $dD=6.7\text{cm}$  and a geometric length of  $L=5.5\text{cm}$ . If two ducts of different cross section are connected, an additional acoustic mass is to be taken into account. Inserting an acoustic mass is one possibility but it is also possible to simulate with a somewhat longer duct which is called 'end-correction'.

If the duct radiates into free space the end-correction is given by the radiation-reactance only. But inside enclosures the function depends also on the difference of cross sections. In our example a good end-correction was found to be  $L+=3.8\text{cm}$  for the inner vent Du\_iv, i.e. the effective duct-length is  $Len=9.3\text{cm}$ .

When air is pressed through a narrow tunnel there are losses due to viscosity and turbulence. By default, the program calculates viscosity losses inside waveguides. There are additional losses in sound fields with a strong velocity profile, for example, at the input stream to a narrow passage. To take into account this additional losses the Duct-parameter 'QD/fo' can be used.

The common enclosure is formed by a chain of ducts, too. At the center is mounted the vent 'Du\_ov'. At its outlet the radiation into the free space takes place. Note, that, in spite of being orthogonal to each other, no position parameters are applied to the duct-model. Only by additional masses the effect of changing direction is maintained.

The outlet-vent Du\_ov is just a hole in the enclosure wall. At the inner side an end-correction of  $L+=1.7\text{cm}$  is specified. The aperture of the outer side radiates into free space. No end-correction is necessary here because the simulator offers a complex radiation element called 'Radiator'. On the one hand the Radiator-element implements the correct radiation impedance including diffraction and reflection effects and taking into account the form of the diaphragm (conical, dome etc.). On the other hand the radiation into the listening room is performed, also taking into account diaphragm form, diffraction and reflection.

## Script

Now we can transform the equivalent circuit of Fig. 34 into the script syntax of AkAbak (see below, see example file 'Enclos2.aks'). Since the program itself has no graphical input device, usually, at the first stage we would do a sketch to understand the fundamental principals of the structure under investigation. Then an equivalent circuit might be developed and then the script is written.

```
| AkAbak Script: Multi Chamber Bandpass Enclosure
Def_Driver 'Drv 1'
  dD=16.5cm          |Outer diaphragm diameter
  dDl=5cm            |Inner diaphragm diameter
  tDl=2.5cm          |Cone depth
  fp=2.0kHz          |Mass reduction control frequency
  Mms=36g            |Mass of vibrating assembly
  Cms=0.53e-3m/N     |Suspension compliance
  Rms=1.23Ns/m       |Suspension resistance
  Bl=10.25Tm         |Conversion factor
  Re=6ohm            |Voice coil resistance factor
  fre=2.1kHz         |Voice coil resistance control frequency
  ExpoRe=0.78        |Voice coil resistance modifier
  Le=12.11mH         |Voice coil inductance factor
  ExpoLe=0.45        |Voice coil inductance modifier
Def_Const
{ z = 40e-2;         |Enclosure height
  y_f = 10e-2;       |Front enclosure depth
  y_r = 28.4e-2;     |Rearward enclosure depth
  x_fr = 51.6e-2;    |Front, rearward enclosures width
```

```

    x_c = 18.4e-2;      |Common enclosure width
    x_D1 = 12.1e-2;     |Position of first driver
    x_D2 = 36.3e-2;     |Position of second driver
}
System 'S1'
Resistor 'Rg' Node=1=2 R=0.5ohm |Generator resistance
Driver 'D1' Def='Drv 1' Node=2=0=100=120
Driver 'D2' Def='Drv 1' Node=2=0=110=130
|Front enclosure -----
Duct 'Du_f1' Node=100 Len={x_D1} HD={z} WD={y_f}
Duct 'Du_f2' Node=100=110 Len={x_D2 - x_D1} HD={z} WD={y_f}
Duct 'Du_f3' Node=110=330 Len={x_fr - x_D2} HD={z} WD={y_f}
|Rearward enclosure-----
Duct 'Du_r1' Node=200 Len={x_D1} HD={z} WD={y_r}
Duct 'Du_D1' Node=120=200 Len=4.5cm dD={18e-2 - 14e-2}
Duct 'Du_r2' Node=200=210 Len={x_D2-x_D1} HD={z} WD={y_r}
Duct 'Du_D2' Node=130=210 Len=2.5cm dD={18e-2 - 14e-2}
Duct 'Du_r3' Node=210=220 Len={x_fr-x_D2-1.6e-2} HD={z} WD={y_r}
|Inner vent (5.5cm+3.8cm end correction)-----
Duct 'Du_iv' Node=220=350 Len=9.3cm dD=6.7cm QD/fo=0.5
|Acoustical mass of bend -----
AcouMass 'Ma_Bend' Node=330=335 Ma=3.5kg/m4
|Common enclosure -----
Duct 'Du_c3' Node=335=340 Len={0.5*y_r} WD={x_c} HD={z}
Duct 'Du_c4' Node=340=350 Len=5cm WD={x_c} HD={z}
Duct 'Du_c5' Node=350 Len={0.5*(y_r+y_f)-5e-2} WD={x_c} HD={z}
|Output vent (1.6cm+1.7cm end correction)-----
Duct 'Du_ov' Node=340=400 Len=3.3cm dD=16cm
|Radiation into free space -----
Radiator 'Rad1' Def='Du_ov' Node=400
    x=0 y=0 z=0 HAngle=0 VAngle=0 |Mounting position

```

At the top of the script, in a so-called 'Definition', the parameter of the used drivers are located. The transducer-parameter has been determined with the help of the in-built tool for determining driver parameters from measurement curves. More below in the script, two Driver network-elements refer to this definition.

With the help of a further definition called 'Def\_Const' parameter can be specified which are accessible throughout the whole script. Because Def\_Const uses the in-built formula-parser not only values can be assigned but also formulae or formula-systems. Usually parameter values are specified in the associated unit like 'cm', 'L', 'uF' etc. But within the range of Def\_Const or within formula-specifications we have to use SI-units, for example 2e-2 for 2cm.

In our example the values of Def\_Const describe the dimensions of the enclosure. For example 'z' is the height of the enclosure which is  $z=40\text{cm}$ . With this arrangement it is easy to test the simulation also for other dimensions, simply by modifying a parameter at a central point.

The next lines of the script start with the network. Each network begins with the keyword 'System'. The network-components are connected with node-numbers. Values with the associated units or formulae are assigned to parameters. Formulae-assignments are embraced by curly brackets which can include identifiers specified in the Def\_Const definition.

## Simulation

The output at the chosen observing point is the system impulse response. The observing point can be one of the network parameters, as for example current and voltage or pressure and volume velocity, or a listening point in the free space. Displayed is the frequency or the time response. The latter is calculated with the help of the Fourier-transformation.

## Modeling procedure

The procedure of analysis depends especially on the intent of the simulation. In many cases only an idea shall be tested. Or, for teaching purposes, principal acoustical effects and relations shall be demonstrated.

In the case we use the simulation to help us designing devices such as loudspeaker systems or enclosures we strive for good control over the system properties. On the one hand parameters difficult to be measured shall be investigated such as for example the diaphragm excursion or the pressure somewhere inside the enclosure. On the other hand a good simulation permits us to scale parameter values for optimizing purpose or to test the sensibility of the system-response to parameter tolerances.

The designing procedure is accompanied by building up and measuring the device. Initially, the idea is formulated and a so-called blind simulation is carried out. The structure and the parameter values should be plausible and limited to the main properties of the structure under investigation. After the building up we specify the observing points which should be located in such a way that the system is totally described. In the current example we use the curve of the electrical driving point impedance and the sound pressure response directly at the output tunnel as observation points. After building up the enclosure the frequency response curves are measured at this observing points and then the measurement curves are copied into the simulation diagrams for comparison.

In the so-called post-processing stage the parameter values are modified and possibly elements are added or changed until the measurement and simulation curves match in a plausible way. Because nature uses an infinite amount of elements, whereas the simulator can model only in a rough way, a lot of effects will be displayed too sharp and exaggerated. Training is necessary to find this plausibility. But in many cases the post-processing proves to be a process-stage where we learn the most of our device under investigation.

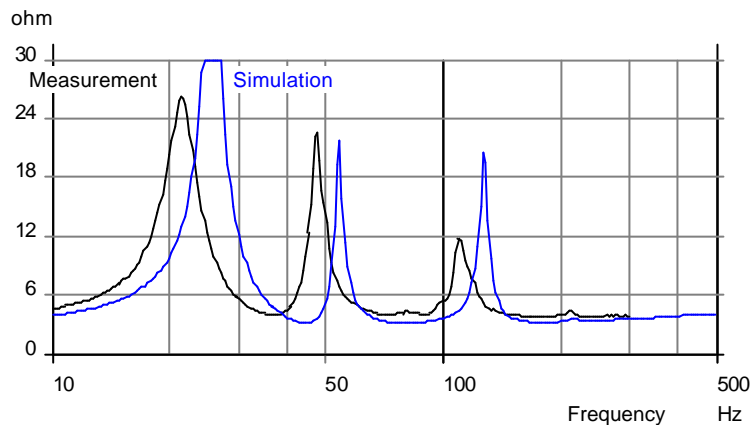


Fig. 35 First try. Magnitude of input impedance. Blind simulation. Measurement for comparison.

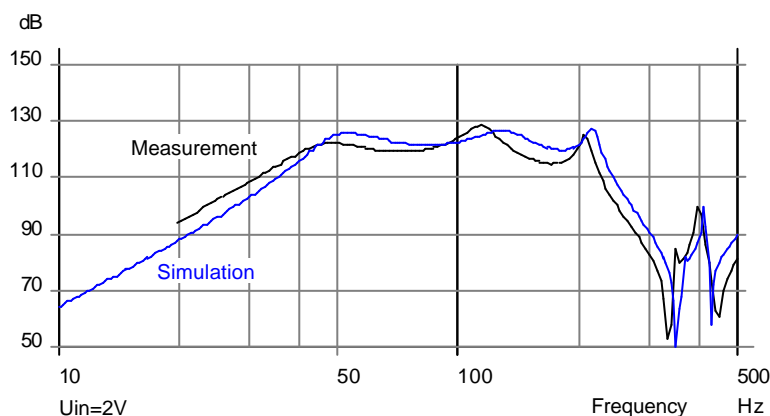


Fig. 36 First try. Sound pressure directly at the output vent. Blind simulation. Measurement for comparison.

## Blind simulation

Fig. 35 and Fig. 36 display the simulations of the electrical input impedance and of the sound pressure level exactly at the output vent. At this stage of development there was no loudspeaker available. Thus the parameter values were derived only from geometrical dimensions. The structure of the first script is identical with the final script with the exception of some details.

In comparison with the measurement curves it is obvious that the pattern is similar but the eigen-frequencies are located a little bit too high.

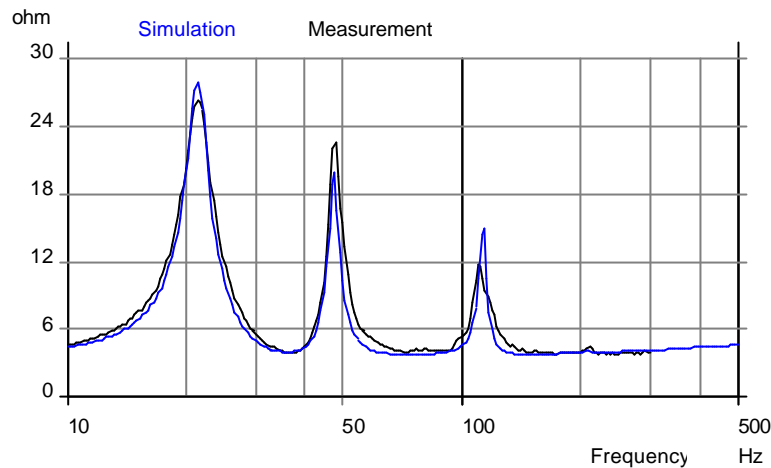


Fig. 37 Magnitude of input impedance Final simulation and measurement

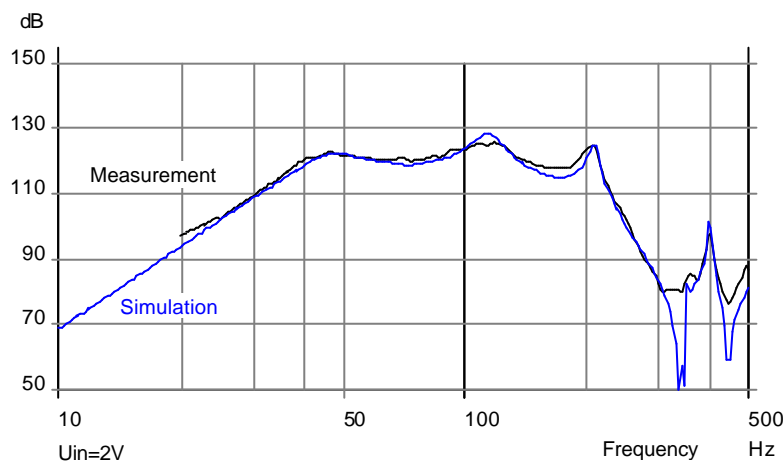


Fig. 38 Sound pressure at output vent. Final simulation and measurement

## Final simulation

Fig. 37 and Fig. 38 display the curves at the observing points of the final simulation in comparison with the measurement curves. The certainty that with this curves the simulation is confidential is gained during the post-processing stage. By carefully changing the one or another value and immediately observing the results of that, the post-processing owns a certain dynamic. Again and again it is astonishing to find out how exact the geometrical specifications must be entered. For example, a small variation of the waveguide-length or a slide change of the diaphragm-diameter leads to drastically changes in the response curves, and it is impossible to compensate for this with the variation of other parameter values.

After the script has become a confidential (but rough) electroacoustical figure of the loudspeaker, we are now able to investigate interesting parameters of the system which might be difficult to measure. You can try yourself after loading the script MultiBas.aks.

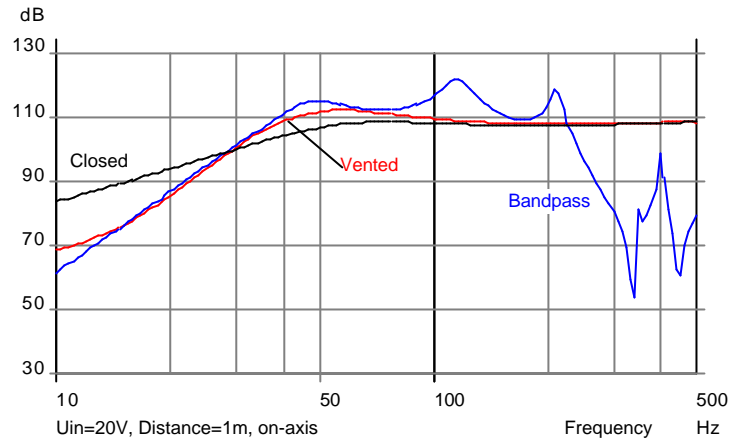


Fig. 39 Sound pressure level of equivalent enclosure types

## Comparison with other enclosure types

Fig. 39 compares the sound pressure curve of three equivalent bass loudspeaker constructions. Namely, the closed enclosure, the vented enclosure and the band pass enclosure as investigated in this paper. All three cavities have the same volume (approx. 100L).

With rising number of poles and proper chosen quality factors the available energy can be concentrated in a specified frequency band. The closed enclosure has only one pole-frequency and therefore it has the broadest bandwidth. The simple reflex enclosure has two poles. Tuned in the right way the energy at very low frequencies can be transformed into the listening range to enhance the bass response. Band pass enclosure can be modeled complex enough to create multiple resonances by which the sound energy is concentrated in a narrow bandwidth. To guarantee good efficiency it is important to gain amplification by high quality factors. Otherwise, in many cases the band limiting effect can be realized easier by electronic filtering.

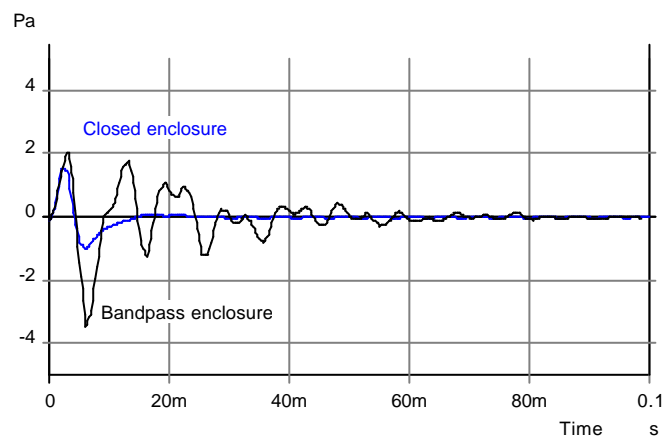


Fig. 40 Time pulse response of sound pressure of band pass and closed enclosure

With rising number of resonances the band pass curve of sound reproduction can be developed to a nearly rectangular form in the frequency domain. If we regard the resonance frequencies as harmonics of a Fourier-sum which build up the rectangular frequency response, it is clear, that the narrower the bandwidth of the rectangle and the steeper the slopes the longer will be the impulse response in the time domain.

Fig. 40 compares the time impulse response of the sound pressure of the band pass enclosure and of the closed enclosure. It is obvious that the gain which we achieve in the frequency domain leads to a prolonged response in the time domain, which, by the way, is clearly audible.

## Compact models

The previous example demonstrates the fundamental application of the lumped element method to the design and investigation of an acoustic structure. This procedure can be applied in the same way to other constructions like horn- and transmission line loudspeakers or any acoustic structures outside the loudspeaker domain.

Because there are a lot of loudspeaker applications using only more or less simple enclosures AkAbak provides some special compact modules:

Enclosure	Network element simulating rectangular and non-rectangular enclosures. Application: closed, reflex and band pass enclosures.
Bassunit	Network element for a complete bass cabinet as described in the associated definition Def_Bassunit. This definition describes closed and reflex enclosures, including a 2nd order electrical high pass filter. The associated dialog provides a wide variety of design aids. Application: Introduction to the design of bass loudspeaker enclosures, simple simulation.

## Helmholtz - Dialog

Any acoustical structure in which an acoustical compliance  $C_a$  resonates with an acoustical mass  $M_a$  is described as a Helmholtz resonator:

$$f_b = \frac{1}{2\pi \cdot \sqrt{M_a \cdot C_a}} \quad (\text{Helmholtz frequency})$$

The best known example is the reflex loudspeaker. In this case the volume of the enclosure forms the acoustic compliance:

$$C_a = \frac{V_b}{\rho \cdot c^2}$$

where

$C_a$ : acoustical compliance [ $\text{m}^5/\text{N}$ ]

$\rho$ : air density in enclosure [ $\text{kg}/\text{m}^3$ ]

$c$ : velocity of sound [ $\text{m}/\text{s}$ ]

(applicable for frequencies in which the wavelength is much larger than the dimensions of the loudspeaker; see also chapter Net/AcouCompliance)

The acoustical mass forms a vent, which connects the interior of the enclosure with the exterior room. If the dimensions of the vent are much smaller than the wavelength, the acoustical mass is:

$$M_a = \frac{\rho \cdot L}{SD} + M_a'$$

where

$M_a$ : acoustical mass of the vent [ $\text{kg}/\text{m}^4$ ]

$L$ : vent length in [ $\text{m}$ ]

$SD$ : vent cross-sectional area [ $\text{m}^2$ ]

$M_a'$ : acoustical masses of the air load at both sides of the  
acoustical masses at both sides of the duct [ $\text{kg}/\text{m}^4$ ]

The end correction is designated  $M_a'$ , since the imaginary part of the radiation impedance acts at both ends of the duct (see chapter Net/AcouMass). For ducts that terminate on both sides in a baffle, the termination correction in air works out to the following approximate value (both sides):

$$M_a' = \frac{\rho}{\sqrt{SD}}$$

Summarizing the parameter, the Helmholtz resonance works out as:



$$fb = \frac{1}{2\pi \cdot \sqrt{\frac{Vb}{c^2} \cdot \left( \frac{L}{SD} + \frac{1}{\sqrt{SD}} \right)}}$$

As already mentioned, this formula only applies in a frequency range in which the wavelength is much larger than the dimensions of the enclosure and of the vent. In the cases in which the Helmholtz formula leads to incorrect results in practice, hidden masses and compliances are present somewhere in the system. In most cases the wavelength is too short and the frequency is too high, respectively, for these simple formulae.

## Edit dialog

The Helmholtz dialog uses the above formula to calculate a reflex enclosure. It is reached via the 'Def/Helmholtz...' menu or as a sub-dialog in the entry dialogs for the Def\_BassUnit dialog and the network element Enclosure for the entry field of the Helmholtz resonance fb. To open the sub-dialog goto the 'fb'-field and press **Alt+up/down** or the right mouse button.

The screenshot shows a software dialog box titled "Enclosure". It is divided into several sections. At the top, "Element identification" includes a text field and a "Node" dropdown menu set to "s", with a "Copy" button. Below this is the "Helmholtz Resonator" section, which contains three columns of input fields: "Helmholtz resonance" (fb, 50Hz), "Enclosure volume" (Vb, 16L), and "Vent length" (Len, empty). The units for these fields are indicated as Hz, m3, and m respectively. Below the input fields is an "Evaluate" section with three buttons: "fb", "Vb", and "Len". At the bottom is a "Copy and close" section with two buttons: "fb, Vb" and "Len, Vb". On the left side, there is a "Vented enclosure" checkbox which is checked, and a "Vent cross section" (dD) field set to 10cm.

Fig. 41 Helmholtz resonance entry field 'fb' and the associated calculator sub-dialog for vented enclosures.

The 'Evaluate' group contains the action buttons for calculating the particular parameter.




If the entry box for the vent length is empty, the vent length L is set to zero. In this case the acoustical mass consists only of a hole in a thin enclosure wall.

When the 'fb' or the 'Len' button is pressed, the entered data are transferred to the dialog from which the sub-dialog was called, or copied into the clipboard.




# File, Edit, Search

This chapter describes the functions of the AkAbak menus 'File', 'Edit', 'Search' in short form. These menus comprise the operating functions of AkAbak. Please, refer to the help document where the program operation is explained in detail (Menu 'Help' or key F1).



## File

New script		Creates a new script and displays it in a window. The default file name is 'Script1', 'Script2', etc.
Open Scripts and Diagrams	Ctrl + O 	Opens the file dialog for loading a file already saved. This file can be a simulation script, an import script, a diagram or any text-file.
Save	Ctrl + S 	Saves any script or diagram currently active. If 'Preferences/ Make backup scripts' is switched on then the previous copy of the current file is saved.
Save As...		Opens the file dialog for storing the script or diagram currently active under a different file name.
Desktop -> Open Save Save as		With program exit AkAbak stores script- and diagram-windows in a so-called desktop-file. With this command you can choose from different desktop files. This option is useful when you are working on several projects.
Export -> as ASCII  as metafile	Ctrl + T 	This function transforms the data of any diagram into a file of ASCII-text-format. By this, you are able to process simulation results by other programs.  Saves the current active diagram to disk in the Windows Metafile format (Alternative to 'Edit/ Copy as metafile')
Printing -> Do print Printer Settings...	Ctrl + P	Prints the contents of the currently active script window or diagram window.  Choose one of the installed printers.
Preferences > Screen Diagram Style... Printed Diagram Style...  Physical System Constants...  Company Name... Directory Def_Import files...  Directory Def_MeasRadiator files...  Show Button Panel Make backup scripts		Set the colors, line thickness and line style of the lines in all diagrams for the screen and for print outs.  A small database with settings of the velocity of sound and density of the medium. The setting is effective throughout the program. Default: Air @ T=20°, c=343.3m/s, $\rho=1.188\text{m}^3/\text{kg}$ .  Opens a dialog for entering the company name.  Default folder for import files.  Default folder for Def_MeasRadiator files.  If switched on the button bar is displayed  If switched on AkAbak will create backup-files each time a script is modified and saved. The name and path of the backup-file is the same, but it gets a different filename-extension (*.bks).
Exit	Alt + F4	This command terminates AkAbak. The next time AkAbak is called up, all windows, except for the dialogs, are shown in the same state as when they were left.

## Edit

Undo	Ctrl + Z	Undo the last delete or copy action. 'Undo' also restores the last deleted graph of a diagram.
Cut	Ctrl + X	Copies the marked text into the clipboard and deletes it. In the diagrams, the marked graph is copied into an internal clipboard and removed from the diagram.
Copy	Ctrl + C	Copies the marked text into the clipboard. Diagram copies the marked graph into an internal clipboard.
Paste	Ins, Ctrl + V 	Inserts the text from the clipboard at the cursor position. Any marked text is overwritten. If a diagram window is active, the last copied graph is inserted into the diagram as a so-called 'guest graph', with its own ordinate.
Delete	Del	Deletes the marked text or a graph (without copying into the clipboard).
Diagram		
Copy as metafile		Copies the entire diagram currently active into the Windows clipboard. Format: Windows Metafile (vector graphic)
Copy as bitmap		Copies the entire diagram currently active into the Windows clipboard. Format: Bitmap (pixel graphic)
Range...	Return 	Dialog for the diagram range management.
Comment...	Ctrl + Return 	Attaching of any text to the diagram which appears in the print out.
Legend...		Editing the legend and unit of a graph.

## Search

Find...		Searches for text in a script from the cursor position.
Replace...		Like the 'Find...' function. The found text is additionally replaced by the specified text.
Next	Ctrl + L	Repeats the last 'Find...' or 'Replace' operation.
Move Nodes	Ctrl + N 	This command shifts the node numbers entered for the network elements.
Current Element	Ctrl + E 	Opens the associated input dialog for that particular element or definition in which lines the script cursor is currently located.

# Import

Menu: File/ Import/

The import facility of AkAbak is held as flexible as possible. Nearly any data-rows in ASCII-format exported by another program can be displayed in a diagram and processed by the functions available in the 'Calc'-menu. The intention of the AkAbak import is:

1. Comparison of measurement with simulated curves.
2. Calculation and processing of a time/frequency/directivity response.
3. Transducer parameter determination
4. Using measurement curves as a component.

## The three stages when importing data:

1. Data row in ASCII-format, usually a time or frequency response.


Adjust the exporting program so that the necessary conditions are met (see below). Export to any directory (preferably to \AkAbak\Import\) or to the Windows clipboard.

2. Descriptor, which tells AkAbak how to interpret and process the data.

The interpretation and processing information are specified and saved in a special definition called `Def_Import`. `Def_Import` definitions can be part of any simulation script or saved in a special file called Import Script. `Def_Import` can be edited directly or with the help of the `Def_Import` Dialog. The `Def_Import` Dialog can load the data also directly without a `Def_Import` definition.

3. Diagram to displays the data

Once a `Def_Import` is entered move the cursor in its lines and in AkAbak issue the command 'File/Import/Do

Import...' or click on  to load the data. Initially the data are displayed in the Import Display Dialog where necessary adjustments can be carried out before the curve is copied to any diagram as a guest graph or a new diagram is created.

## Data Format Requirements

- The data have to be in ASCII-text form and be stored in the Windows clipboard or on a data medium.
- The abscissa values may be arranged in increasing or decreasing order (auto-detection).
- If an abscissa value is present, the abscissa value comes first in a line, then the ordinate value(s).
- Space- and text-lines are ignored.
- Leading blanks or tabs in a line are ignored.
- The values in a line are separated by one of the following characters: space, tabulator (ASCII no. 9), comma ( , ), semicolon ( ; ), colon ( : ), underline ( \_ ), slash ( / or \ ) or vertical bar ( | ).
- The decimal point is a stop (.), Thousand separators generate an error.
- Units are not supported.
- The valid magnitude range is:  $0 < |a| < 10^{15}$
- At least **2** value-pairs have to be available for normal import.
- At least **30** value-pairs have to be available for import for the tools.
- The maximum number of data points which can be imported are 64000. From this a maximum of 1024 points can be displayed and processed. In the case there are more points to be imported the program performs automatically a five point smoothing before picking out the points to be displayed. Any number smaller then 1024 points are displayed one to one. In the case a lin-log transformation is to be done, linear interpolation is applied.

## Import Script

'Import Script' is intended to save and comprise Def\_Import definitions. It should help you to organize measurement data or other files to be imported. Use these scripts similar to the normal simulation script, with respect to editing, loading and saving. Note, that it is also possible to place a Def\_Import-definition anywhere in simulation script.

Similar, but in some points different is the import for tools using the import facility. There is a logistical problem, namely, there can be open only a *single* version of the Import Script, i.e. AkAbak and AkTools can not *open the same Import Script at the same time*. A warning message is displayed when you try to open an Import Script twice.




### Hints for using Import Scripts

An approved way of management is to have Import Scripts for import in AkAbak and other Import Scripts for tools. After some time Import Scripts can become very large. Here some organization hints:

- Use generic import specifications, i.e. a handful all purpose Def\_Import definitions. In many cases, data are imported only temporarily and must not be documented. Export then, for example, always to special files which are override with each export or use the clipboard. Define yourself some Def\_Import-definitions, like one for sound pressure measurements and another with the setup for impedance measurements. Then the only thing you have to do is, to move the cursor in the lines of the specific Def\_Import and issue 'File/Import/Do Import...' or with tools issue 'Import...'.
- Use different Import Scripts for each project or tool.
- For quick search use unique identifiers for the Def\_Import-definitions.
- Don't spare with comments.

## Import Script in AkAbak

There can be open only **one** Import Script at the same time. There is a yellow band on the left hand side of an Import Script-window. For tools, see Import Script Editor for Tools.

Operation	Menu	Button bar	Key
Open the recently used Import Script	File/ Import/ Do Import...		Ctrl + R
Open any Import Script	File/ Import/ Open Import Script... or File/ Open Scripts and Diagrams...' with 'File Format' = 'Import *.aki'.		Ctrl + O
Save an Import Script	File/ Save or File/ Save as...		Ctrl + S
Create a new Import Script	File/ Import/ New Import Script...		

## Import Script in Tools

The tools program has its *own* Import Script editor where you can edit and save the Def\_Import definitions and which controls also the import. The mechanics are very similar to the 'all purpose import' of AkAbak.

The imported data are directly displayed in the tool-dialog. Following tools evaluate import data

Tools/Create MeasRadiator file

Tools/Dyn. driver parameter

### Tools/Piezo driver parameter

These tools are located in an alone standing program (AkTools.exe) although launched by the AkAbak menu system. And therefore it is not possible to access Def\_Import descriptors which are located inside scripts of AkAbak directly.

There is a logistically problem, namely, there can be open only a *single* version of the Import Script, i.e. AkAbak and AkTools can *not open the same Import Script at the same time*. A warning message is displayed when you try to open an Import Script twice.

### Activating the Tools Import Script Editor

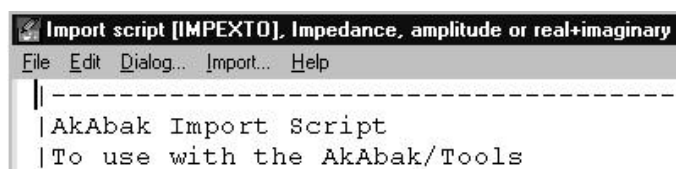
In the particular tools dialog there is a typical button 'Import Script...' which opens or activates the Tools - Import Script Editor.



Depending which tool called this editor, the window-header of the editor reports the type of import and in which format the curve will be displayed later (see below).

### Operating the Tools Import Script Editor

All commands are issued by a menu at the top of the window.



Operation	Menu	Key
Create a new Import Script	File/ New	
Open Import Script	File/ Open...	Ctrl + O
Save an Import Script	File/ Save or File/ Save as...	Ctrl + S
Activate the associated tools dialog	File/ Goto Dialog	F11
Closing of editor	File/ Close	Alt + F4
Edit commands	Edit/ Undo, Cut, Copy, Paste	Ctrl + Z, X, C, V
Search commands	Search, Replace, Next	Ctrl + L
Opens the Def_Import - dialog	Dialog...	Alt + D
Import using Def_Import definition where the script cursor is currently placed.	Import...	Alt + I

### Import...

The curve is displayed in the diagram of the tool window. The format is fixed and reported in the header of the editor. When the complex mode is switched on, the conversion is done automatically.

Importing in non-complex mode the imported data should match to the above listed format. Otherwise switch to complex mode and convert the data appropriately using the ordinate formula.

Tool	Data	Scalar	Vector
MeasRadiator	Sound pressure	Level	Level + Phase
MeasRadiator	Impedance	Amplitude	Real + imaginary
Dyn. driver parameter	Impedance	Amplitude	Real + imaginary
Piezo driver parameter	Impedance	Amplitude	---

## Def\_Import

*Dialog: Def/ Def\_Import*

In this definition interpretation and processing information for import files are specified and saved. Once specified move the script cursor into its lines and start the import by issuing 'File/Import/Do Import...' or click on



. Def\_Import definitions can be part of any simulation script or saved in a special file called Import Script (see above).

## Parameter

Def_Import	Keyword
'...'	Identifier of definition. In the current version there is no element referring to Def_Import. But in large Import Scripts a unique identifier has been proved practical to search for an item.
Legend='...'	Comment on the measurement like driving voltage, distance etc. This string appears as curve legend in the diagram.
Filename='...'	The filename of import file. There is no reserved filename extension for import files. When no path is specified for the filename the default path is valid. The default path for import files can be set in File/Preferences/Def_Import Directory. By default it is the AkAbak\Import\ directory. No assignment to the filename switches to the clipboard as import source.
Complex	Keyword, when assigned the imported data are interpreted and processed as complex values. See: Complex mode Non-complex mode
For1={...} For2={...}	Ordinate-formula systems to modify the import data. For2={...} is not used for 'Complex'-mode.
ForAbsc={...}	Abscissa-formula system to modify the import data.
NoAbsc	Special case when the import data has no abscissa on its own.
ColNo=...	Special case when multiple ordinate columns are present.

---

BeginId='...'	
EndId='...'	Optional file interpretation parameter. Markers which frame the actual data points.
FirstLine=...	
LastLine=...	Optional file interpretation parameter. Line numbers where the actual data starts and/or end.
DataBegin	
FileBegin	
EndOfFile	Optional file interpretation parameter. Controls the LastLine= parameter. Default setting is 'DataBegin'.
PointSep='...'	Point separator.

## Ordinate Formula System

You can enter a formula system that converts the ordinate values given in the file. More than one formula can be entered. Each of these formulae is separated by a semicolon (;). The result of the last formula gives the ordinate value of the curve in the diagram. It may have an identifier or not. The entry is limited to **255** characters.

## Complex Mode

Import and processing of data points in complex calculation mode.

```
Def_Import  'SPL1'
  Complex
  ...
```

When the **Complex**-keyword is assigned within the Def\_Import definition AkAbak assumes that each point consist of a real and an imaginary part, where first comes the real part and then the imaginary part. The values are assumed to be a peak value. In this example of a data file the first column is the abscissa, the second column is the ordinate-real part and the third column is the imaginary part:

```
20.0    10.0   -5.0
22.0    11.0   -4.5
...
```

The advantage of the Complex mode is that on the one hand complex calculation can be applied and on the other hand the Bode type of the frequency response can be chosen freely in the output diagram.

### Complex ordinate formula

With the **Complex**-keyword assigned, the **Um** variable in the ordinate-formula-system is complex and carries exactly the values of the first and second column of the import data. In the following example the formula 'ZL = Rv/(Uo/Um - 1);' is complex since **Um** is a complex variable. Further, the result of the formula-system is also complex.

```
Def_Import  'Imp_Z'
  Legend='Impedance of driver'
  Complex
  Filename='Meas_U.cur'
  For1={ Rv=102.3;   Uo=2;   ZL = Rv/(Uo/Um - 1); }
```

When your import data are not of the format 'real and imaginary' but for example of 'amplitude and phase' and complex processing shall take place then you should transform the data to 'real and imaginary' format with the help of special functions available in the ordinate-formula (phase in Degree).



<code>Uz = APtoRI(Um)</code>	Amplitude+phase is converted to real+imaginary
<code>Uz = LPtoRI(Um)</code>	Level+phase is converted to real+imaginary
<code>Uz = LpPtoRI(Um)</code>	Sound pressure level+phase is converted to real+imaginary

The result of the `LPtoRI(Um)` and `LpPtoRI(Um)` function is a peak value.

For example, when your measurement device cannot export in 'real and imaginary' format, but for instance in 'amplitude and phase' then the above `Def_Import` example would be:

```
Def_Import 'Imp_Z'
  Legend='Impedance of driver'
  Filename='Meas_U.cur'
  Complex
  For1={ Uz = APtoRI(Um); Rv=102.3; Uo=2; ZL = Rv/(Uo/Uz - 1); }
```

The complex mode works also even when there is no imaginary part in your import data. When no ordinate formula is assigned the imaginary part is zero then. Otherwise the complex result depends on the result of the formula system. Note, that the result of the last formula in a formula system creates the output curve. It may have an identifier or not.

## Non-Complex Mode

Import and processing of data points where each component of the ordinate is processed independently.

When the 'Complex'-keyword is **not** assigned within the `Def_Import` definition, AkAbak displays the ordinate values in the form as they are given in the import file, eventually modified by the ordinate formulae.

In contrast to the Complex-mode where the Bode type is calculated from the real and imaginary parts, here, the output-Bode type cannot be calculated but is given by the format of the import data. You can select a Bode-type in the non-complex mode, too, but it would only change the units and the grid of the output diagram.

The values of the ordinate points can be of any type, e.g. scalar, vector, amplitude + phase, real + imaginary, etc.

The non-complex mode is usually used for quick and simple calculations and when the import data format is the same as the format to be displayed.

## Real ordinate formulae

When the 'Complex'-keyword is **not** specified two formulae can be used. The first one modifies the first ordinate value and the second formula modifies the second ordinate value.

In both formulae '**Um**' stands for the associated import value. For example:

```
Def_Import 'Drv1_Lp'
  Legend='Lp of drv1, Uin=1V, r=1m'
  Filename='Meas_Lp1.cur'
  For1={ r = 2; Lp = Um + 20*log(r) }
  For2={ c = 344; k = 2*pi*f/c;
         r = 2; pp = Um - deg(k*r) }
```

In the above example a loudspeaker has been measured in a distance of  $r=2\text{m}$ . The diagram shall display the curve as it would be in one meter distance. The first formula-system '**For1**' calculates the sound pressure level. Here '**Um**' is the measured level. The second formula-system '**For2**' extrapolates the phase for one meter distance. Here '**Um**' is the measured phase response in degree.

Another example, where only the level is shifted by 6dB:

```
Def_Import 'Drv2_Lp'
  Legend='Lp of drv2, Uin=1V, r=1m'
  Filename='Meas_Lp2.cur'
  For1={ Um + 6 }
```

Note, that the result of the last formula in a formula system creates the output curve. It may have an identifier or not.

### 'Um' - Access to the ordinate values

During import the ordinate value(s) of each point of data is copied to a variable called **Um**. The variable **Um** thus represents the imported file value of the ordinate and you can use it as often as you like in the formula-system. The result(s) of the formula system(s) are then the modified import values of the ordinate. A simple example would be:

```
For1={ Um + 6 }
```

In this example the value '6' is added to the ordinate-value of each point of the imported data. In the case the imported data are vectors, i.e. consists of two columns like real and imaginary, the number '6' is added to the first ordinate component of the data point.

When the complex mode is active the number '6' of this example is added to the real part of **Um**. To add '6' to the imaginary part you would enter

```
For1={ Um + j6 }
```

When the non-complex mode is active 'For1={ Um + j6 }' would make no sense because **Um** is a real number. To modify the second component of the ordinate use the second formula-system. For example:

```
For2={ Um + 6 }
```

Here '6' is added to the phase, imaginary part or whatever makes the second component of the ordinate.

### 'x, f, t' - Access to the abscissa values

During import the abscissa values of each point is copied to a variable with the reserved names:

**x, f or t**

These variables thus represent the imported file values of the abscissa. Use either of them or mixed and you can use them as often as you like in the formula-system. 'x', 'f' or 't' all carry the same value. Only the identifier is different to fit best to your documentation of the formulae.

```
...
Complex
For1={ w = 2*pi*f; Uz = Um/jw }
```

In this example the complex import points are differentiated with respect to the time domain.

## Abscissa Formula System

You can enter a formula system that converts the abscissa values given in the file. With the help of this feature scaling, generating, mirroring, etc. of the abscissa is possible.

More than one formula can be entered. Each of these formulae is separated by a semicolon (;). The result of the last formula gives the abscissa value of the curve in the diagram. It may have an identifier or not. The entry is limited to **255** characters.

```
ForAbsc={ ... }
```

### 'x, f, t' - Access to the abscissa values

During import the abscissa values of each point is copied to a variable with the reserved names:

**x, f or t**

These variables thus represent the imported file values of the abscissa. Use either of them or mixed and you can use them as often as you like in the formula-system. 'x', 'f' or 't' all carry the same value. Only the identifier is different to fit best to your documentation of the formulae.

Note, that 'Um' is not used in the abscissa formula because it represents an ordinate value.

The result of the last formula gives the new abscissa value of the curve in the diagram.

```
ForAbsc={ t/1000 }
```

This example might be useful when you read a time signal and the time axis is normed to milli-seconds. The curve is then displayed in seconds (important when FFT is used later).

## Import of Files Without an Abscissa

It is also possible to import data when there are only ordinate values but no abscissa, for example, when you import results of an FFT calculation or similar. In this case the first column in your data-file is the first ordinate value.

In case there is no abscissa the keyword **NoAbsc** must be specified within the lines of Def\_Import. The program creates numbers 1,2,3,4,... for each point. The first valid line is assigned the number one. With the abscissa formula these can be scaled and shifted to any desired values. For example:

```
Def_Import 'NoAbsc'
Legend='N=1024, To=1ms'
Filename='NoAbsc.cur'
NoAbsc
ForAbsc={ To = 1e-3; Tx = (x - 1)*To }
```

In this example a data row shall be displayed which consists only of 1024 ordinate values. There is no abscissa. Known is only the time interval To=1ms. The first value begins with Tx=0.

With the specification of 'NoAbsc' and 'ForAbsc={ To=1e-3; Tx=(x - 1)\*To }' following abscissa is created:

$T(1)=0, T(2)=1e-3, T(3)=2e-3, \dots T(1024)=1.023$

i.e. when the first valid number line is read in, '1' is copied to the variable **x**. '2' is copied to **x** with the next number line, and so on.

## File Interpretation Parameter

AkAbak, is only interested in the sequence of numbers of the curve. The import interpreter therefore separates the sequence of numbers from commentaries and other information which might be in the file to be imported.

Note, that the import interpreter has a lot of options from which you would need probably only one setting. For many files to be imported you even need no special setting at all, for example for files the MLSSA system exports. Once set, maintain this parameters by copying and pasting of the Def\_Import - definition.

## Default setting

All lines which are non-numeric are treated as blank lines and are thus ignored. The natural end of the point-collection is the end of file.

## Example

In the following example the first two lines are ignored and, since the sequence of numbers are continued to the end of file, no interpretation parameter is necessary.

```
'Sensitivity Bode Plot - dB SPL/watt'
'Hz' 'Mag (dB)' 'deg'
```

14.7964,	87.23951,	142.7237
29.5928,	84.6204,	106.7425
44.3892,	79.54994,	77.7931
59.18561,	71.94746,	75.3687

## Marked by identifier

The beginning, and possibly also the end, of the sequence of numbers can be marked by an identifier. The identifier is not case sensitive. Both are optional and override the parameter 'FirstLine=' and 'LastLine=' (see below).

BeginId='...' marks the line in front of the first number line.

EndId='...' marks the line one behind the last number point.

## Example

There are cases where numeric lines are part of a header which describes the settings of the measurement device as in following example. When you try to import this file without interpretation parameter, AkAbak would report an error message. Here the numbers of the header would be interpreted as points of the data. This measurement device (Kirchner ATB) exports always the identifier '/Values' in front of the actual data which continues to the end of file.

```
/DBINPUT
-30
/DBOUTPUT
6
/SMOOTH
2
/DATASIZE
2
/RECORDS
250
/VALUES
1000.0000    72.7048
1012.1037    73.9701
1024.3539    76.0440
1036.7524    75.8420
...
```

When using the above example you should specify following interpretation parameter. EndId='...' is not needed here because the sequence of points continues up to the end of file.

```
Def_Import
...
BeginId='/VALUES'
...
```

## Marked by line numbers

Another way to cut out a sequence of points is the specification of line-numbers. This option is used only in special cases.

FirstLine=... marks the first number line counted from the file begin starting with 1.

LastLine=... marks the last line number. If not specified the end of file is set by default. Here the switch DataBegin, FileBegin, EndOfFile is important.

---

DataBegin	Keyword. LastLine is counted from the begin of the data. Default value.
FileBegin	Keyword. LastLine is counted from the begin of the file.
EndOfFile	Keyword. LastLine is counted from the end of file.

The contents of the 'FirstLine=...' and 'LastLine=...' parameter are ignored if an entry has been made in the BeginId='...' and EndId='...' parameter (see above).

### Example

If the data are output in a fixed structure and the sequence of numbers is not initiated or terminated by an identifier, you can also give the first and last line numbers of the sequence of numbers. The first line receives the number 1. For example:

```
any text1, any text2,
any text3
20.0    92.3
23.4    92.4
...
19234   65.3
any text4
...
```

The parameter 'FirstLine=' then contains the number 3, since the relevant data start in the third line. Assuming there are 200 points, the 'LastLine=' parameter is numbered 200. The last data line is then in the 200th line, counted from the first valid data line.

If the switch 'FileBegin' is set, the numbering of the 'LastLine=' entry refers to the first line of the file. According to the above example, 'LastLine=202' must be specified. The last valid data line is then line 202, counted from the start of the file.

If the switch 'EndOfFile' is set, the entered number for 'LastLine=' cuts off the last lines of the file.

## Select Column Number

In the case there are multiple columns present in your import data file, the Def\_Import-parameter **ColNo=** is used to select a specific column.

The default value is 'ColNo=1', i.e. the program reads the first and second column and processes it as vector data.

If you enter for example

```
Def_Import
...
ColNo=2
...
```

the program will read the second and third (if existing) column for the ordinate.

If you want to display only real ordinates, neglect that vector data are read in, and select 'Real' in the Bode type listbox of the Import Display Dialog.

## Point Separator

### Default

Usually each measurement point i.e. one pair of abscissa/ordinate-values is placed in one line. In this case your export device marks the end of line by the character pair CR+LF (= #13#10). This is the default case.

### Special case

There may be cases where your export device can not export data in this manner. Enter here the appropriate limiter-character, for example a comma (,) etc.

## Import Display Dialog

*Dialog: File/ Import/ Do Import...*

*Key: Ctrl + R*



Once a Def\_Import is entered move the cursor in its lines and issue the command 'File/Import/Do Import...' or click on the button with the bottle to load the data.

When the Def\_Import - dialog is open, the import uses the settings of that dialog. Otherwise the settings of that particular Def\_Import definition is used in which lines the text-cursor is located.

Initially the data are displayed in the Import Display Dialog where necessary adjustments can be carried out before the curve is copied to any diagram as a guest graph or a new diagram is created.

If an error occurs and the data can not be displayed in a diagram a list appears instead, which shows the data as text which might help in finding the error.

The Import Display Dialog is not available when importing data for tools. Instead the curve is directly displayed in the diagram of the tool-dialog, see Import for Tools.

### Auto range estimation

On loading the Import Display Dialog estimates your data and, initially, selects the units and adjusts the range for you. It is always possible to edit the range and the units manually. This automatic clings to the identifier of the Def\_Import definition. When you reload from the same Def\_Import, the range will not be modified.

## Bode type, output curve

This list controls the format of a frequency response or any scalar format, like time responses, directivity or phase curves.

The contents and range of the list depends on the operating modes, called complex mode and non-complex mode and on the type of data, whether there are scalar or vector data.

### Non complex mode

In this default mode this list switches only the units of the diagram, but the values of the import data are not modified in format. For example: When your import data having the format 'level and phase' and you select from the list 'Real+Imaginary' then the curve is unaltered but the ordinate changes. When you want to post-process your data later using the operators of the 'Calc'-menu for instance, you should adjust the Bode type. Once set, the units and range can be adjusted manually if necessary, in this dialog or later.

### Scalar values

Time domain                      Abscissa in seconds, ordinate estimated.

Lp                                      Abscissa in Hertz, ordinate in dB.  
Internally a flag is set to distinguish sound pressure from level.

Level	Abscissa in Hertz, ordinate in dB.
Real	All purpose mode for scalar data. Initially abscissa in Hertz or in Degree, ordinate estimated.
Phase	Abscissa in Hertz, ordinate Degree.

### Vector values

Lp (Phase)	Abscissa in Hertz, ordinate in dB, phase in degree Internally a flag is set to distinguish sound pressure from level which is important when iFFT might be applied later in order to get correct levels.
Amplitude (Phase)	Abscissa in Hertz, ordinate estimated, phase in degree.
Level (Phase)	Abscissa in Hertz, ordinate in dB, phase in degree.
Real	Abscissa in Hertz, ordinate estimated. Only the first component is displayed as a real value curve.
Imaginary	Abscissa in Hertz, ordinate estimated. Only the second component is displayed as a real value curve.
Real + Imaginary	Abscissa in Hertz, ordinate estimated.
Phase	Abscissa in Hertz, ordinate Degree. Only the second component is displayed as a phase curve.

### Complex mode

In the complex mode the Def\_Import definition has the keyword 'Complex' in its lines. In this mode all the import stages uses the 'real and imaginary' data format assumed to be a peak value.

At the input stage of this chain the format can be converted with the help of the ordinate formula system.

At the output stage the Bode type list selects the format of the complex data, i.e. in this case not only the units and the range but also the data are modified. The following list entries convert the complex import value or the result of the ordinate formula system:

Lp (Phase)	Abscissa in Hertz, ordinate in dB, phase in degree. The level Lp of the rms-sound pressure is calculated with respect to the threshold of hearing $L_p = 20 \cdot \log(\sqrt{0.5} p_{\text{peak}} / p_0)$ with $p_0 = 20 \mu\text{Pa}$ .
Amplitude (Phase)	Abscissa in Hertz, ordinate estimated, phase in degree.
Level (Phase)	Abscissa in Hertz, ordinate in dB, phase in degree. The level is calculated with respect to $0\text{dB} = 1$ peak value, whatever the actual unit might be: $L = 20 \cdot \log( y )$ .
Real	Abscissa in Hertz, ordinate estimated. Only the real part is displayed as a real value curve.
Imaginary	Abscissa in Hertz, ordinate estimated. Only the imaginary part is displayed as a real value curve.
Real + Imaginary	Abscissa in Hertz, ordinate estimated.
Phase	Abscissa in Hertz, ordinate Degree. The phase is calculated and displayed as a real valued curve.

## Diagram settings

### Lin / log abscissa

The data row is displayed with a linear or logarithmic abscissa-grid. Depending on the original data points linear interpolation is applied. When your data have an abscissa with zero or negative values only a linear abscissa is possible.

When you intend to create a polar diagram subsequently (see below) then the 'linear abscissa' setting is necessary.

### Units, legend...

Opens the sub-dialog to edit the units and the legend.

### Range...

Opens the sub-dialog to edit the range of the diagram.

### Def\_Import...

Opens or activates the Def\_Import - dialog.

## Reload / from Import Script / from Dialog...

When the Def\_Import dialog is open, the import uses the settings of that dialog. Otherwise the settings of that particular Def\_Import-definition is used in which lines the text-cursor is located.

## Copy graph

Copies the curve into the clipboard from where it can be pasted into any other diagram.

Drag and drop works also here. For this click on the legend of the curve without releasing the mouse-button and move the mouse to the target diagram.

Double clicking on the legend copies the curve into the same diagram (Ins-key works also after 'Copy graph'). This might be practical when you want to compare measurement curves on the fly.

## Close and diagram

The Import Display Dialog closes and a usual diagram window with the import graph is created. At this stage the connection between the original data and the curve is cut. The only way of re-import is via the Def\_Import definition. Depending of the selected Bode type, the calculations of the 'Calc' menu can be applied like Fourier transformation, etc.

To create polar diagrams the 'Polar' checkbox must be switched on (see below).

### Polar

The Import Display Dialog can only display Cartesian diagrams. Nevertheless it is possible to create polar plots from import data. Before issuing the button 'Close and diagram' (see above) check this box.

The 'Polar' checkbox is only active when range and units are properly set. Note also, that in polar diagrams only sound pressure levels (Lp) can be displayed, i.e. no phase.

The 'Polar'-checkbox is only active when

Abcissa = linear and

Bode type = Lp or Lp (Phase) and

Abcissa range is within  $-180^{\circ}$ ... $+180^{\circ}$ .



# Examples

## Measuring the electrical impedance

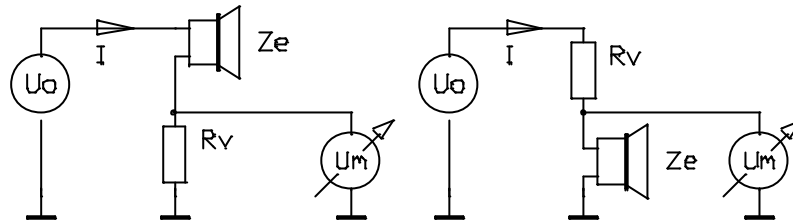


Fig. 42 Measurement circuits for impedance measurements, left type A, right type B

The measurement circuit in Fig. 42 can be used for measuring the electrical impedance of loudspeakers. **Uo** is the source voltage of the frequency generator. **Um** is the measured voltage.

### Vector measurement

The easiest and most exact way is to measure the voltage drop **Um** across **Rv** or **Ze** by a vector measurement device and export the spectrum in real and imaginary format. In this case both types of circuits of Fig. 42 are equally suitable. The resistor **Rv** should be in the range of  $|Z_e|$ .

Then the impedance is for circuit type A:

$$Z_e = R_v \cdot (U_o / U_m - 1)$$

And for the circuit type B:

$$Z_e = R_v / (U_o / U_m - 1)$$

The equivalent `Def_Import` definition is switched to the complex calculation mode and an ordinate formula system (for type A) must be specified. The result of the last formula '**Ze**' is then the new ordinate value in units 'Ohm'.

```
Def_Import 'Imp_Z_A'
  Filename='Z_A.imp'
  Legende='Impedance, complex'
  Complex
  For1={ Rv=100; Uo=1; Ze=Rv*(Uo/U_m - 1); }
```

For the type B circuit of Fig. 42 the following definition can be entered in the import script:

```
Def_Import 'Imp_Z_B'
  Filename='Z_B.imp'
  Legende='Impedance, complex'
  Complex
  For1={ Rv=100; Uo=1; Ze=Rv/(Uo/U_m - 1); }
```

These two definitions could be part of your import script and easily be used as a generic import definition of impedance measurements. In this case, export from your measurement device into the file `Z_A.imp` or `Z_B.imp` file, respectively. Then you would move the script cursor into the lines of the appropriate `Def_Import` definition and issue 'File/Import/Do Import'.

### A variant

A further example demonstrates what is to do when the measurement device exports the vector points in level and phase format where 0dB corresponds to 1V peak. In this case the above formula would lead to wrong results since the ordinate variable would contain a level in the real part and a phase in the imaginary part. Therefore a conversion function must be inserted at the beginning of the formula system:

```
...
Complex
For1={ Uz=LPToRI(Um); Rv=100; Lo=0;
        Uo=10^(Lo/20); Ze=Rv/(Uo/Uz - 1); }
```

The parser-function LPTpRI() converts a voltage level given in dB and a phase given in Degree into the real and imaginary format. For convenience use the Def\_Import-Dialog (Def-menu) and there the sub-dialog 'Ordinate formula' to edit the formula system.

In this example the source voltage is also expressed as a level relating to 1V peak (Lo).

### Scalar measurements

To some degree of approximation it is also possible to measure only the modulus of the impedance with the circuits of Fig. 42. In this case we have to distinguish high impedance and low impedance measurements.

The high impedance measurement, for example for piezo loudspeakers, is set up with the type A circuit where the resistance Rv is much smaller then the minimum value of the |Ze| curve and is called constant voltage measurement. When  $R_v \ll |Z_e|$  the magnitude of the impedance can be approximated by

$$Z_e \approx R_v U_m / U_o$$

The low impedance measurement, for example for dyn. drivers, is set up with the type B circuit where the resistor is much larger then the maximum value of the |Ze| curve and is called constant current measurement. When  $R_v \gg |Z_e|$  the magnitude of the impedance can be approximated by

$$Z_e \approx R_v U_o / U_m$$

Note, that both measurments can lead to serious errors, especially when the point of resonance has to be investigated, since here the impedance can rise (or drop) to extreme values. Further processing on the import curve as, for example the determination of the driver parameter, will then calculate unusable values.

When your exported curve is of a scalar type and you apply the above mentioned approximations then the non-complex mode is usually used. A Def\_Import definition for high impedance scalar measurements could have the following form:

```
Def_Import 'Imp_Sc_A'
  Filename='ZSC_A.imp'
  Legende='Impedance, Scalar (A)'
  For1={ Rv=1; Uo=1; Ze=Rv*Um/Uo; }
```

For low impedance scalar measurements it could be:

```
Def_Import 'Imp_Sc_B'
  Filename='ZSC_B.imp'
  Legende='Impedance, Scalar (B)'
  For1={ Rv=1000; Uo=1; Ze=Rv*Uo/Um; }
```

## Measuring the sound pressure level

This example is not intended to cope with all the problems involved in the measurement of the sound pressure but shall demonstrate some of the possibilities you have using the import features.

### Vector spectrum measurements

Many audio spectrum analyzers export the vector measurement of the sound pressure in the format Lp and phase where the phase is usually already freed from the delay due to the traveling sound wave from the source to the receiver. Using this starting point there are two alternatives to import:

### Non-complex mode

In this mode the program displays the data as they are. The first component is in this case the sound level in dB and the second component is the phase in Degree. Def\_Import could be written as follows:

```
Def_Import 'SPL1'
  Filename='Test1.spl'
  Legende='Sound pressure, r=1m, Uin=2.83V'
```

You might modify the level and phase with the help of the ordinate formula system:

```
Def_Import 'SPL1'
  Filename='Test1.spl'
  Legende='Sound pressure, r=1m, Uin=2.83V'
  For1={ Um + 6; } | Level + 6dB
  For2={ -Um } | Phase inversion
```

### Complex mode

In this mode the program processes the data in the real and imaginary format and you can choose the way the curves are displayed, i.e. as Lp and phase or as real and imaginary, etc.. The incoming Lp and phase have to be converted to the real and imaginary format within the ordinate formula system.

```
Def_Import 'SPL_2'
  Filename='Test2.spl'
  Legende='Sound pressure, r=1m, Uin=2.83V'
  Complex
  For1={ LpPToRI(Um) }
```

Here the conversion function is LpPToRI() which converts the sound pressure level and the phase (in Degree) to the real and imaginary format.

You might want to compensate for a delay of  $z_0=1m$ . In this case extend the formula system to:

```
Def_Import 'SPL_2'
  Filename='Test2.spl'
  Legende='Sound pressure, r=1m, Uin=2.83V'
  Complex
  For1={ c=344; z0=1; k=2*pi*f/c;
          pm=LpPToRI(Um); p=pm*exp(-jk*z0) }
```

### Fourier transformation

Measured curves can be Fourier transformed using the 'Calc/ Time to Spectrum' or 'Calc/ Spectrum to Time' functions of AkAbak. Before these functions can be applied the measured curve has to be correctly imported. When you intend to transform from the frequency domain to the time domain your import stream should be a vector curve.

As an example, the procedure of importing a time curve shall be demonstrated. If you have, for example, a MLSSA time file exported to 'dodect.dat' then the definition would be:

```
Def_Import 'SPL_3'
  Filename='Dodect.dat'
  Legende='Sound pressure, time impulse'
  AbscFor={ t/1000 }
```

Since the import data is scalar the non-complex mode is appropriate. In this example the abscissa formula system 'AbscFor={ t/1000 }' is used to adjust the time-abscissa which is exported by MLSSA in milli-seconds.

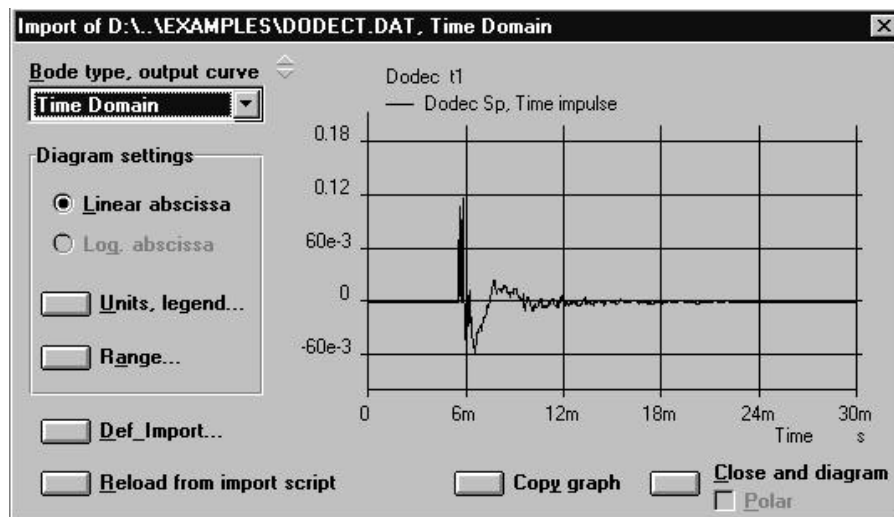


Fig. 43 Import Display Dialog

When you move the script-cursor into the lines of this definition and issue 'File/ Import/ Do Import', AkAbak displays the curve in the 'Import Display Dialog'. The 'Bode Type' list should have switched to 'Time Domain' automatically. Select this mode in this case whenever the import-estimator was wrong. The right settings for a subsequently applied FFT are displayed in Fig. 43. Press the Button 'Close and diagram' to generate a diagram window which is then the source for further processing.

# Definitions

*Menu: Def/*

The definition part comes in front of the first `System`. This part of the script is reserved for global definitions.

In all electroacoustic transducers, the details of the parameter are listed here. This definition precedes the first occurrence of the associated network element, usually at the start of the script. You can generally recognize the keywords of the definitions by the prefix `Def_`, for example `Def_Driver`.

Each definition is given an arbitrary, unique name (identifier). The associated network element has a parameter whose identifier is called `Def=`. `Def=` is followed by the name of the definition whose parameter values you intend the network element to use, for example: `Def='B1'`.

We have found this method of input useful in practice: Firstly, the network descriptions are often very complex; the extensive list of the driver parameter is therefore given separately to avoid cluttering.

Secondly, any number of network elements can refer to the same definition. In this case they all have the same parameter, but different network and radiation positions.

Furthermore, it is easier to test the characteristic of different drivers in the constellation if they are described centrally. For this purpose you enter the definitions of all drivers to be tested at the start of the script and then only change the names.

## Filter elements in the definition part

`Filter` elements may be located in the definition part. All `Filter` elements that are listed before the first `system` keyword are multiplied together. The input voltage `Uin` which can be specified in the control dialogs is weighted with the result. The filter thus has a global effect. The output voltage is the source voltage of the succeeding systems with their networks and filter structures. The filters of the definition part cannot be networked with a `feedback` formula.

## Def\_Import

In this definition interpretation and processing information for import files are specified and saved. Please see chapter `Import/ Def_Import` for a detailed description.

## Def\_Const

*There is no input dialog for this definition.*

In the definition `Def_Const` you can enter constants or entire formula systems and use them in formulae within the subsequent script. Formulae are evaluated during the interpretation of the script shortly before the simulation itself. Their result is then assigned to the respective identifiers in the definitions and network elements.

`Def_Const` allows you to control certain parameter centrally in the script, be they the temperature, position or filter parameter, or much more.

The syntax of the formula is described in chapter: `Introduction/Formula Parser`. The keyword `Def_Const` is followed by an opening brace (curly bracket). The formulae are separated by a semicolon. The brace is then closed.

After the interpreter has processed `Def_Const`, each formula name corresponds to a numerical value. In the following formulae, this numerical value is used wherever the name appears again.

Note that no conversion of units may take place within the formulae. For example, it is inadmissible to enter 1 mm. The best policy is to use only SI units throughout, 1 mm then corresponds to 1e-3.

### ☑ Example

Assume that two tweeters are located symmetrically on a triangular projection, as the diagram in Fig. 44 attempts to show.  $h$  is the height in the  $z$  direction.  $b$  is the width of the base. The loudspeaker is in the center in each case. The origin of the baffle/coordinate system is in the center below the projection. First the constants  $z$ ,  $x$  and  $H\text{Ang}$  are evaluated.  $H\text{Ang}$  is the horizontal mounting angle in degrees, that is why the function  $\text{Deg}()$  is provided, which converts radians into degrees.

The two speaker elements, with the names 'right' and 'left' use only the constants  $x$ ,  $z$  and  $H\text{Ang}$  in their inline formulae.

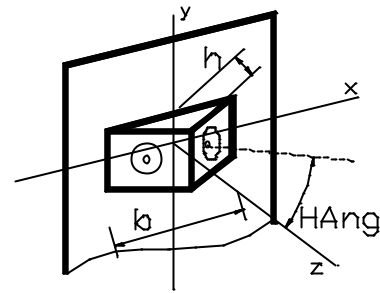


Fig. 44 Diagram for Def\_Const example

```
Def_Const
{ b = 30e-2; h = 10e-2; z = h/2; x = b/4;
  HAng = Deg(arctan(x/z));
}

...

System 'S1'

Speaker 'right' Def='K1' Node=1=0
  x={x} z={z} HAngle={HAng}

Speaker 'left' Def='K1' Node=1=0
  x={-x} z={z} HAngle={-HAng}
```

### ➔ Note

In the filter element, the coefficients are also evaluated with the aid of the formula parser. The names of the coefficients are reserved ( $b0=$ ,  $b1=$ , ...  $a0=$ ,  $a1=$ ,...) Their values have to be assigned within the filter elements. It is not sufficient to enter, for example,  $\{a0=5; a1=a0 + 1;\}$  in `Def_Const` but to leave it out in a filter element in which the entry is needed. In this case  $\{... a0=a0; a1=a1;\}$  should be given there.

## Def\_ListeningPoint

By means of this entry, the reference point of the sound radiation is shifted. The analyses 'Sound Pressure', 'Directivity Pattern' and 'Acoustical Power' determine the sound pressure at a measurement point in the room which is located at a specific distance from and a specific angle with respect to the origin of the baffle coordinate system. The origin of the baffle coordinate system is  $x=0$ ,  $y=0$ ,  $z=0$ . That means a radiating element, for example a `Radiator`, lies at the origin of the baffle if its position entries have these coordinates.

Assume the bass loudspeaker of a loudspeaker system was first designed and its position set to (0,0,0). A mid-range speaker and tweeter have been successively added. If then for instance the origin shall be congruent with the position of the tweeter, the coordinates of the tweeter thus have to be entered at the `Def_ListeningPoint` element. The distance and angle of the modes of analysis now relate to this new point.

Another feature of `Def_ListeningPoint` is to shift the phase of radiation or in other words to compensate for the delay from a radiator to the listening point. For the phase calculation `AkAbak` takes the origin of the baffle coordinate system as reference, i.e. a radiator at  $r=(0,0,0)$  needs no further phase compensation because it is done by `AkAbak` automatically. But there are cases where the phase compensation must be modified. Then use the parameter `zo=` or `to=` of `Def_ListeningPoint`.

Example: The origin of the co-ordinate systems of the loudspeaker is shifted 10 cm upwards.

```
Def_ListeningPoint
x=0 y=10cm z=0
```

Example: Additionally phase compansation

The distance from the origin to the radiator is  $z=-10\text{cm}$ . The phase response shall be freed from phase rotation due to the delay.

```
Def_ListeningPoint
zo=-10cm
```

## Parameter

$x = \dots \text{m}$	Horizontal position of the new origin on the baffle. Positive values point to the left and negative values to the right, seen from the baffle. Unit: meter [m].
$y = \dots \text{m}$	Vertical position of the new origin on the baffle. Positive values point upwards, negative values downwards.
$z = \dots \text{m}$	Position of the new origin with respect to the baffle. positive values point forwards, negative values backwards.
$zo = \dots \text{m}$	Additional time delay expressed as distance: $to=zo/c$ ; with $c$ : sound velocity. Unit: meter [m].
$to = \dots \text{s}$	Additional time delay of radiation. Unit: meter [m].

### → Note

You do not need to modify the distances and angles in the `Def_Reflector` element. The program automatically adapts the coordinates.

## Def\_Reflector

*Dialog: Def/ Def\_Reflector*

This definition installs reflectors located close to the loudspeaker. Using `Def_Reflector`, you can estimate the effect of walls on the sound power level, the directivity characteristic and the acoustic power. The reflectors influence the radiation as well as the radiation impedance of a radiator.

Bass loudspeakers are particularly subject to dramatic differences in reproduction characteristics when reflectors are present. Compared with the other room acoustics, the reflections of the speaker from nearby walls have such a strong influence because they feed back directly to the source of the sound.

Since not only the distance, but also the positional angle of the loudspeaker with respect to the wall can be entered, a loudspeaker can be accurately positioned to generate the desired reproduction characteristic at a given measurement point in the room.

Walls, the ceiling and floor act as reflectors. You can enter a maximum of three reflectors. All walls are perpendicular to one another and reflect the sound without losses and frequency independent.

You use keywords to communicate the particular wall arrangement to the program:

room corner at floor	<code>BottomCorner</code>
room corner at ceiling	<code>TopCorner</code>
horizontal edge	<code>HorizEdge</code>
vertical edge	<code>VertEdge</code>
single wall	<code>Wall=</code>

There is no keyword for the last-mentioned case. When you enter `Wall=`, you also enter the distance from the wall at the same time, in the other cases, you enter the distance depending on the mode.

If a `Def_Reflector` definition is to be effective, the `Reflection` keyword has to be entered for those radiation elements that are considered to be subject to reflection. That means that not all radiators have to be subject to reflection. Particularly for tweeters, the walls usually lie outside the so-called diffuse-field distance (see below), and the walls usually have a more intensive attenuation effect at high frequencies, so that no reflectors are provided here by omitting the keyword `Reflection`.

The program assumes by default that all radiators are located in a baffle of infinite extent, i.e. they radiate into the so-called  $2\pi$ -sr.-room and no radiation takes place to the rearward side of the baffle. If an element is subject to reflection it cannot be mounted in an infinite baffle. The element is rather located in a finite baffle, around which the sound diffracts. For elements that are involved in the reflection, additional parameter of sound diffraction has to be entered (`WEdge=`, `HEdge=` etc.). The reflection is then correctly calculated for all possible mounting positions. Radiators may also be located on the other side of the cabinet, for example (see chapter 'Introduction-Radiation environment')

By default the specified reflecting wall does not absorb. The resulting simulated curves tend to display an exaggerated ripple. In reality the interference effects are much more smeared due to absorption, to the finite size of the reflectors and due to room modes.

Optional it is possible to add an absorption coefficient to the walls. It is advisable to use this parameters with care since a strong smearing effect often covers the principal resonances.



### Examples

1. The reflector consists of a wall. The box is standing on the floor, since `VAngle=-90°`. The distance of the origin of the baffle coordinate system to the floor is 30 cm.

```
Def_Reflector Wall=30cm
HAngle=0 VAngle=-90°
```

2. The reflector consists of two walls. The loudspeaker hangs from a vertical room edge. It is rotated 45° away from the left wall horizontally. The loudspeaker is inclined downwards by 30°.

```
Def_Reflector VertEdge
Left=1.0m Right=1.0m
HAngle=45.0° VAngle=-30.0°
```

3. The reflector consists of two walls. The loudspeaker is located in front of a horizontal room edge. The loudspeaker is inclined backwards by 20° and 30° to the right. The walls reflecting the sound energy only by 50% at all frequencies.

```
Def_Reflector HorizEdge
Bottom=30.0cm Top=1.0m
HAngle=30.0° VAngle=20°
AbsorbCoeff=0.5
```

4. The reflector consists of three walls. The loudspeaker is hanging from the ceiling inclined 45° to the right and 45° downwards. The walls reflecting the sound energy only by 50% at frequencies above 100Hz. Below there is a total reflection.

```
Def_Reflector TopCorner
Left=1.0m Top=1.0m Right=1.0m
HAngle=45.0° VAngle=-45.0°
AbsorbCoeff=0.5 AbsorbFrequ=100.0Hz
```

5. The reflector consists of three walls. The loudspeaker is inclined backwards by 20° and 45° to the right. The walls reflecting the sound energy only by 50% at frequencies below 100Hz. Above 100Hz total the walls perform total absorption.

```
Def_Reflector BottomCorner
Left=1.0m Bottom=30cm Right=1.0m
```



HAngle=45.0° VAngle=20°  
 AbsorbCoeff=0.5 AbsorbFrequ=100.0Hz AllAbsorb

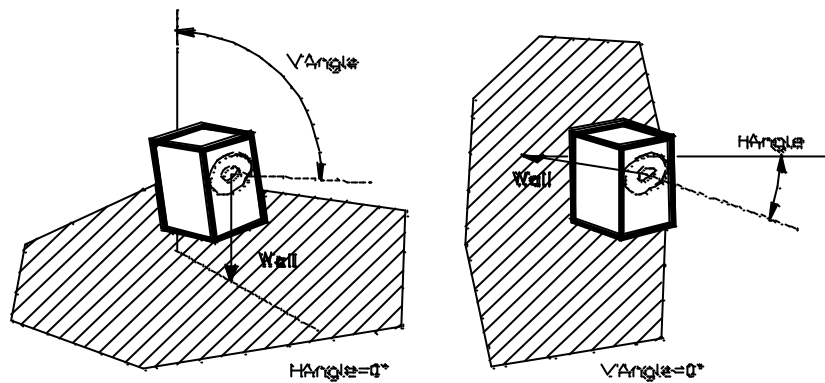


Fig. 45 Reflectors with one wall (Wall=)  
 left: vertical inclination  
 right: horizontal inclination

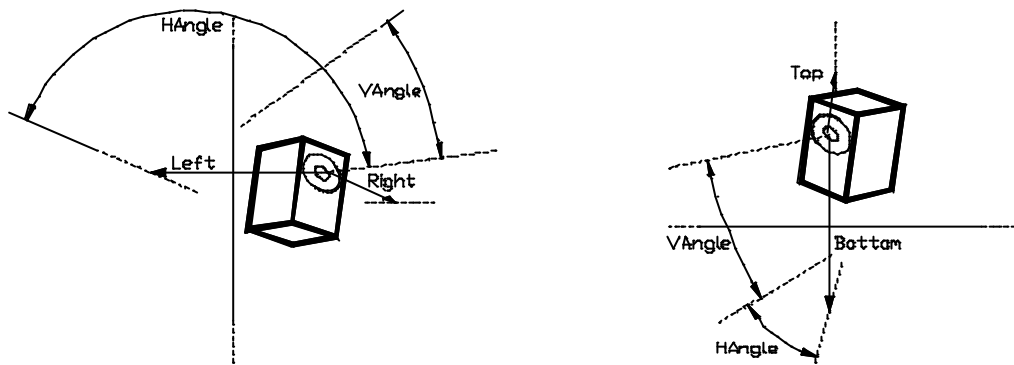


Fig. 46 Reflectors with two walls (left VertEdge, right HorizEdge)

## Parameter

Def\_Reflector

Keyword

Wall=...m

The reflector is formed by only one wall (Fig. 45). No other keyword need be entered. The loudspeaker may be rotated either horizontally with HAngle= or vertically with VAngle= in front of the wall, but not with both. That means one of the two angles has to be set to zero or not given. If both angles are equal to zero, the wall lies behind the speaker.

*Example:* For a loudspeaker standing vertically on the floor VAngle=-90°.

*Example:* For a loudspeaker whose axis is parallel to the left-hand wall (plan view) HAngle=-90°.

*Range:* HAngle= and VAngle= each from -90°...90°.

The value of Wall= describes the distance of the origin of the baffle coordinate system from the wall.

Unit: meter [m] or inch [in].

VertEdge

Keyword. If two walls form the reflector (Fig. 46, left). One wall is located to the left of the loudspeaker, the other to the right of it. The loudspeaker thus stands in front of a vertical room edge.

The distance from the left-hand wall (plan view) is entered using Left= and that to the right-hand wall using Right=.

The default settings HAngle=0 and VAngle=0 mean that the loudspeaker is located plan with the left-hand wall (plan view).

*Example:* If the loudspeaker whose axis is parallel to the right-hand wall, then

HAngle=90°.

*Range:* HAngle=0...90° and VAngle=-90°...90°.

### HorizEdge

Keyword, if two walls form the reflector (Fig. 46, right). One wall is located below the loudspeaker and the other behind it. The loudspeaker is thus standing in front of a horizontal room edge.

The distance from the floor is entered using Bottom= and that to the rear wall using Top=.

The default settings HAngle=0 and VAngle=0 mean that the loudspeaker is standing perpendicular on the floor and the edge is parallel to the x-axis of the baffle coordinate system.

*Example:* If the loudspeaker is lying on its back, so to speak, or is suspended in a room edge below the ceiling, then VAngle=90°.

*Range:* HAngle=-90°...90° and VAngle=0...90°.

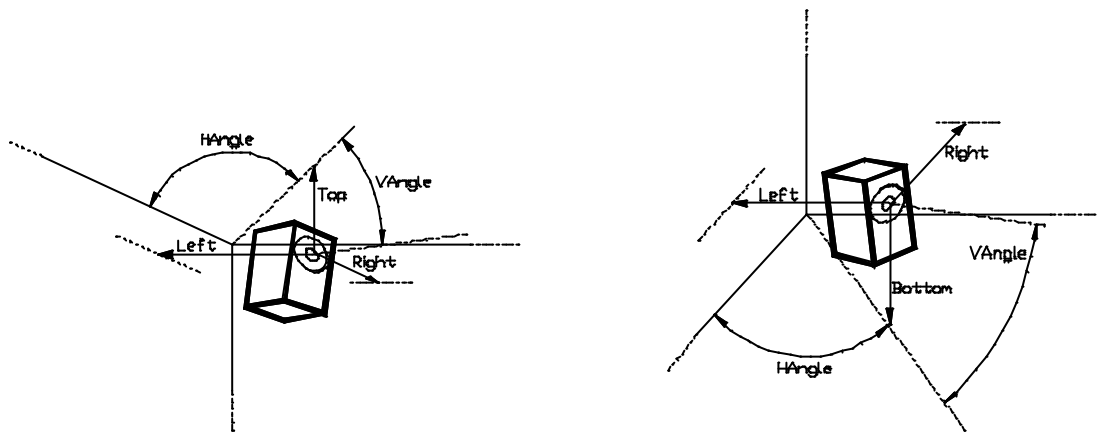


Fig. 47 Reflectors with three walls (left TopCorner, right BottomCorner)

### TopCorner

Keyword if three walls from the reflector (Fig. 47, left). One wall is located to the left of the loudspeaker, the other to the right and the third wall above the loudspeaker. The loudspeaker is thus suspended from the ceiling in a room corner.

The distance from the left-hand wall (top view) is entered using Left= and that to the right-hand wall using Right=. Top= indicates the distance from the ceiling.

The default settings HAngle=0 and VAngle=0 means that the loudspeaker is hanging with its axis parallel to the left-hand wall and the ceiling.

*Example 1:* If the loudspeaker is suspended with its axis parallel to the right-hand wall, then HAngle=90°.

The loudspeaker is inclined downwards with VAngle=, which in this case is always negative or zero.

*Range:* HAngle=0...90° and VAngle=0...-90°.

### BottomCorner

Keyword if three walls from the reflector (Fig. 47, right). One wall is to the left of the loudspeaker, the other to the right and the third wall is below the loudspeaker. The loudspeaker is thus located below the loudspeaker. The loudspeaker is thus located in a room corner on the floor.

The distance from the left-hand wall (plan view) is entered using Left= and that from the right-hand wall using Right=. Bottom= indicates the distance from the floor.

The default settings HAngle=0 and VAngle=0 mean that the loudspeaker is standing with its axis parallel to the left wall and the floor.

*Example:* If the loudspeaker axis is parallel to the right-hand wall, then HAngle=90°.

With VAngle=, the loudspeaker is inclined upwards.

*Range:* HAngle=0...90° and VAngle=0...90°.

Bottom=...m	Distance of the origin of the baffle coordinate system from the floor. Only applies to BottomCorner and HorizEdge. Unit: meter [m] or inch [in].
Top=...m	Distance of the origin of the baffle coordinate system from the ceiling. Only applies to TopCorner and HorizEdge. Unit: meter [m] or inch [in].
Left=...m	Distance of the origin of the baffle coordinate system from the left-hand wall (top view). Only applies to BottomCorner, TopCorner and VertEdge. Unit: meter [m] or inch [in].
Right=...m	Distance of the origin of the baffle coordinate system from the right-hand wall (top view). Only applies to BottomCorner, TopCorner and VertEdge. Unit: meter [m] or inch [in].
HAngle=...°	Angle of rotation of the loudspeaker in the horizontal plane. It is rotated about the origin of the baffle coordinate system. One of the legs of the angle is the z-axis of the coordinate system, the other is determined by the type of reflector (see above). HAngle= is always positive is the loudspeaker is rotated to the right. (In other words: Rotation to the right in plan view). <i>Example:</i> In the corner (BottomCorner, TopCorner) or in front of a vertical edge (VertEdge), HAngle=0...90°, i.e. from the left-hand to the right-hand wall. <i>Example:</i> If the loudspeaker is located in front of a horizontal edge (HorizEdge) or a wall (Wall=), the loudspeaker is rotated to the left using HAngle=0...-90° and to the right using HAngle=0...90°. Unit: degree [°] or [Deg].
VAngle=...°	Angle of rotation of the loudspeaker in the vertical plane. It is rotated about the origin of the baffle coordinate system. One of the legs of the angle is the z-axis of the coordinate system, the other is determined by the type of reflector (see above). VAngle= is always positive is the loudspeaker is rotated upwards. <i>Example:</i> Suspended in the corner (BottomCorner) or in front of a horizontal edge (HorizEdge), VAngle=0...90°, i.e. from the left-hand to the right-hand wall. <i>Example:</i> If the loudspeaker is located in front of a vertical edge (VertEdge) or a wall (Wall=), the loudspeaker is rotated downwards using VAngle=0...-90° and upwards using VAngle=0...90°. Unit: degree [°] or [Deg].
AbsorbCoeff=	The absorbtion coefficient which is in the range: 0...1. With an AbsorbCoeff=0 there is no absorption and full reflection (default value). When AbsorbCoeff=1 no reflection takes place, i.e. all energy is absorbed by the walls.
AbsorbFrequ=	This frequency specifies optional the frequency range where absorption takes place. Depending on the switch 'AllAbsorb' two modi are available: When 'AllAbsorb' is not specified then absorption is active in the frequency range above 'AbsorbFrequ'. Below, no absorption or total reflection takes place. When 'AllAbsorb' is specified then the absorption coefficient is active in the frequency range below 'AbsorbFrequ'. Above, total absorption or no reflection takes place. Unit Hertz [Hz]
AllAbsorb	Switches the operating frequency range of 'AbsorbFrequ='.

## Diffuse-field distance

The effect of a wall on radiation and radiation resistance decreases with distance. Above a certain distance, corresponding to the so-called diffuse-field distance, we can neglect the effect of a wall.

In closed rooms, a sound field builds up. Its average level depends not only on the acoustic power of the sound source, but also on the reverberation time of the room. As a guideline to which distance from the sound source the sound field is determined more by the loudspeaker than by the room acoustics, the so-called diffuse-field distance is defined. At a distance from the loudspeaker equal to the diffuse-field distance, the acoustic free-field sound pressure is equal to the average sound level in the listening room.

$$r_H \approx 0,057 \cdot \sqrt{\frac{V}{T_N}}$$

where

$r_H$ : diffuse-field distance  
 $V$ : volume of the listening room  
 $T_N$ : reverberation time of the listening room  $T_N$  is the time during which the average sound level in the room falls by 60dB [Zwi]

In normal rooms and in the bass range, the diffuse-field distance is 1 to 3 m. Applied to the reflectors, that means that we only need to consider those walls that lie within the diffuse-field distance, i.e. within a three-meter radius.

### Comparison with the diagram option in the Def\_BassUnit dialog 'Bassunit Calculator'

By contrast with Def\_Reflector, the diagram option of the Def\_BassUnit dialog calculates the effects of walls with the aid of a closed formula [Wat]. This formula functions for the acoustic power of a single radiator, which is actually displayed there. Unfortunately, the enclosure shapes, mounting angles, and the interplay of several radiators cannot be taken into account there.

## Def\_OpAmp

*Dialog: Def/ Def\_OpAmp*

Def\_OpAmp is the definition part for the network element OpAmp. The set of parameter is the same for both declarations but the OpAmp element has additionally the node numbers. Please refer to chapter E-Net/ OpAmp for details. The declaration of an opamp is much more flexible then, for example, of a transducer. There are several ways of parameter specification with an opamp:

### Using the definition

When you use a lot of OpAmp's of the same type it is more convenient to enter the opamp-parameter only once in the definition Def\_OpAmp and relate to it with the help of the Def='...' parameter. For example, the created network list of the active filter synthesis tool uses this technique. For example:

```
Def_OpAmp  'Op1'
  vo=1e5  Rg=100ohm  Rin=100kohm  ft=3MHz

System  'S1'
...
OpAmp  'Op11'  Def='Op1'  Node=100=101=101
OpAmp  'Op12'  Def='Op1'  Node=200=201=201
...
```

### Only as network element

All parameters are specified within the network element OpAmp. For example:

```
OpAmp  'Op11'  Node=100=101=101
      vo=1e5  Rg=100ohm  Rin=100kohm  ft=3MHz
```

### Mixed specification

In some cases it is very practical to be able to overriding some of the parameter of the Def\_OpAmp, for instance for testing purpose, etc. For example:

```
Def_OpAmp  'Op1'
      vo=1e5  Rg=100ohm  Rin=100kohm  ft=3MHz

System  'S1'
...
OpAmp  'Op11'  Def='Op1'  Node=100=101=101
      ft=100kHz  Rg=1kohm  |overriding ft and Rg
```

## Parameter

Def_OpAmp	Keyword
'...'	Identifier of definition.
vo=	Open-loop gain at very low frequencies (without feedback).
Rg=...ohm	Open-loop generator impedance, i.e. the internal impedance of the OpAmp output stage.
ft=...Hz	Optional. Transition frequency of the open-loop gain vo. At ft, vo=1
Rdiff=...ohm	Optional. Resistance between non-inverting and inverting input.
Rin=...ohm	Optional. Resistance of non-inverting and inverting input to ground.

## Def\_Transistor

*Dialog: Def/ Def\_Transistor*

Def\_Transistor is the definition part for the network element Transistor. The set of parameter is the same for both declarations but the Transistor element has additionally the node numbers. Please refer to chapter E-Net/ Transistor for details. The declaration of a transistor is much more flexible then, for example, of a transducer. There are several ways of parameter specification with a transistor:

### Using the definition

When you use a lot of Transistor's of the same type it is more convenient to enter the transistor -parameter only once in the definition Def\_Transistor and relate to it with the help of the Def='...' parameter. For example:

```
Def_Transistor  'Q1'
      h11e=3.5kohm  h12e=2e-4  h21e=270  h22e=32uS
      ft=200MHz  Ccb=3.5pF

System  'S1'
...
Transistor  'Q11'  Def='Q1'  Node=100=101=101
Transistor  'Q12'  Def='Q1'  Node=200=201=201
...
```

### Only as network element

All parameters are specified within the network element Transistor. For example:

```
Transistor 'Q11' Node=100=101=101
  h11e=3.5kohm h12e=2e-4 h21e=270 h22e=32uS
  ft=200MHz Ccb=3.5pF
```

### Mixed specification

In some cases it is very practical to be able to overriding some of the parameter of the Def\_Transistor, for instance for testing purpose, etc. For example:

```
Def_Transistor 'Q1'
  h11e=3.5kohm h12e=2e-4 h21e=270 h22e=32uS
  ft=200MHz Ccb=3.5pF

System 'S1'
...
Transistor 'Q11' Def='Q1' Node=100=101=101
  h21e=100 |overriding h21e
...
```

## Parameter

Def_Transistor	Keyword
'...'	Identifier of definition.
h11e=...ohm	Differential base-emitter-resistance rBE. In the case of the FET and the valves, h11e is symbolically set to zero or not specified.
h12e=...	Optional. Voltage reaction Ar.
h21e=...	Differential current gain of transistors or forwards transconductance g standardized to one siemens for FET's and valves.
h22e=...S	Optional. Differential output conductance.
ft=...Hz	Optional. Transistors: transition frequency of the differential current gain h21e. FET's, valves: transition frequency of the transconductance g (=h21e).
Ccb=...F	Optional. Collector-base-capacitance. In the case of FET, Ccb represents the value of the drain-gate capacitance and in the case of the valves, the value of the anode-grid capacitance.

## Def\_Element (DLL)

*Dialog: Def/ Def\_Element*

Def\_Element is the definition of a component which transfer functions are implemented in a user-made Windows Dynamic Link Library (DLL). In other words, with the help of this interface you can add your own

specific network-model. The model may be as complex as you want. Use any Windows code compiler for C, Pascal or the like to produce the DLL.

The Def\_Element definition specifies the file name of the DLL and an optional string which contains entries you might want to pass to your element.

The 'Element' component inserts your model in the network as often as needed. 'Element' may be a component with two to six poles depending on the definition in your dll (actually ten poles are possible but the input dialogs does not support it).

The specification of the DLL functions are documented in the example files shipped with the program package:

TestElem.pas                      Pascal unit implementing a simple model of an open capacitor specified by the parameter 'Distance=' and 'Area='.

ElemExpl.pas                      Pascal main unit to compile a Windows DLL.

This examples can be directly compiled using the 16Bit Delphi Pascal compiler from Borland. Since the Windows API is written in C the example code is very similar when written C.

### Example

Using the compiled version of ElemExpl.pas you can check the dll with the following example (ElemExpl.aks). 'Area=' and 'Distance=' are user defined parameters submitted as a string to the SetParameter() function of the dll (Area in m<sup>2</sup> and Distance in m).

```
Def_Element  'DF1'
  Filename='Elemexpl.dll'
  Param={Area=0.01 Distance=0.001}

System 'S1'
  Element  Def='DF1'  Node=1=0
```

### Short cut admittance matrix

The element you intent to implement must be expressed in terms of the so-called short cut admittance matrix [Yp]. The order n of this matrix is equal the number of ports of your sub-network.

$$I] = [Yp] \cdot U]$$

The components of [Yp] are the admittance functions  $y_{ij}$  driven by  $U_j$  at port j when all others are short cut:

$$y_{ij} = \left. \frac{I_i}{U_j} \right|_{U_{1..i,k..n}=0} \quad \text{with } i = 1 \dots n, j = 1 \dots n$$

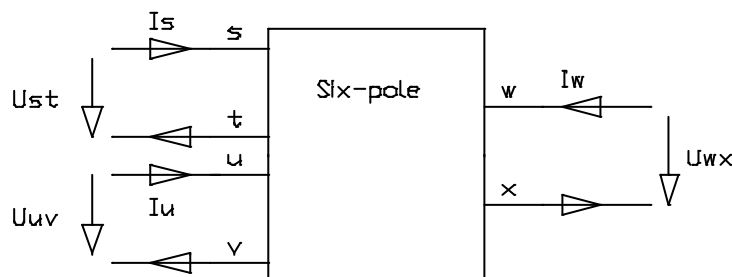


Fig. 48 Six-pole with port potentials and flows

Fig. 48 is a three port with six-poles. Ust, Uuv and Uwx are the test potentials and Is, Iu, Iw are the port flows. For this six-pole you would have to write down nine complex admittance functions in  $\omega$ :

$$[Y_p] = \begin{bmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \\ y_{31} & y_{32} & y_{33} \end{bmatrix}$$

with

$$y_{11} = \frac{I_1}{U_1} \Big|_{U_2=0, U_3=0} \quad y_{12} = \frac{I_1}{U_2} \Big|_{U_1=0, U_3=0} \quad y_{13} = \frac{I_1}{U_3} \Big|_{U_1=0, U_2=0} \quad \text{and so on}$$

Note, that the flows are symmetric across the ports, i.e.:  $I_t = -I_s$ ,  $I_v = -I_u$  and  $I_x = -I_w$ . In the case of coupled branches, i.e. odd number of poles, introduce grounded poles. For example, a three pole would be specified as a six pole ( $Node=s=0=u=0=w=0$ ).

The kernel function in your DLL-code to calculate the  $y_{ij}$ 's is called `Admittance()` where you load the  $Y_p[i, j]$  matrix for each angular frequency (see example files).

## Parameter

<code>Def_Element</code>	Keyword
<code>'...'</code>	Identifier of definition.
<code>Filename='...'</code>	Name of the DLL-file. By default the program searches in the <code>\zProgram\</code> directory. The extension 'dll' is appended automatically.
<code>Param={...}</code>	String of max. 4095 characters which is passed to your DLL-function <code>'SetParameter()'</code> . Note, that within the brackets your own syntax applies.

## Def\_Driver

*Dialog: Def/ Def\_Driver*

`Def_Driver` defines an electrodynamic driver with one voice coil. This definition describes the parameter of the four-pole network element `Driver`.

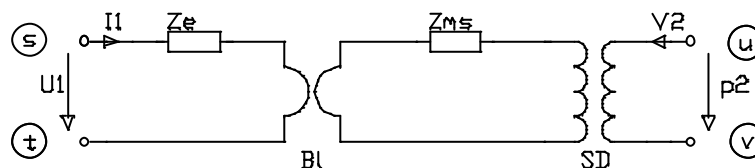


Fig. 49 Equivalent circuit diagram of the `Def_Driver` definition

Fig. 49 shows the equivalent circuit diagram of the electrodynamic driver as described in the chapter 'Introduction/ Transducer'. The poles **s** and **t** designate the electrical connection terminals of the driver. Poles **u** and **v** are at the acoustic side. Pole **u** represents the front side of the diaphragm and pole **v** its rear side.

### ☒ Examples

#### 1. Loudspeaker with conical diaphragm

The resonance is 50 Hz. The equivalent air volume of the mechanical compliance is  $V_{as}=10L$ . The mechanical quality has a value of 3 and the electrical quality a value of 1. The electrical impedance of the voice coil has a dc. resistance of 5 ohm and the voice coil inductance-factor is 1 mH. Eddy currents cause the real part to be doubled at  $f_{re}=10kHz$ . The reactance vs. frequency is modified by the factor  $ExpoLe=0.5$ . The diaphragm is



15 cm in diameter. It is conical. The diameter of the dust cap is 3 cm and is 2 cm deep on average. The control frequency of mass reduction is 1200Hz. By this you inform the program that at 1200Hz the diaphragm diameter has reduced to  $dD(f)=(dD+dD1)/2=9\text{cm}$ . The effective mass is diminished accordingly. These parameter describe, for example, a small bass loudspeaker with a relatively rigid suspension. Since nothing to the contrary has been specified, the program automatically inserts the keyword `Meas_Dipole`. This establishes that the entered parameter were measured under conditions of free radiation, i.e. the driver was not installed for the measurement.

```
Def_Driver    'Drv1'
  fs=50Hz  Vas=10L
  Qms=3  Qes=1  Re=5ohm  fre=10kHz  Le=1mH  ExpoLe=0.5
  dD=15cm  dD1=3cm  tD1=2cm  fp=1200Hz  |Cone
```

## 2. Dome tweeter

In this example, the driver is firmly mounted in the enclosure, whose volume is unknown. The measured parameter thus describe a loudspeaker in a sealed enclosure, in a similar manner to the `Speaker` element. To justify the type of design, take care with the associated `Driver` element to connect one of the output network nodes to ground and not, as is usually the case, to connect it to an `Enclosure` element or the like. Since the dome tweeter is connected backwards, and the driver was not installed for determining the parameter, you should enter the keyword `Meas_TubeEnd` (see above). The dome tweeter has a small horn of 3mm depth. The control frequency of mass reduction is  $fp=8000\text{Hz}$ .

```
Def_Driver    'dome'
  Meas_TubeEnd
  dD=2.5cm  tD1=0.7cm  t1=3mm  fp=8000Hz  |Dome
  fs=2.5kHz  Mms=1.1g
  Qms=1.5  Qes=1  Re=5ohm
```

## Parameter

Def_Driver	Keyword
'...'	Name of the definition. Indispensable here, since the <code>Driver</code> network elements with the identifier <code>Def='...'</code> refer to their definition. Range: any characters, but max. 20.

### Parameter of transducer see 'Introduction/ Transducer'

<code>fs=...Hz</code>	Resonance frequency.
<code>Mms=...kg</code>	Mass of diaphragm assembly inclusive airload.
<code>Vas=...m3</code> (alternative)	Equivalent compliance-volume.
<code>Cms=...m/N</code> (alternative)	Mechanical compliance of suspension.
<code>Qms=...</code>	Mechanical quality factor.
<code>Rms=...Ns/m</code> (alternative)	Mechanical resistance of driver suspension.
<code>Qes=...</code>	Electical quality factor.
<code>Bl=...Tm</code> (alternative)	Conversion factor.
<code>Re=...ohm</code>	Voice coil resistance factor.
<code>fre=...Hz</code> (optional)	Makes <code>Re</code> frequency dependent to simulate the influence of eddy currents.
<code>ExpoRe=...</code> (optional)	Controles the resistance of the voice coil.

$Le = \dots H$	(optional)	Voice coil inductance factor (with non-linear reactance).
$ExpoLe = \dots$	(optional)	Controls the reactance of the voice coil.
$Meas\_ \dots$		Keyword to indicate the radiation impedance of the parameter.
	$Meas\_Dipole$	no baffle (default value)
	$Meas\_TubeEnd$	no baffle, one side radiation
	$Meas\_Baffle1$	baffle, one side radiation
	$Meas\_Baffle2$	baffle, both side radiation
	$Meas\_DoNotModify$	The parameter are not modified.

### Parameter of diaphragm dimensions see 'Introduction/ Diaphragm Forms'

$dD = \dots m$		Diameter of diaphragm.
$WD = \dots m$	(alternative)	Width and height of rectangular diaphragm.
$HD = \dots m$		
$SD = \dots m^2$	(alternative)	Area of circular diaphragm.
$dD1 = \dots m$	(optional)	Diameter of inner diaphragm.
$SD1 = \dots m^2$	(alternative)	Area of inner diaphragm.
$tD1 = \dots m$	(optional)	Depth or height of diaphragm form.
$t1 = \dots m$	(optional)	Displacement of diaphragm within baffle.
$f_p = \dots Hz$	(optional)	Mass and area reduction control frequency.
$Diffuse = \dots \%$	(optional)	Directivity control factor

See also: `Driver`, `Def_TwoCoilsDriver`, `Def_PiezoDriver`,  
`Def_MeasRadiator`, `Def_BassUnit`, `Def_Speaker`

## Def\_TwoCoilsDriver

Dialog: `Def/ Def_TwoCoilsDriver`

`Def_TwoCoilsDriver` defines an electrodynamic driver with two voice coils.

This definition describes the parameter of the `Driver` six-pole network element. You have to enter six connection poles for the related `Driver` network element.

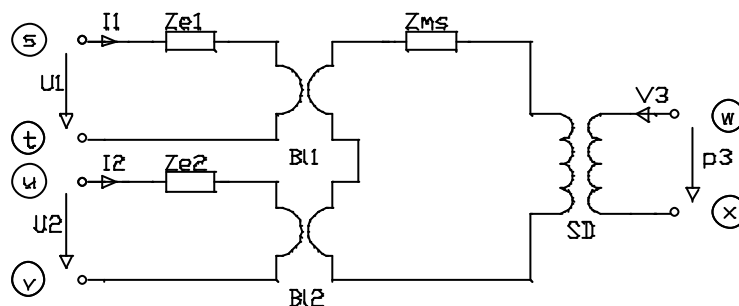


Fig. 50 Equivalent circuit diagram of a dynamic driver with two voice coils

The equivalent circuit diagram of `Def_TwoCoilsDriver` in Fig. 50 is similar to that of the `Def_Driver` element. The two voice coils with impedances  $Ze1$  and  $Ze2$  are in the same magnetic field and, in most loudspeaker models, also have the same wire length.

There are some practical reasons for equipping a dynamic driver with two voice coils. For example, the level of the driving-point impedance can be switched over depending on whether the two coils are connected in series or in parallel. Furthermore, the transmission characteristic can be modified predominantly in the region of the resonance by means of additional components, for example an inductor, in a manner that is difficult with only one voice coil. Feedback for monitoring the diaphragm movement is also easier with two voice coils (see chapter Filter/Feedback).

As can be seen from Fig. 50, the input voltages  $U_1$ , across the poles **s** and **t**, and  $U_2$ , across the poles **u** and **v**, generate a force  $F = Bl_1 \cdot I_1 + Bl_2 \cdot I_2$  in the mechanical part of the driver. The velocity of the oscillating mechanics is the same for both voice coils.

The diaphragm surface couples the mechanical part to the acoustic part. Pole **w** represents the front side of the diaphragm and pole **x** its reverse side.

### ☒ Example

Bass loudspeaker with two (identical) voice coils.

```
Def_TwoCoilsDriver    'Drv 1'
  dD=17cm  dD1=5cm  tD1=5cm  fp=1800Hz  |Cone
  fs=30Hz  Vas=135.3L  Qms=1.92
  Qes1=0.8  Re1=5ohm  fre1=5kHz  Le1=1.3mH  ExpoLe1=0.5
  Qes2=0.8  Re2=5ohm  fre2=5kHz  Le2=1.3mH  ExpoLe2=0.5
```

## Parameter

Def_TwoCoilsDriver	Keyword
'...'	Name of the definition. Indispensable here, since the <code>Driver</code> network elements with the identifier <code>Def='...'</code> refer to their definition. Range: any characters, but max. 20.

### Parameter of transducer see 'Introduction/ Transducer'

<code>fs=...Hz</code>	Resonance frequency.
<code>Mms=...kg</code>	Mass of diaphragm assembly inclusive airload.
<code>Vas=...m3</code> (alternative)	Equivalent compliance-volume.
<code>Cms=...m/N</code> (alternative)	Mechanical compliance of suspension.
<code>Qms=...</code>	Mechanical quality factor.
<code>Rms=...Ns/m</code> (alternative)	Mechanical resistance of driver suspension.
<code>Qes1=...</code>	Electical quality factor, 1st voice coil.
<code>Bl1=...Tm</code> (alternative)	Conversion factor, 1st voice coil.
<code>Re1=...ohm</code>	Voice coil dc. resistance, 1st voice coil.
<code>fre1=...Hz</code> (optional)	Makes <code>Re1</code> frequency dependent to simulate the influence of eddy currents at high frequencies.
<code>ExpoRe1=...</code> (optional)	Controls the resistance of the 1st voice coil.
<code>Le1=...H</code> (optional)	Voice coil inductance, 1st voice coil (with non-linear reactance).

<code>ExpoLe1=...</code>	(optional)	Controls the reactance of the 1st voice coil.
<code>Qes2=...</code>		Electical quality factor, 2nd voice coil.
<code>Bl2=...Tm</code>	(alternative)	Conversion factor, 2nd voice coil.
<code>Re2=...ohm</code>		Voice coil dc. resistance, 2nd voice coil.
<code>fre2=...Hz</code>	(optional)	Makes Re2 frequency dependent to simulate the influence of eddy currents at high frequencies.
<code>ExpoRe2=...</code>	(optional)	Controls the resistance of the 2nd voice coil.
<code>Le2=...H</code>	(optional)	Voice coil inductance, 2nd voice coil (with non-linear reactance).
<code>ExpoLe2=...</code>	(optional)	Controls the reactance of the 2nd voice coil.
<code>Meas_...</code>		Keyword to indicate the radiation impedance of the parameter.
		<code>Meas_Dipole</code> no baffle (default value)
		<code>Meas_TubeEnd</code> no baffle, one side radiation
		<code>Meas_Baffle1</code> baffle, one side radiation
		<code>Meas_Baffle2</code> baffle, both side radiation
		<code>Meas_DoNotModify</code> The parameter are not modified.

#### Parameter of diaphragm dimensions see 'Introduction/ Diaphragm Forms'

<code>dD=...m</code>		Diameter of diaphragm.
<code>WD=...m</code>	(alternative)	Width and height of rectangular diaphragm.
<code>HD=...m</code>		
<code>SD=...m2</code>	(alternative)	Area of circular diaphragm.
<code>dD1=...m</code>	(optional)	Diameter of inner diaphragm.
<code>SD1=...m2</code>	(alternative)	Area of inner diaphragm.
<code>tD1=...m</code>	(optional)	Depth or height of diaphragm form.
<code>t1=...m</code>	(optional)	Displacement of diaphragm within baffle.
<code>fp=...Hz</code>	(optional)	Mass and area reduction control frequency.
<code>Diffuse=...%</code>	(optional)	Directivity control factor

See also:

`Driver`, `Def_Driver`, `Def_PiezoDriver`, `Def_MeasRadiator`,  
`Def_BassUnit`, `Def_Speaker`

## Def\_PiezoDriver

*Dialog: Def/ Def\_PiezoDriver*

`Def_PiezoDriver` defines an electroacoustic piezo driver. The associate network element is the `Driver` four-pole.

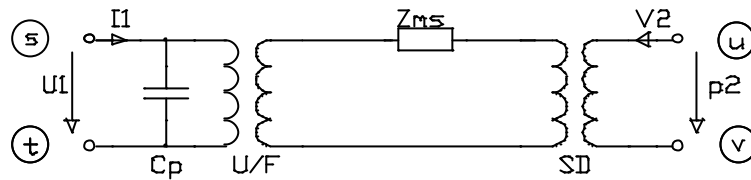


Fig. 51 Equivalent circuit diagram of an electroacoustic piezo driver

AkAbak implements a relatively simple model of the piezo driver. Its equivalent circuit diagram is shown in Fig. 51. Piezo drivers have a piezoelectric crystal, which deflects as soon as the voltage  $U1$  is applied, and thus generate the force  $F$ , which drives the diaphragm. The conversion factor of the piezo driver is  $U/F$ .  $Z_{ms}$  is mechanical impedance of the vibrating system, including the compliance of the piezo.  $C_p$  represents the piezoelectric capacitance. You can read more about these parameter and how to measure them in the chapter 'Tools/Piezo Driver Parameter...'.  
 The poles **s** and **t** designate the electrical connection terminals of the driver. Poles **u** and **v** are at the acoustic side. Pole **u** represents the front side of the diaphragm and pole **v** its rear side.

### ☒ Example

Conical tweeter closed at the back.

```
Def_PiezoDriver    'Drv1'
  Meas_TubeEnd
  dD=4.3cm  dD1=2mm  tD1=1.5cm  fp=10000Hz  |Cone
  fs=3.74kHz  Mms=0.7g  Qms=12.6  Cp=0.147uF  F/U=0.05N/V
```

## Parameter

Def_PiezoDriver	Keyword
'...'	Name of the definition. Indispensable here, since the <code>Driver</code> network elements with the identifier <code>Def='...'</code> refer to their definition. Range: any characters, but max. 20.
$C_p = \dots F$	Piezoelectric capacitance that can be measured across the driver terminals at very high frequencies, i.e. well above the resonance range ( $f_s \ll f$ ). Unit: farad [F].
$F/U = \dots N/V$	Conversion factor: force due to voltage: Unit: newton/volt [N/V].

### Parameter of transducer see 'Introduction/ Transducer'

$f_s = \dots Hz$	Resonance frequency.
$M_{ms} = \dots kg$	Mass of diaphragm assembly including air load.
$V_{as} = \dots m^3$ (alternative)	Equivalent compliance-volume.
$C_{ms} = \dots m/N$ (alternative)	Mechanical compliance of suspension.
$Q_{ms} = \dots$	Mechanical quality factor.
$R_{ms} = \dots Ns/m$ (alternative)	Mechanical resistance of driver suspension.

Meas_...	Keyword to indicate the radiation impedance of the parameter.	
	Meas_Dipole	no baffle (default value)
	Meas_TubeEnd	no baffle, one side radiation
	Meas_Baffle1	baffle, one side radiation
	Meas_Baffle2	baffle, both side radiation
	Meas_DoNotModify	The parameter are not modified.

### Parameter of diaphragm dimensions see 'Introduction/ Diaphragm Forms'

dD=...m		Diameter of diaphragm.
WD=...m	(alternative)	Width and height of rectangular diaphragm.
HD=...m		
SD=...m <sup>2</sup>	(alternative)	Area of circular diaphragm.
dD1=...m	(optional)	Diameter of inner diaphragm.
SD1=...m <sup>2</sup>	(alternative)	Area of inner diaphragm.
tD1=...m	(optional)	Depth or height of diaphragm form.
t1=...m	(optional)	Displacement of diaphragm within baffle.
fp=...Hz	(optional)	Mass and area reduction control frequency.
Diffuse=...%	(optional)	Directivity control factor

See also: Driver, Def\_Driver, Def\_TwoCoilsDriver, Def\_MeasRadiator, Def\_BassUnit, Def\_Speaker

## Def\_MeasRadiator

*Dialog: Def/ Def\_MeasRadiator*

Def\_MeasRadiator implements any loudspeaker. Def\_MeasRadiator does not describe a model, but points to a list whose values have been obtained from the sound pressure curve and impedance curve of a loudspeaker. Def\_MeasRadiator can be used to include in the simulations loudspeakers or loudspeaker systems that it is otherwise difficult or even impossible to describe.

Before you can use the MeasRadiator element, you first have to measure the loudspeaker by means of a suitable spectrum analyzer. First measure the sound pressure curve on axis at a specific distance and a specific voltage, then the electrical impedance curve across the driver terminals. If possible, amplitude and phase or real and imaginary should be measured. This is compulsory if the transmission of the radiator is not of minimum-phase as for instance the horn-speaker.

By means of the Tools menu 'Tools/Generate Def\_MeasRadiator File...', the measured data are used to generate the list. The list is stored in a file on the hard disk. Only the name of this file is then given in the definition of Def\_MeasRadiator.

The reference to the network is then formed using the MeasRadiator network element. As usual as many MeasRadiator network elements as desired can then be relate to a definition.

### ☒ Example

Measured loudspeaker.

```

Def_MeasRadiator      'M1'
  Filename='RAD_0.AKR'      | On-axis
  |Filename='RAD_30.AKR'
  |Filename='RAD_60.AKR'

```

## Parameter

Def_MeasRadiator	Keyword
'...'	Name of the definition. necessary here, since the <code>MeasRadiator</code> network elements are linked to their definition by the identifier <code>Def='...'</code> . Range: Any characters, but no more than 20.
Filename=...	Name of the file containing the measurement data prepared via the Tools menu: 'Tools/Generate Def_MeasRadiator File'. The file name has to be in quotation marks ('...' or '...'). As for numerical entries, it must not contain spaces. If you do not give the file name extension, it is automatically set to AKR. The AkAbak installations routine creates a directory \MEASRAD for storing the files associated with the <code>MeasRadiator</code> element. If you do not give the drive and path name with the file name in the script, the program looks for the file in the \MEASRAD directory.
See also:	<code>MeasRadiator</code> , <code>Def_Driver</code> , <code>Def_PiezoDriver</code> , <code>Def_TwoCoilsDriver</code> , <code>Def_BassUnit</code> , <code>Def_Speaker</code>

## Def\_Speaker

*Dialog: Def/ Def\_Speaker*

`Def_Speaker` implements a simplified model of the dynamic loudspeaker in a sealed enclosure. Using the speaker element, you can quickly and easily install a simple loudspeaker, including enclosure, in the network. The enclosure is in this case firmly connected to the driver, as is the case, for example, in many tweeters and dome loudspeakers. You do not enter the enclosure volume, instead all driver parameter relate to the 'installed' state. The associate network element is the `Speaker` two pole.

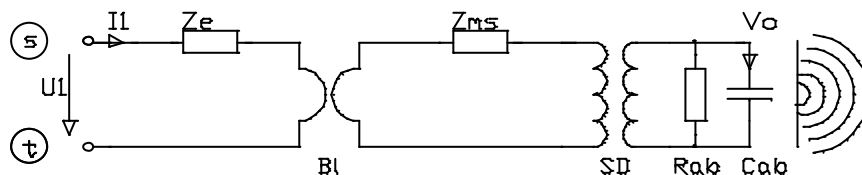


Fig. 52 Equivalent circuit diagram of the `Speaker` element

### Enclosure

$C_{ab}$  and  $R_{ab}$  represent the acoustical compliance of the enclosure and the losses in the enclosure. Since no explicit details of the enclosure will be required here, some values are combined. The details of the compliance of the diaphragm ( $V_{as}$ ,  $C_{ms}$ ) thus also include the compliance of the enclosure, and those of the mechanical quality ( $Q_{ms}$ ,  $R_{ms}$ ) include the acoustical losses of the enclosure  $R_{ab}$ .



Examples

1.: Conical loudspeaker

```

Def_Speaker 'Spl'
  fs=50Hz  Vas=10L
  Qms=3  Qes=1  Re=5ohm  fre=10kHz  Le=0.3mH  ExpoLe=0.5
  dD=15cm  dD1=3cm  tD1=2cm  fp=1600Hz  |Cone
  Rg=0.5ohm  mb=0.95

```

## 2.: Dome loudspeaker

```

Def_Speaker 'dome'
  fs=1580Hz  Mms=1.2g
  Qms=0.73  Qes=1.49  Re=5.22ohm  Le=0.12mH  ExpoLe=0.8
  dD=26mm  tD1=7mm  fp=4500Hz  |Dome
  Rg=0.5ohm  mb=0.95

```

## Parameter

Def_Speaker	Keyword
'...'	Name of the definition. Indispensable here, since the <code>Speaker</code> network elements with the identifier <code>Def='...'</code> refer to their definition. Range: any characters, but max. 20.

### Parameter of transducer see 'Introduction-Transducer'

<code>fs=...</code>	Hz	Resonance frequency of free radiating driver.
<code>Mms=...</code>	kg	Mass of diaphragm assembly inclusive airload.
<code>Vas=...</code>	m <sup>3</sup> (alternative)	Equivalent compliance-volume of free radiating driver.
<code>Cms=...</code>	m/N (alternative)	Mechanical compliance of suspension.
<code>Qms=...</code>		Mechanical quality factor of free radiating driver.
<code>Rms=...</code>	Ns/m (alternative)	Mechanical resistance of driver suspension.
<code>Qes=...</code>		Electical quality factor of free radiating driver( $R_g=0$ ).
<code>Bl=...</code>	Tm (alternative)	Conversion factor.
<code>Re=...</code>	ohm	Voice coil dc. resistance.
<code>fre=...</code>	Hz (optional)	Makes <code>Re</code> frequency dependent to simulate the influence of eddy currents at high frequencies.
<code>ExpoRe=...</code>	(optional)	Controles the resistance of the voice coil.
<code>Le=...</code>	H (optional)	Voice coil inductance (with non-linear reactance).
<code>ExpoLe=...</code>	(optional)	Controles the reactance of the voice coil.
<code>Meas_...</code>		Keyword to indicate the radiation impedance of the parameter.
	<code>Meas_Dipole</code>	no baffle
	<code>Meas_TubeEnd</code>	no baffle, one side radiation (default value)
	<code>Meas_Baffle1</code>	baffle, one side radiation
	<code>Meas_Baffle2</code>	baffle, both side radiation
	<code>Meas_DoNotModify</code>	The parameter are not modified.



dD= . . . m		Diameter of diaphragm.
WD= . . . m	(alternative)	Width and height of rectangular diaphragm.
HD= . . . m		
SD= . . . m <sup>2</sup>	(alternative)	Area of circular diaphragm.
dD1= . . . m	(optional)	Diameter of inner diaphragm.
SD1= . . . m <sup>2</sup>	(alternative)	Area of inner diaphragm.
tD1= . . . m	(optional)	Depth or height of diaphragm form.
t1= . . . m	(optional)	Displacement of diaphragm within baffle.
f <sub>p</sub> = . . . Hz	(optional)	Mass and area reduction control frequency.
Diffuse= . . . %	(optional)	Directivity control factor

Rg= . . . ohm	(optional)	<p>The generator resistance Rg is in series with the resistance of the voice coil Re. It comprises the cable impedances and transition resistances or special values provided by the power amplifier (these values may even be negative, in this case take care that <math>Re+Rg &gt; 0</math>). If a series resistor is specified in the network take care to adjust Rg within Def_Speaker.</p> <p>Unit: ohm [ohm].</p>				
mb= . . .	(optional)	<p>Mass load factor. An empirical factor which takes into account the greater air load when the loudspeaker is installed in an enclosure and the parameter have been measured in the free-radiating environment. As a consequence of mb, the resonance fs of the free-radiating driver (<math>fsb=mb\cdot fs</math>) is reduced. The qualities of the free-radiating driver Qes and Qms increase in value (<math>Qesb=Qes/mb</math>, <math>Qmsb=Qms/mb</math>).</p> <p>The parameter of the speaker has been measured:</p> <table border="0" style="margin-left: 20px;"> <tr> <td style="padding-right: 40px;">mb approx. 0.95</td> <td>free radiating</td> </tr> <tr> <td>mb = 1</td> <td>mounted in baffle</td> </tr> </table> <p style="text-align: right;">(value if mb not specified).</p>	mb approx. 0.95	free radiating	mb = 1	mounted in baffle
mb approx. 0.95	free radiating					
mb = 1	mounted in baffle					

See also: `Speaker`, `Def_BassUnit`, `Def_Driver`, `Def_TwoCoilsDriver`,  
`Def_PiezoDriver`, `Def_MeasRadiator`

*Shortcut: Ctrl + B*

This element is intended either for quick simulation or as an introduction to the design of a loudspeaker system. Since the 'Def BassUnit calculator' (see below) forming part of `Def BassUnit` provides extensive

design aids, it is best to start to design the bass speaker using this dialog. The system can be examined in greater detail at a later time by subdividing it into `Driver`-, `Enclosure`- and `Radiator` elements. The associated network element is `Bassunit`.

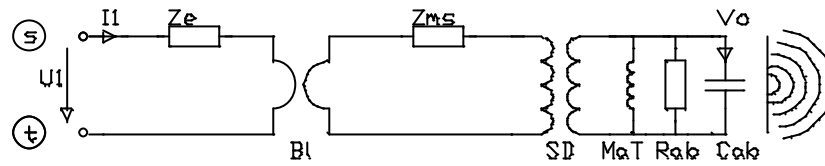


Fig. 53 Equivalent circuit diagram for `Def_BassUnit`

Fig. 53 shows the equivalent circuit diagram of this element. `Cab`, `Rab` and `MaT` represent the acoustic compliance of the enclosure, the losses in the enclosure and the acoustic mass of any reflex vent present. These parameters are dealt with at length in the description of the `Enclosure` element. Most of what is stated there also applies in this case. Standing waves in the enclosure, however, cannot be taken into account in the `BassUnit` element. Also, no additional vent quality can be specified for the reflex port.

## 2nd order high-pass filter

An electrical high-pass filter connected before the loudspeaker can substantially improve reproduction characteristics and the load capacity of the bass loudspeaker. The filter limits the excursion of the diaphragm. The model used in the `Def_BassUnit` element is relatively simple and allows a mathematically closed description. It is therefore an obvious step to specify the required high-pass filter at the same time. The dialog forming part of `Def_BassUnit` element supports high-pass filtered system and also calculates the performance data.

### Transfer function

In the actual design, the voltage across the driver must obey the transfer function of high-pass filter. It doesn't really matter in practice how this is accomplished; usually a small filter circuit before the power amplifier is used.

The HP filter of the `BassUnit` element can only be implemented by an active filter circuit. The filter is therefore connected before the current network in which the `BassUnit` is located. It weights the input voltage `U1` of the network with the following transfer function:

$$HP(s) = \frac{S^2}{S^2 + S / Qe + 1}, \text{ where } S = \frac{s}{2\pi f_e} \text{ and } s: \text{ complex frequency variable}$$

Only one `BassUnit` HP filter can therefore be located in a system. If several different filters were to be used, the overall transfer function would be completely different, since the filter functions are multiplied together. This is not a restriction of the program. In the practical circuit, a separate filter with power amplifier should also be provided for each high-pass filtered bass loudspeaker.

The program checks the entered structure. If several `Def_BassUnit/BassUnit`-elements have been used in one network, the following procedure is carried out:

1. The parameter `Qe=` and `fe=` are the same in the `Def_BassUnit` elements:  
→ The driving point voltage `U1` is weighted with the function `HP(s)`.
2. A filter has only been entered for one `Def_BassUnit` definition. No filter has been entered for the other definitions used in the same network (system):  
→ The input voltage `U1` is weighted with the function `HP(s)`, so that in practice the other `BassUnits` are also high-pass filtered.
3. A network (system) contains `BassUnit` elements whose definitions have different `Qe=` and `fe=` parameter:  
→ error message.

Simulation of the overall input parameter current and power (Menu: `Sum/Driving point current`, `Sum/Driving point power`):

These types of simulations display the curves of current and power. The voltage source has to apply all networks connected in parallel. If several Def\_BassUnit/BassUnit elements are used (in all systems), the following takes place:

1. The parameter  $Q_e$  and  $f_e$  are the same in all Def\_BassUnit elements:  
→ The source voltage  $U_{in}$  is weighted with the function  $HP(s)$ .
2. A filter has been entered for only one Def\_BassUnit definition. No filter has been entered for the other definitions used:  
→ The source voltage  $U_{in}$  is weighted with the function  $HP(s)$ , so that in practice all other Bass Units are also high-pass filtered.
3. BassUnit elements are used whose definitions have different  $Q_e$  and  $f_e$  parameter:  
→ error message.

### ☒ Examples

#### 1.: Loudspeaker in a sealed enclosure.

```
Def_BassUnit 'Bass3'
fs=30Hz Vas=40L
Qms=1 Qes=0.6 Re=6ohm Le=0.8mH ExpoLe=0.618
dD=15cm |Piston
Xmss=1cm Rg=0.5ohm mb=0.94
Vb=18.22L Qb/fo=0.11
|Performance of sealed enclosure:
| fc      Qtc      fD      f3
| 50.4Hz   0.66     728.5Hz  54.3Hz
| Lwmax    Pelmax   Uomax    t60      Ripple
| 95.5dB   15.8W     14.32V   28.2ms   0
```

#### 2.: Loudspeaker in a sealed enclosure, high-pass filtered, conical diaphragm and with mass reduction at high frequencies ( $f_p=1200\text{Hz}$ )

```
Def_BassUnit 'Bass2'
fs=30Hz Vas=40L
Qms=1 Qes=0.6 Re=6ohm Le=0.8mH ExpoLe=0.618
dD=15cm dD1=4cm tD1=4.5cm fp=1200Hz |Cone
Xmss=1cm Rg=0.5ohm mb=0.94
Vb=30.0L Qb/fo=0.11
Qe=1.6 fe=36.7Hz
|Performance of sealed enclosure:
| fc      Qtc      fD      f3
| 43.1Hz   0.564    728.5Hz  35.7Hz
| Lwmax    Pelmax   Uomax    t60      Ripple
| 92.4dB   7.7W      10.02V   0.1s     8.2mdB
```

#### 3.: Loudspeaker in a reflex enclosure, high-pass filtered, rectangular diaphragm.

Instead of  $V_{as}$ , you specify the oscillating mass  $M_{ms}$ .

```
Def_BassUnit 'XR-201-B'
fs=30Hz Mms=30.5g
Qms=1 Qes=0.6 Re=6ohm Le=0.8mH ExpoLe=0.618
WD=11cm HD=16cm |Piston
Xmss=1cm Rg=0.5ohm mb=0.94
Vb=52.32L fb=34.5Hz Qb/fo=0.11
Qe=0.637 fe=35.1Hz
|Performance of vented enclosure:
| fsb     Qtr      fD      f3
| 28.2Hz  0.419     730.0Hz  36.3Hz
```

## Parameter

## Keyword

Name of the definition. Indispensable here, since the `Bassunit` network elements with the identifier `Def='...'` refer to their definition.  
Range: any characters, but max. 20.

$f_s = \dots \text{Hz}$		Resonance frequency of free radiating driver.
$V_{as} = \dots \text{m}^3$		Equivalent compliance-volume of free radiating driver.
$M_{ms} = \dots \text{kg}$	(alternative)	Mass of diaphragm assembly inclusive airload.
$C_{ms} = \dots \text{m/N}$	(alternative)	Mechanical compliance of suspension.
$Q_{ms} = \dots$		Mechanical quality factor of free radiating driver.
$R_{ms} = \dots \text{Ns/m}$	(alternative)	Mechanical resistance of driver suspension.
$Q_{es} = \dots$		Electical quality factor of free radiating driver( $R_g=0$ ).
$B_l = \dots \text{Tm}$	(alternative)	Conversion factor.
$R_e = \dots \text{ohm}$		Voice coil dc. resistance.
$f_{re} = \dots \text{Hz}$	(optional)	Makes $R_e$ frequency dependent to simulate the influence of eddy currents at high frequencies.
$Exp_{oRe} = \dots$	(optional)	Controles the resistance of the voice coil.
$L_e = \dots \text{H}$	(optional)	Voice coil inductance (with non-linear reactance).
$Exp_{oLe} = \dots$	(optional)	Controles the reactance of the voice coil.
$Meas\_ \dots$		Keyword to indicate the radiation impedance of the parameter.
		$Meas\_Dipole$ no baffle (default value)
		$Meas\_TubeEnd$ no baffle, one side radiation
		$Meas\_Baffle1$ baffle, one side radiation
		$Meas\_Baffle2$ baffle, both side radiation
		$Meas\_DoNotModify$ The parameter are not modified.

dD= . . . m		Diameter of diaphragm.
WD= . . . m	(alternative)	Width and height of rectangular diaphragm.
HD= . . . m		
SD= . . . m <sup>2</sup>	(alternative)	Area of circular diaphragm.
dD1= . . . m	(optional)	Diameter of inner diaphragm.
SD1= . . . m <sup>2</sup>	(alternative)	Area of inner diaphragm.

$t_{D1} = \dots m$	(optional)	Depth or height of diaphragm form.
$t_1 = \dots m$	(optional)	Displacement of diaphragm within baffle.
$f_p = \dots Hz$	(optional)	Mass and area reduction control frequency.
$Diffuse = \dots \%$	(optional)	Directivity control factor

### Specific Parameter

$X_{ms} = \dots m$	(optional)	Maximum diaphragm excursion of the loudspeaker. $X_{ms}$ is not required for simulation by means of the script, but is required within the <code>Def_BassUnit</code> dialog to calculate the maximum possible sound pressure $L_{wmax}$ . Unit: meter [m] or inch [in].
$R_g = \dots ohm$	(optional)	The generator resistance $R_g$ is in series with the resistance of the voice coil $R_e$ . It comprises the cable impedances and transition resistances or special values provided by the power amplifier (these values may even be negative, in this case take care that $R_e + R_g > 0$ ). $R_g$ provides in particular the possibility of investigating, within the <code>Def_BassUnit</code> dialog, the effect of an actual series resistor on $Q_{es}$ . If a series resistor is specified in the network take care to adjust $R_g$ within <code>Def_BassUnit</code> . Unit: ohm [ohm].
$m_b = \dots$	(optional)	Mass load factor. An empirical factor which takes into account the greater air load when the loudspeaker is installed in an enclosure and the parameter have been measured in the free-radiating environment. As a consequence of $m_b$ , the resonance $f_s$ of the free-radiating driver ( $f_{sb} = m_b \cdot f_s$ ) is reduced. The qualities of the free-radiating driver $Q_{es}$ and $Q_{ms}$ increase in value ( $Q_{esb} = Q_{es}/m_b$ , $Q_{msb} = Q_{ms}/m_b$ ). <div style="display: flex; justify-content: space-between;"> <div> <math>m_b \approx 0.9</math>  <math>m_b \approx 0.95</math>  <math>m_b = 1</math> </div> <div> enclosure filled with insulating material.  enclosure empty.  The driver parameter have been measured in a baffle (value if <math>m_b</math> not specified). </div> </div>
$V_b = \dots m^3$		Effective volume of the enclosure. The volume is increased if damping material is present (see also A-Net/ AcouCompliance). Unit: cubic meter [ $m^3$ ], in cubic inch [ $in^3$ ] or liter [Liter, L]
$Q_b / f_o = \dots$	(optional)	Enclosure quality factor divided by the frequency at which $Q_b$ has been measured. The value of $Q_b / f_o = 0.1$ if, for example, with a Helmholtz resonance of $f_b = 50 Hz$ , a quality of $Q_b = 5$ has been measured. The value of $Q_b / f_o = 0.1$ is typical for a medium sized enclosure. A smaller $Q_b / f_o$ value means higher enclosure losses ( $R_{ab}$ small). If not specified $AkAbak$ sets $Q_b / f_o = 1000$ which stands for the lossless case. Range: $0.0001 < Q_b / f_o < 1000$
$f_b = \dots Hz$	(optional)	Helmholtz resonance of the reflex enclosure. When $f_b$ is specified, the bass cabinet becomes a reflex enclosure. Otherwise <code>Def_BassUnit</code> describes a sealed enclosure. Unit: hertz [Hz].
$Q_e = \dots$	(optional)	Quality of any 2nd order electrical high-pass filter present. $Q_e$ has to be specified together with $f_e$ (see below). $Q_e = 1/d$ , where $d$ is the attenuation factor of the High-pass filter.

`fe=...Hz` (optional) Pole frequency of any 2nd order electrical high-pass filter present.  
`fe` has to be specified together with `Qe=` (see above).  
 Unit: hertz [Hz].

## Several drivers in one enclosure

If several **identical** loudspeakers are distributed in one enclosure, the volume of each driver has to be specified. If, for example, two identical drivers are mounted in a 100L enclosure, `Def_BassUnit Vb=50L` has to be specified. Two `Bassunit` elements with the same `Def='...'` then appear in the network. The `BassUnit` element is a self-contained module for quick analysis. It is not suitable for analyzing the reproduction of several drivers that are acoustically coupled. You construct such a circuit from discrete elements (`Driver`, `Radiator`, `Enclosure`, ...), especially if they do not have identical parameter.

## Enclosure losses

As with the equivalent circuit diagram of the `Enclosure` element, the acoustic impedance `Rab` represents losses in the enclosure.

The quality `Qb`, which is a function of frequency, has the following relationship with `Rab`:

$$Qb = 2 \cdot \pi \cdot fb \cdot Cab \cdot Rab.$$

`fb` is the Helmholtz frequency at which `Qb` has been measured. If you want to measure `Qb/fo` for a sealed enclosure, install a vent temporarily to produce a reflex enclosure. Measure the Helmholtz frequency `fb` and the quality `Qb` of this reflex enclosure. `Qb/fo` is then `Qb/fb`.

`Rab` especially represents losses due to leakages in the enclosure. A further loss impedance, which represents absorption is connected in series with the compliance of the enclosure `Cab` (not shown in Fig. 53). Losses occur due to absorption of the sound energy in the enclosure wall, in the bass range due to sympathetic vibration of the walls, and, at higher frequencies, due to cavities. Since, if the boxes are constructed with some degree of stability, this impedance is negligibly small, in the model it is contained in `Rab`.

If the air current is not restricted in the reflex vent, then here, too, a series impedance with the vent mass `MaT` is usually negligible and is reflected in `Rab`. `Rab` thus not only represents leakages, but also other losses in the enclosure and in the vent, if the latter do not predominate.

## Transformation of the BassUnit into discrete elements.

The `BassUnit` model is very useful for starting the design. However, a more accurate analysis is obtained with the discrete elements `Driver-Radiator-Enclosure-Filter`. The parameter of `Def_BassUnit` therefore have to be distributed among these elements.

### Example

```
Def_BassUnit    'B1'
  fs=30Hz  Vas=40L                                | -> Def_Driver
  Qms=1  Qes=0.6
  Re=6ohm  fre=10kHz  Le=0.8mH  ExpoLe=0.5
  dD=15cm  dD1=4cm  tD1=4.5cm  fp=1.5kHz          | -> Radiator
  Xms=0.5cm  mb=0.94
  Rg=0.5ohm                                         | -> Resistor
  Vb=30.0L  Qb/fo=0.11                             | -> Enclosure
  Qe=1.6  fe=36.7Hz                                | -> Filter
```

Convert into:

```
Def_Driver      'D1'
  fs=30Hz  Vas=40L
```

```

Qms=1  Qes=0.6
Re=6ohm  fre=10kHz  Le=0.8mH  ExpoLe=0.5
dD=15cm  dD1=4cm  tD1=4.5cm  fp=1.5kHz

System  'Bass'
...
Resistor  'Rg'      Node=1=2      R=0.5ohm
Driver    Def='D1'   Node=2=0=3=4
Radiator  Def='D1'   Node=3=0      x=-10cm
Enclosure 'E1'      Node=4        Vb=30.0L  Qb/fo=0.11  Lb=20cm
Filter    'F1'
    fo=36.7Hz  { Qe=1.6;    b2=1;  a2=1;  a1=1/Qe;  a0=1;  }
...

```

The definition `Def_Driver` contains the parameter of the driver. The `Resistor` element inserts the generator resistance `Rg`. The `Driver` -element locates the driver with the name 'D1' in the current network of the 'Bass' system. Node 2,0 is the electrical connection. Pole 3 stands for the front of the diaphragm and pole 4 for the reverse side. The `Radiator` element receives the parameter entered for diaphragm shape from the `Def_Driver` and locates the radiator on the baffle. The housing is modeled by means of the `Enclosure` element. The parameter `Lb=20cm` generates the correct radiation impedance in the enclosure and replaces the air-load factor `mb`. The `Filter` element describes the HP filter of the high-pass filtered bass speaker and weights the input voltage `U1` of the network.

See also: `BassUnit`, `Def_Speaker`, `Def_Driver`, `Def_TwoCoilsDriver`,  
`Def_PiezoDriver`, `Def_MeasRadiator`

## Def\_BassUnit / Calculator

The dialog 'Def\_BassUnit/ Calculator' for the definition of `Def_BassUnit` is used, both for input of the aforementioned parameter and for design of bass loudspeaker cabinets.

### Data input

After you open the dialog, the input boxes have been initialized with data if the script cursor is located within one of the following definitions:

```
Def_BassUnit, Def_Speaker, Def_Driver, Def_TwoCoilsDriver
```

The dialog contains the 'From Script' button, which starts the process of reading in from the script while the dialog remains open. Since the `Def_BassUnit` dialog is non-modal, you can leave it and move, for example, the script cursor to read in the parameter from a different driver. This function allows you to test the reproduction characteristics of different drivers in the same enclosure. The driver parameter may be stored, for example, as `Def_Driver` definitions in a special script in the form of a database.

If `Def_BassUnit` describes a reflex enclosure, click on the 'Vented'. switch. The input box for the Helmholtz resonance `fb` is now accessible. From this box, you can start the 'Helmholtz' subdialog. There the information on the reflex vent are provided only for information. The `BassUnit` element uses a compact model for the calculation and, by contrast with the `Enclosure` element, does not need details of the vent geometry.

The 'HP Filter' switch makes the input boxes `fe` and `Qe` of the electrical high-pass filter accessible.

Using the button 'Copy to clipboard and close', you format the data of the input boxes and store them in the clipboard, from where you can insert them into the script as a definition.

The dialog also outputs three commentary lines, which summarize the results of reproduction calculation. If you do not require these, you can delete them.

## Design of bass loudspeakers

The further buttons and displays of the Bassunit dialog are used for designing sealed and reflex enclosures, with and without a 2nd order electrical high-pass filter. The model used and the concept follow the description give in [Pan1]<sup>3</sup>.

This model summarizes the electrical, mechanical and acoustic part in one transfer function. It is thus possible to determine and optimize the reproduction characteristics and load-capacity from the diaphragm excursion graph of the bass loudspeaker. You can set the parameter with the aid of filter functions, which are given in the form of tables. These characteristic graphs show known characteristics that are applied to the loudspeaker system.

There are a few minor differences from [Pan1], particularly in the form of the presentation:

1. The alignment tables are output in denormalized form and thus relate to the driver parameter input in the main dialog (Def\_BassUnit).
2. Since the alignments are output in denormalized form, the  $k_p$  factor is dispensed with. Instead, the sound power level at maximum diaphragm excursion  $L_{wmax}$  is calculated.  $L_{wmax}$  relates to a  $4\pi$ -sr radiation environment (free-radiating set up).
3. Instead of the enclosure quality  $Q_b$ , the quality factor  $Q_b/f_0$  is used. Here,  $f_0$  is the frequency at which  $Q_b$  has been measured, this is usually the Helmholtz resonance  $f_b$ .
4. The quality factor  $Q_b/f_0$  of the enclosure has to be specified even for sealed enclosures.
5. The electrical power  $P_{elmax}$  is that power that the generator has to supply to generate the sound power level  $L_{wmax}$  at a driver driving-point impedance equal to  $R_e + R_g$ .
6. Instead of the attenuation factor  $d$  of the electrical high-pass filter, the quality  $Q_e$  of this filter is used. The quality is the reciprocal of the attenuation factor ( $d=1/Q_e$ ).

## Calculations in the 'Def\_BassUnit / Calculator' dialog

At the bottom left-hand side of the dialog, a panel is displayed showing the results of the calculation of the reproduction characteristics of the loudspeaker. You start the calculation using the 'Evaluate' button or Enter ↵. If some of the parameter have not been specified, you may not be able to calculate all the values.

The first two outputs change depending on the position of the 'Vented' switch. If it is switched off then the loudspeaker is high-pass filtered or a simple sealed enclosure. Otherwise a high-pass filtered or simple reflex enclosure is evaluated.

$f_c$	Hz	Resonance frequency of the loudspeaker in a sealed enclosure. Unit: hertz [Hz]	$f_c = m_b \cdot f_s \cdot \sqrt{1 + \frac{V_{as}}{V_b}}$
$Q_{tc}$		Total quality of the loudspeaker in a sealed enclosure	$Q_{tc} = \frac{m_b}{\frac{Q_{ms}}{m_b} + \frac{m_b}{Q_{es} \cdot (1 + R_g/R_e)} + \frac{1}{(Q_b / f_0) \cdot m_b \cdot f_s}}$
$f_{sb}$	Hz	Resonance of the loudspeaker in a reflex enclosure	$f_{sb} = m_b \cdot f_s$

<sup>3</sup>This book is unfortunately only available in the German language. Among the topics it covers the work of Benson, Thiele and Small and Quasi-Butterworth-alignments (see [Pan3]).



Qtr		Total quality of the loudspeaker in a reflex enclosure	$Q_{tr} = \frac{1/mb}{\frac{1}{Q_{ms}} + \frac{1}{Q_{es} \cdot (1+R_g/R_e)}}$
Directivity frequency f <sub>D</sub>	Hz	Directivity frequency of the (circular) diaphragm	$f_D = \frac{c}{\pi \cdot dD}$ , where c: velocity of sound
f <sub>3</sub>	Hz	Lower cut off frequency of the loudspeaker. At the frequency f <sub>3</sub> , the sound power level has dropped by 3 dB with respect to the transmission band.	
L <sub>wmax</sub> , 4-π-sr	dB	<p>Maximum sound power level that the loudspeaker can achieve with free-radiating radiation. The modulus is calculated from the entry for the maximum possible diaphragm excursion of the driver X<sub>ms</sub> and the curve of the diaphragm excursion of the loudspeaker in the enclosure. If a high-pass filter is present, it is taken into account. The sound power level L<sub>wmax</sub> is a 'peak' value, i.e. if L<sub>wmax</sub> is exceeded, even only slightly, at the frequency at which the maximum excursion occurs severe distortions occur and the driver may be damaged. The maximum operating sound level of the system is about 10 dB less than L<sub>wmax</sub>.</p> <p>L<sub>wmax</sub> is a so-called asymptotic value. The level applies for high frequencies (f<sub>s</sub> &lt;&lt; f) without taking into account the effect of the voice coil induction and sound directivity. L<sub>wmax</sub> relates to the 4-π-sr listening room and is thus 3 dB less than with baffle conditions (2-π-sr). Entering this makes it easier to calculate the level in the presence of reflectors (see sub-dialog 'Def_BassUnit Calculator Diagram').</p>	
P <sub>elmax</sub>	W	Maximum electrical generator power to generate L <sub>wmax</sub> . The load impedance is the voice coil impedance R <sub>e</sub> plus the generator impedance R <sub>g</sub> .	
U <sub>orms</sub>	V	Rms value of the generator voltage to generate L <sub>wmax</sub> and P <sub>elmax</sub> .	
Reverb. -T <sub>60</sub>	s	Equivalent reverberation time T <sub>60</sub> of the loudspeaker system. T <sub>60</sub> describes the time from switching off the signal until the level has reduced by 60dB. The reverberation time T <sub>60</sub> is an indicator of the pulse characteristic of the bass speaker and can be directly compared with the reverberation time of the listening room, since the latter is defined in the same manner.	
Ripple	dB	Waviness of the sound level curve. If this has a value greater than zero, the L <sub>w</sub> curve has an amplitude response peak in the lower transmission range. Graphs with this characteristic should always be regarded critically, especially if the level of the peak is several decibels. A strong ripple leads to a poor build-up and decay behavior (long reverberation time) and often to a selective overemphasis of the bass positions if the loudspeaker is positioned opposite reflecting walls.	

For a given driver, you can thus obtain reliable information about the reproduction (f<sub>3</sub>, T<sub>60</sub> and Ripple), the power (L<sub>wmax</sub>) and the power consumption (P<sub>elmax</sub>, U<sub>orms</sub>) of the loudspeaker. You can quickly and easily test a change of the parameter of the driver environment, such as R<sub>g</sub>, V<sub>b</sub>, Q<sub>b</sub>/f<sub>0</sub>, f<sub>b</sub>, Q<sub>e</sub> and f<sub>e</sub>. You can just easily look for a driver and a high-pass filter matched to a specific environment, such as a fixed enclosure volume.

## Diagram

If you click on the 'Diagram' button, a subdialog with a diagram is opened.

### Acoustic power level

The first graph represents the level curve of the acoustic output power. It indicates the reproduction characteristics of the loudspeaker at low frequencies. In the upper frequency range, the graph is asymptotic, i.e. the effect of the voice coil induction and the directivity effect are not taken into account here. The graph is normalized to the asymptotic value at high frequencies. Zero decibels corresponds to the level  $L_{wmax}$  at the input voltage  $U_{omax}$ , i.e. to a level in a  $4\pi$ -sr listening room.

### Level of the diaphragm excursion

The second graph shows the level curve of the diaphragm excursion at low frequencies. This also does not take into account the effect of the voice coil induction at low frequencies. The graph is normalized to the asymptotic value at low frequencies when the high-pass filter is not present. The maximum load capacity of the speaker is oriented to the maximum of this function. The load capacity of the loudspeaker is all the higher the better the enclosure and the high-pass filter attenuate the diaphragm excursion. Here reflectors have no effect on the diaphragm excursion graph.

### Transmission of high-pass filter

The third graph only appears when an electrical high-pass filter is involved in the transmission. It shows the level of the transfer function of the high-pass filter. The level is normalized to the asymptotes at high frequencies. From the graph, you can read the additional amplification of the electrical filter and therefore of the following power amplifier, when the transfer function of the filter shows an amplitude magnification. This ripple always occurs when the quality of the high-pass filter  $Q_e > 0.707$ .

### Representation ranges

The input boxes of the 'Range' group in the diagram dialog relate to the representation range of the abscissa and the ordinates.

### Reflections

In the input boxes at the lower right-hand side of the diagram dialog, you can enter the distances from the reflecting walls. The three boxes are equivalent. If you only enter one distance, the loudspeaker is only influenced by one wall. Two reflectors require you to fill in two boxes. If all three are entered, the loudspeaker is located in a corner of a room. The perpendicular measured from the diaphragm center point to the wall describes the distance. If you click on the 'Reflections on' switch and the 'Repaint' button, the sound level graph of the acoustic output power, including reflections, is displayed. This is the reason why the maximum sound power level  $L_{wmax}$  relates to free radiation ( $4\pi$ -sr room). Each reflecting wall increases the level at low frequencies by 3dB. You can therefore easily read how high  $L_{wmax}$  is in, for example, a room corner and for which frequency range that applies. The definition `Def_Reflector` in this chapter gives further details about the reflectors.

### Filter characteristic curves (alignments)

Despite the aids described above, it is not always easy to find the optimum setting of the parameter. One of the most effective and fastest means to arrive at a suitable setting is to use filter characteristic curves. These filter characteristic curves have the same transfer functions as the model of the `Bassunit` element. Since the general characteristics, such as bandwidth, resonance behavior etc. of these functions are known, a list of all the possible meaningful characteristic curves is generated and applied to the loudspeaker model. The result is a catalog from which the setting with the desired characteristics is selected. In the 'Closed Cabinet Alignments' and 'Vented Cabinet Alignments' subdialogs, this catalog is built up according to the given driver data.

### Characteristic curves for sealed enclosures

The 'Closed Cabinet Alignments' subdialog opens when the 'Vented' switch in the 'Def\_BassUnit / Calculator' dialog is switched off and the 'Alignments' button has been pressed. In this case a value for the enclosure quality  $Q_b/f_0$  has to have been entered. The result of this subdialog is the matching enclosure volume  $V_b$  and possibly the high-pass filter parameter.

### Characteristic curves for reflex enclosures

If the 'Vented' switch has been activated, then the 'Alignments' button opens the subdialog 'Vented Cabinet Alignments'. The return values in this case are the enclosure volume  $V_g$ , the enclosure quality  $Q_{b/fo}$ , the Helmholtz resonance  $f_b$ , and possibly the high-pass filter parameter.

In the first line of the alignment dialog, some intermediate results are listed:  $f_{sb}$  is the resonance of the driver, including the air load, as given by the parameter  $m_b$  ( $f_{sb}=m_b \cdot f_s$ ),  $Q_{tsbg}$  is the overall quality consisting of  $Q_{es}$  and  $Q_{ms}$ , including the air load and the generator resistance  $R_g$  (the formula is the same as the formula for  $Q_{tr}$  given above).

### List

In the center of the dialog is the list of the alignments. This is evaluated when you press the 'Alignment List' button. Depending on the model of computer, the evaluation may take some time. A numerical coprocessor accelerates the evaluation considerably. The number of characteristic curves depends on the loudspeaker parameter and on further settings in the dialog. A maximum of 64 characteristic curves are evaluated. You can abort the evaluation at any time using 'Esc'. The meaning of the data in the list has already been explained above.

In the bottom right-hand side of the 'Alignment Type' group, you can set whether the characteristic curves are to contain a high-pass filter and which types of filter functions are to be listed.

### Quasi-Butterworth (QB)

This group of characteristic curves is related to the Butterworth filter functions and, so to speak, inherits the characteristics of the Butterworth function. Butterworth filters have a flattest possible amplitude response and thus have the greatest possible bandwidth without showing an amplitude magnification. The transition from the attenuation band to the pass band is very steep. This is associated with sharp resonance points.

If the parameter of a high-pass filtered reflex loudspeaker are set according to a quasi-Butterworth function, an extremely power system in a relatively small enclosure is obtained.

### Chebyshev (CH)

Filters with a rippling Chebyshev setting ( $k < 1$ ) are extremely steep in the attenuation band. The pass band has a uniform ripple. Based on the loudspeaker enclosure, a considerable bandwidth extension is in some cases obtained. The critical frequency  $f_3$  may be up to an octave below the driver resonance. Rippling Chebyshev settings, however, are problematic in most cases. On one hand it is difficult in practice to produce the very high resonance qualities on the other hand these characteristic curves lead to an extreme diaphragm excursion, so that the bandwidth extension is at the cost of the load capacity (principal of unsharpness).

In the non-rippling Chebyshev settings ( $k > 1$ ), on the other hand, the sound level graph has a rounded transition characteristic.

### Butterworth-Thomson (BT)

This group of filter functions glides from the Butterworth function ( $M=0$ ) to the Bessel function ( $M=1$ ). Butterworth filters have a flattest possible amplitude response and thus have the greatest possible bandwidth without manifesting an amplitude magnification. Instead, their group delay curve has a intense, selective delays. The Bessel filters, on the other hand, have a flattest possible group delay curve (as low-pass), but with a highly rounded transition characteristic in the reproduction curve of the magnitude. Butterworth-Thomson characteristic curves represent a compromise between the two extremes. The greater the factor  $M$  of the characteristic curve, the better is the build-up and decay characteristic of the loudspeaker system.

After the evaluation, the characteristic curves can be selected in the list. The name of the characteristic curve is displayed at the top right-hand side in the dialog. The name consists of the abbreviation of the type of characteristic curve, the pole class, the order and the alignment factor. For example: BT1 4 M= 0 designates a 4th order Butterworth-Thomson characteristic curve, pole class 1, and the characteristic curve factor is  $M=0$  (Butterworth curve) (see [Pan1]).

## Diagram

The characteristic curves are shown in the diagram as soon as you activate the 'Diagram' button. The function is the same as described above. If the diagram dialog is opened from the Alignment dialog, the diagrams of the selected characteristic curve follow.

If you drag the active Alignment dialog downwards somewhat so that the diagram is visible, you can see the characteristic curves without the Diagram dialog popping up in front of them.

The Diagram dialog knows from which dialog it has been called up, and remembers the settings.

## Query

By means of the input boxes in the 'Query' group of the Alignment dialog, you can filter out certain characteristic curves. For example, you may want to display only the alignments that do not exceed a certain enclosure size and/or conform to a certain bandwidth. The characteristic curve generator can often generate considerably more characteristic curves than can be displayed (max. 64 characteristic curves). The query option provides you with more information about the relevant circumstances.

The query input boxes have one special feature. A 'smaller than' (<) or 'greater than' (>) character may come before the first digit. Only those characteristic curve whose corresponding parameter are smaller than or greater than the entered query value are displayed.

Vb	enclosure volume
Qe	quality of the high-pass filter
f3	lower -3dB critical frequency
Lwmax	max. sound power level in the 4- $\pi$ -sr room at 1m distance
Qb/fo	Helmholtz quality factor (only relevant to the reflex enclosure)

## Transferring a setting into the 'Def\_BassUnit / Calculator' dialog

With the 'Copy Alignment to Def\_BassUnit / Calculator' button you copy the relevant data to the 'Def\_BassUnit / Calculator' main dialog.

## → Note

In the closed analysis (i.e. using a transfer function) of the sealed enclosure, it is advisable to regard the loss impedance  $R_{ab}$  in series with the enclosure compliance. The resulting error is generally very small. With the (more accurate) script analysis of the diaphragm excursion, however, this fine distinction is more clearly apparent. The diaphragm excursion of a sealed enclosure is greater at very low frequencies than the diaphragm excursion curve of the Diagram dialog indicates. You can also indicate this analytically. The diaphragm excursion of the sealed enclosure is proportional for very low frequencies:

$$X_{(f \rightarrow 0)} \propto C_{ms} \quad R_{ab} \text{ is parallel to } C_{ab} \text{ and } R_{ab} \text{ (script simulation).}$$

$$X_{(f \rightarrow 0)} \propto C_{mt} \quad R_{ab} \text{ is in series with } C_{ab} \text{ and } C_{ms} \text{ (Def\_BassUnit / Calculator simulation)}$$

with  $C_{ms}$  being the mechanical compliance of the driver diaphragm and  $C_{mt}$  the series connection of  $C_{ms}$  and  $C_{ab}$ .

In the parallel circuit used in the script analysis, the diaphragm excursion at very low frequencies is thus independent of the enclosure shape if serious losses, that is to say leakages, are present, and it is these that  $R_{ab}$  represents. In the 'Def\_BassUnit / Calculator' subdialog, on the other hand, the curve of the diaphragm excursion does not show this effect of the pole point, since in this case  $R_{ab}$  is in series with  $C_{ab}$  in the sealed enclosure.

This fact is usually of no significance in practice, since the difference is only noticeable at frequencies well below the resonance of the system  $f_c$ . If in doubt, use the results of the script simulation.

# General Network Components

## Coupler

*Dialog: Net/ General/ Coupler*

The **Coupler** is a special element which implements an ideal transformer. Ideal transformers are used most frequently in modeling coupling of the mechanical to the acoustical domain by means of a diaphragm. Another common place where we find this element is within the model of the electro-static transducer.

The **Coupler** sets in relation the potentials and flows of the primary side to that of the secondary side.

If we use voltage **U** and current **I** in place of the potential and the flow as in Fig. 54 then the relation between the primary and secondary side is:

$$U_2 = \text{ratio} \cdot U_1$$

$$I_2 = -\frac{1}{\text{ratio}} \cdot I_1$$

In this formula **U** and **I** can be replaced by the force **F**, the velocity **v**, the pressure **p** and the volume velocity **V**, respectively. The primary and the secondary side need not to be of the same domain, you can easily mix the electrical with the mechanical or acoustical domain.

### Restrictions

With the **Coupler** there are no self-inductivity and no losses. Further there is no coupling factor, that means, there is no mutual inductance as in the **Transformer**-model. The **Coupler** model is totally different to that of the **Transformer**-model. The matrix solver of AkAbak processes the special case of the **Coupler** by adding rows and columns of the node-potential-matrix.

Because of the latter you can not connect **Couplers** in any arbitrary way. There are some restrictions in wiring multiple **Coupler**-elements directly together. There are no restrictions with other elements.

- The poles of the primary side of different **Couplers** can be connect together, in series or in parallel.
- One of the secondary pole has to be grounded.
- The non-zero secondary pole must not be directly connected with any other pole of a **Coupler** or with the driving point of the network (pole 1).

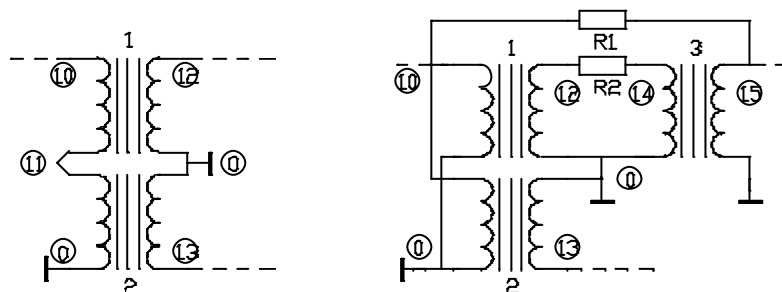


Fig. 55 Possible wiring of primary side. Left: serial, right: parallel.

The left and right hand sides of Fig. 55 demonstrate the correct wiring of multiple **Couplers**. For instance, if you remove the impedances in the right schematic, AkAbak would stop and output an error message.

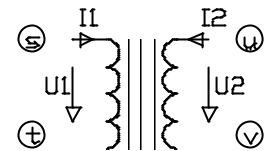


Fig. 54 Circuit diagram of the **Coupler** element.

## ☑ Examples

1. Left hand side of Fig. 55. This configuration creates a symmetric signal. At node 12 is the non-inverted signal and at node 13 you will find the inverted signal. `Coupler C2` is assigned a runtime formula for the sake of the example. That means the ratio of this `Coupler`-element is not only frequency dependent, but also complex. ' $f$ ' is the abscissa frequency. In this example the ratio will be imaginary for frequencies below 1kHz and becomes more and more real in the high frequency range.

```
...
Coupler  'Cpl1'   Node=10=11=12=0
  ratio=2

Coupler  'Cpl2'   Node=11=0=0=13
  ratio={ 1 + j1000/f }    | f is the abscissa frequency
...

```

2. The right hand side of Fig. 55 is:

```
...
Coupler  'Cpl1'   Node=10=0=12=0
  ratio=2

Coupler  'Cpl2'   Node=10=0=0=13
  ratio={ 4 + j1000/f }    | f is the abscissa frequency

Resistor 'R2'     Node=12=14
  R=1.5ohm

Coupler  'Cpl1'   Node=14=0=15=0
  ratio=2

Resistor 'R1'     Node=15=10
  R=2ohm
...

```

3. Schematic of a dynamical transducer in the way AkAbak uses it with its `Def_Driver` element.  $Z_e$  and  $Z_m$  are the frequency dependent impedances of the voice coil and the mechanical elements  $M_{ms}$ ,  $C_{ms}$  and  $R_{ms}$ , respectively.  $Bl$  is the force-factor or conversion factor of the motor. The mechano-acoustical coupling is performed by the front and rear side of the diaphragm.  $SD_f$  and  $SD_r$  are the areas of the diaphragm. The marked path-direction of the volume velocities  $V_f$  and  $V_r$  are related to the coupler and their flow is inverted in relation to the `Radiator` and `Enclosure` element.

When designing your own electro-mechano-acoustical dynamical drivers this schematic is usually the starting point of more complex circuits. For example if the diaphragm radiates into separated enclosures - or general speaking - into different acoustical paths then the `Coupler` element makes it possible to model the sub-diaphragms. The usual construction of the compression driver would be a good example for a more complex case (see chapter exercises).



```

Common parameter of the driver
Def_Const
{
    Reo=10;           fre=10e3;           ExpoRe=0.7;
    Leo=50e-6;        ExpoLe=0.6;
    Bl=10;
    Mms = 1.2e-3;  Rms = 1.7;           Cms = 33e-6;
    dD = 5e-2;      tD1 = 1e-2;
    fp = 10e3;
    SD = pi*sqr(dD/2);
}

|Definition only used for the second System 'Com'
Def_Driver 'Drv1'
    Meas_DoNotModify
    dD={dD}      tD1={tD1}      fp={fp}
    Re={Reo}      fre={fre}      ExpoRe={ExpoRe}
    Le={Leo}      ExpoLe={ExpoLe}
    Bl={Bl}
    Mms={Mms}    Rms={Rms}    Cms={Cms}

|Transducer as discretized element
System 'Dis'

|Voice coil
Impedance 'Ze' Node=1=2
Z={ Re_f = Reo*(1.0 + f/fre)^ExpoRe;
    wLe = w*Leo;          hh = sqr(wLe/Reo);
    r = (ExpoLe*hh + 1.0)/(hh + 1.0);
    Le_f = wLe^r;
    Ze = Re_f + j*Le_f; }

|Motor
Gyrator 'G1' Node=2=0=3=4
Bl={Bl}

|Mechanical part. Mms is frequency dependent.
|Dome diaphragm becomes a ring radiator at high frequencies
Impedance 'Zms' Node=3=5
Z={ dD1 = 0.8*dD/(1 + sqr(fp/f));
    SD_f = SD - pi*sqr(dD1/2);
    Zms = Rms + j*(w*Mms*SD_f/SD - 1/(w*Cms)) }

```

```

|Reverse side (concave dome) with enclosure
Coupler 'SD_r' Node=4=0=8
  dD={dD} tD1={-tD1} fp={fp}
Enclosure 'Vb' Node=8
  Vb=0.3L Lb=3cm

|Front side (convex dome) with radiation
Coupler 'SD_f' Node=5=0=7
  dD={dD} tD1={tD1} fp={fp}
|Radiation from 10cm diameter baffle
Radiator 'Rad_f' Def='SD_f' Node=7
  dEdge=10cm

|Transducer as compact element
System 'Com'
  Driver Def='Drv1'
    Node=1=0=7=8
  Enclosure 'E1' Node=8
    Vb=0.3L Lb=3cm
|Radiation
  Radiator 'Rad_f' Def='Drv1' Node=7
    dEdge=10cm

```

## Parameter

Coupler	Keyword
'...'	Identifier
Node=s=t=u=v	<p>The terminals of the primary side are between poles <b>s</b> and <b>t</b>. The terminals of the secondary side are either between <b>u</b> and ground or between <b>v</b> and ground. That means that one of the secondary Coupler poles has to be grounded always.</p> <p>Example:</p> <pre> Node=1=2=3 Node=1=2=0=3 </pre>
ratio=...	<p>The potential/flow ratio of primary and secondary side:</p> $U_2 = \text{ratio} \cdot U_1, \quad I_2 = -I_1 / \text{ratio}$
ratio={...}	<p>Alternatively you can specify a runtime formula system instead of a number. The syntax is the same as described in chapter Introduction/Formula Parser. In addition there are two variables which are the abscissa frequency: <b>f</b> and <b>w</b> (<math>=2\pi f</math>). The result of the last formula is assigned to <b>ratio</b>. The result might be real, imaginary or complex.</p>

Alternatively to enter the ratio directly, it is possible to specify it as a diaphragm:

ratio = 1/SD.

See 'Introduction/Diaphragm Forms' for a discussion of diaphragm dimension parameter. As demonstrated in the above example, the element **Radiator** can relate to the **Coupler** element via **Def='...'**.

dD=...m	Diameter of diaphragm.
WD=...m	(alternative) Width and height of rectangular diaphragm.
HD=...m	



SD=...m2	(alternative)	Area of circular diaphragm.
dD1=...m	(optional)	Diameter of inner diaphragm or hole.
SD1=...m2	(alternative)	Area of inner diaphragm or hole.
tD1=...m	(optional)	Depth or height of diaphragm form.
t1=...m	(optional)	Displacement of diaphragm within baffle.
f <sub>p</sub> =...Hz	(optional)	Area reduction control frequency.
Diffuse=...%	(optional)	Directivity control factor
See also:	Impedance	

## Gyrator

*Dialog: Net/ General/ Gyrator*

The four-pole element Gyrator transforms a flow into a potential.

### Electro-dynamical motor

It is ideal to describe a electro-dynamical motor. The Gyrator transforms the current into the force. It couples the electrical to the mechanical domain. The transmission characteristics are:

$$\begin{bmatrix} U1 \\ I1 \end{bmatrix} = \begin{bmatrix} 0 & -Bl \\ 1/Bl & 0 \end{bmatrix} \cdot \begin{bmatrix} F2 \\ v2 \end{bmatrix}$$

where

U1, I1:	voltage and current at input
F2, v2:	force and velocity at output
Bl=R:	conversion factor of the transducer or gyration constant

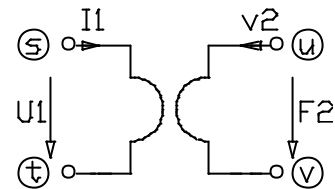


Fig. 57 Circuit diagram of Gyrator

### Flow source

The Gyrator can also be used to implement a current, velocity or vol. velocity source in the AkAbak network. Insert the Gyrator element as the first element in your node-list and if you want a 1:1 conversion omit the parameter 'R' and 'Bl'.

### ☒ Examples

An electro-dynamical transducer motor with a force-factor of 10 Tesla Meter.

```
Gyrator 'Gyr1' Node=2=0=3=0
Bl=10Tm
```

A flow source for a network.

```
Gyrator 'Gyr1' Node=1=0=2=0
```

## Parameter

Gyrator	Keyword
'...'	Identifier
Node=s=t=u=v	The input current flows from pole <b>s</b> to pole <b>t</b> . The output current from pole <b>u</b> to pole <b>v</b> .
B1=...Tm	Gyration constant as the force factor of the electro-dynamical motor. Unit: Tesla Meter [Tm]
R=...ohm	Gyration constant in [ohm].
See also	Coupler

## Impedance

Dialog: Net/ General/ Impedance

This two-pole element implements any impedance in the network. The driving point impedance is specified by a (complex) formula system. The syntax of this formula is identical with the syntax of the formula described in the introduction part of this manual (Introduction/ Formula Parser).

There is one little difference: The formula-system of the `Impedance` element has two more reserved identifiers:  $f$  and  $w$ . The formula system is compiled during the interpretation of the script, but the calculation takes place with each abscissa frequency. The value of the abscissa frequency is copied into  $f$  or  $\omega$ , respectively ( $\omega = 2\pi f$ ). You can use  $f$  and  $\omega$  in a mixed fashion as often as you need them in a formula or in a formula system.

The `Impedance` element can be an electrical, mechanical or acoustical element. Because of this, the `Impedance` element is listed in the list of each Inspect-simulation dialog if it has a name.

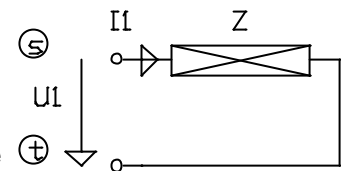


Fig. 58 Circuit diagram of the Impedance element.

### ☒ Examples

1. In this example `Impedance` comprises four parameter controlling the model of a mass-resistance-compliance resonator usually found as part of loudspeaker drivers. `Mms`, `Cms` and `Rms` are specified here as global constants at the top of script. In this way you are able to use them in other elements, too. In the formula the term  $(1 + \omega/\omega_0)$  makes the mass of the vibrating assembly frequency dependent. The effective mass becomes smaller at high frequencies. At 5kHz its value is the half of the specified value. 'j' is the imaginary unit. The last identifier 'Zms' within the formula could also be omitted. The schematic of this formula is:



```

Def_Const
{
  Mms=1e-3;    | Mechanical mass [kg]
  Rms=1;       | Mechanical resistance [Ns/m]
  Cms=70e-6;   | Mechanical compliance [N/m]
}
...
System 'S1'
...
Impedance 'Imp1' Node=100=110
Z={ wo = 2*pi*5000;
  Zms = Rms + j*(w*Mms/(1 + w/wo) - 1/(w*Cms))
}

```

```

    }
    ...

```

2. The effect of viscosity and thermal exchange in a duct leads to a frequency dependent resistance. In its approximation the resistance increases with  $Ra(f) \propto \sqrt{f}$ . The formula for viscosity and thermal exchange is given in chapter 'Acoustical Network Components/ AcouResistance'. Transferred to the AkAbak-formula-syntax an acoustical viscosity-resistance becomes:

```

...
Impedance  'Imp1'  Node=100=110
  Z={  WD=10e-2;      |Width of duct
      HD=1e-2;      |Height of duct
      L=30e-2;      |Length of duct
      roh=1.189;    |Density of air
      c=344;        |Velocity of sound in air
      nue=18.2e-6;  |dyn. viscosity
      gamma=1.402;  |specific heat of air @ 20° and 101kPa
      |Viscosity boundary layer factor
      dvo=sqrt(nue/(roh*pi));
      |Thermal boundary layer factor
      dho=sqrt(5*nue/(3*gamma*roh*pi));
      D=2*(WD + HD); |Perimeter
      S=WD*HD;      |Area
      k=w/c;        |Wave-factor
      Ra=roh*c*D*L/(2*sqr(S))*k/sqrt(f)*(dvo + (gamma-1)*dho);
  }
...

```

## Parameter

Impedance	Keyword
' ... '	Identifier
Node=s=t	The connection terminals of the impedance are between poles <b>s</b> and <b>t</b> .
Z={ ... }	<p>Driving point function of the impedance in form of the AkAbak formula (see chapter Introduction/Formula Parser). The formula system is written in braces (curly brackets) and can go over several lines. The identifier is <b>z=</b>.</p> <p>The result of the last formula is assigned to the impedance. The value of the result is usually complex and frequency dependent. The abscissa-frequency variable can be <b>f</b> or <math>\omega</math> (<math>= 2\pi f</math>)</p> <p>Comments inside the curly brackets are permitted. They commence with <b> </b> and end either with the same character or at the end of the line.</p> <p>Constants that have previously been computed in the definition <b>Def_Const</b> can also be assigned. The values of several <b>Impedance</b> or other elements can thus be changed centrally.</p>

See also:      Coupler

## Potential

*Dialog/ Net/ General*

This is a special element which inserts no impedance or any other function to the network. But when implemented across any two nodes you have access to the potential between these nodes. There will be some

interesting features in connection with the `Potential` element in future versions of AkAbak. For the time being use `Potential` elements when you construct `Filter`-networks by the `Feedback`-formula. Another need might be when you investigate special voltages, forces or pressures inside the network in a branch where no element is placed, for example across the terminals of a `Coupler`.

### ☑ Examples

To use potentials of the current-network with a `Filter`-network a `Potential` element is placed in parallel to `R1`. The abstract `Filter` element '`F2`' is feed by the voltage  $U12 = U1 - U2$ . Note that if the specification would be `Node=2=1` than  $U21 = -U12$  is measured.

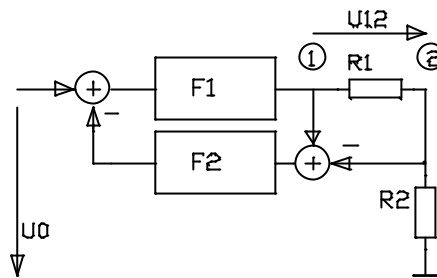


Fig. 59 Schematic of the feedback including network potentials

```
System 'Feed'
  Resistor 'R1'  Node=1=2  R=1ohm
  Resistor 'R2'  Node=2    R=1ohm
  Potential 'U12' Node=1=2
  Filter 'F1'
    fo=1000Hz {b0=1; a2=1; a1=1.414214; a0=1; }
  Filter 'F2'
    fo=1000Hz {b1=1}
  Feedback
    U1={F1*U0 - U12*F1*F2 }
```

## Parameter

Potential	Keyword
'...'	Identifier
Node=s=t	Potential across poles <b>s</b> and <b>t</b> . The potential is a voltage, force or pressure, respectively. $U_{st} = U_s - U_t$
See also:	Feedback

## Element (DLL)

*Dialog: Net/ General/ Element*

Network component of a user defined element where the interface is given by the definition `Def_Element`. The '`Element`' network component is linked to the definition via the definition name. Any number of '`Element`' components may refer to the same definition.

Depending on the definition in your DLL the network component has two, four or six poles.

## Parameter

Element	Keyword
'...'	Identifier
Def='...'	Reference to associated definition.
Node=s=t=u=v=w=x	Network poles of your element. A maximum number of 6 poles can be specified. The actual meaning and number of the poles depends on the type of your component. Note, that pair of poles make a port. The in- and output flow at a port must be independant of the in- and output flows of the other ports. The ports are s and t, u and v, w and x

## Driver

*Dialog: Net/ Transducer/ Driver*

Driver creates a compact model of an electroacoustic transducer in the network. The parameter are described in the associated definitions:

Def\_Driver

Def\_TwoCoilsDriver

Def\_PiezoDriver

The drivers are linked to the definition via the definition name. Any number of `Driver` elements may refer to the same definition.

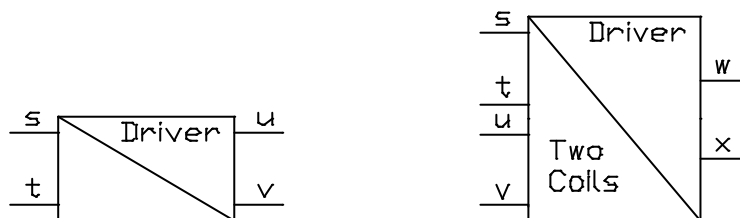


Fig. 60 Circuit diagram of the `Driver` element. Left: four-pole, right: six-pole.

If the `Driver` network element refers to the `Def_Driver` or `Def_PiezoDriver` definition, then the `Driver` element is a four pole driver (Fig. 60 left). If, on the other hand, it refers to the `Def_TwoCoilsDriver` definition, then the driver is six-poled. (Fig. 60 right).

The input poles `s` and `t` are in the electrical part of the network, the current flowing from pole `s` to pole `t`.

In the four-pole driver, the output poles `u` and `v` are in the acoustical part. In this case the volume velocity flows from pole `u` to pole `v`.

To pole `u` are connected all acoustical elements that are mounted on the front side of the diaphragm and have the same sound pressure at their input as across the diaphragm of the driver, for example the radiator or duct.

All elements at the reverse side of the diaphragm receive the number of the node lying at the pole `v`.

If the associated definition describes a loudspeaker already mounted in an enclosure, then, if the enclosure is mounted at the rear of the diaphragm, pole `v` must be grounded (0).

If `Driver` refers to `Def_TwoCoilsDriver`, then the driver element is a six-pole driver. Poles `u` and `v` are then also in the electrical network part and are the input of the second drive, in an analogous fashion to poles `s` and `t`.

Poles `w` and `x` are now in the acoustical part.

## ☑ Examples

1. The element refers to a `Def_Driver` or `Def_PiezoDriver` driver with the name 'Drv 1'. The positive pole of the driver is connected to node 1 and the other to ground (node 0). The diaphragm front is connected to node 2, its rear to node 3.

```
Driver Def='Drv 1'
Node=1=0=2=3
```

2. As the first example. However, the associated `Def_Driver` or `Def_PiezoDriver` definition describes a loudspeaker installed fixed in the enclosure, e.g. a dome tweeter. The rear side of the diaphragm radiates into an enclosure that (as always) has to be grounded (node 0).

```
Driver Def='Drv 1'
Node=1=0=2=0
```

3. The element refers to the `Def_TwoCoilsDriver` definition with the name 'Drv 1'. The positive pole of the voice coil 1 of the driver is connected to node 1 and the other pole (node 2) to the positive pole of the voice coil 2 (node 2). The two voice coils are thus connected in series.

The front side of the diaphragm is connected to node 3, the rear side to node 4.

```
Driver Def='Drv 1'
Node=1=2=2=0=3=4
```

## Parameter

Driver	Keyword
'...'	Identifier
Def='...'	<p>Reference to the associated definition:</p> <pre>Def_Driver, Def_TwoCoilsDriver or Def_PiezoDriver.</pre> <p>Exactly the same character sequence as for the name of the definition has to be entered between quotation marks ('...' or '...'). No distinction is made between upper and lower case characters.</p>
Node=s=t=u=v	<p>Four pole (<code>Def_Driver</code>, <code>Def_PiezoDriver</code>)</p> <p>The electrical connection terminals of the driver are between poles <b>s</b> and <b>t</b>. The phase position of the radiated sound is inverted if the poles <b>s</b> and <b>t</b> are exchanged.</p> <p>Pole <b>u</b> is the front of the diaphragm, pole <b>v</b> its rear.</p>
Node=s=t=u=v=w=x	<p>Six-pole (<code>Def_TwoCoilsDriver</code>)</p> <p>The electrical connection terminals of the voice coil 1 are between poles <b>s</b> and <b>t</b>, those of the 2nd voice coil between poles <b>u</b> and <b>v</b>. The phase position of the radiated sound is inverted if the poles <b>s</b> and <b>t</b> or <b>u</b> and <b>v</b> respectively are exchanged.</p> <p>Pole <b>w</b> is the front of the diaphragm, pole <b>x</b> its rear.</p>

## → Note

The network has to be grounded at both sides of the driver, i.e. the circuit must not 'float'. A loudspeaker connected, for example, via a resistor, to the power amplifier, radiating forwards and mounted in a sealed enclosure has the following network description:

```

System 'S1'
  Resistor Node=1=2
    R=1ohm
  Driver Def='Drv 1'
    Node=2=0=3=4 |Grounding at the electrical side
  Radiator Def='Drv 1'
    Node=3=0 |Grounding at the acoustical side
    x=0 y=0 z=0 HAngle=0 VAngle=0
  Enclosure
    Node=4=0 |Grounding at the acoustical side
    Vb=20L Qb/fo=0.1 Lb=10cm

```

See also: `Def_Driver`, `Def_TwoCoilsDriver`, `Def_PiezoDriver`

## MeasRadiator

*Dialog: Net/ Transducer/ MeasRadiator*

`MeasRadiator` creates in the network a `MeasRadiator` element whose parameter in this case are described by the measurement data file of the associated `Def_MeasRadiator` definition.

The measured input impedance appears across the poles **s** and **t**. The radiation is calculated from the transfer function of the radiation.

The entries for the diaphragm area and shape in the `Def_MeasRadiator` file, influence the directivity characteristic and the calculation of the diaphragm movement (see also chapters `Def/Def_MeasRadiator` and `Tools/Generate MeasRadiator File`).

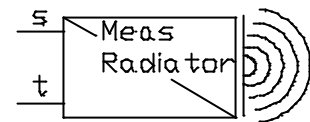


Fig. 61 Circuit diagram for the element `MeasRadiator`

If you do not enter a value for the cross section of the aperture, `AkAbak` treats `MeasRadiator` as a point source. In this case it is assumed that the directivity is already included in the measured SPL curve. In most cases there will be multiple files, each for a specific listening angle.

If the `Def_MeasRadiator`-file has no sound pressure curve then there will be no radiation but just an impedance.

### ☒ Example

The element refers to the definition `Def_MeasRadiator` with the name `Meas1`. A normal bass loudspeaker was measured. The positive pole of the driver is connected to node 1 and the other to ground (node 0). It is also displaced about 10 cm to the left (seen from the baffle) and 20 cm downwards. The diaphragm radiates forwards. In addition, if `Def_Reflector` is defined, it acts on the reproduction of this `MeasRadiator` element because the keyword `Reflection` is entered, the parameter of diffraction also has then to be entered.

```

Def_MeasRadiator 'Meas1'
  Filename='MBass00.ahr'
  | Filename='MBass30.ahr' Measurement under 30°
  | Filename='MBass60.ahr' Measurement under 60°

System 'S1'
  MeasRadiator Def='Meas1' Node=1=0
    x=10cm y=-20cm
    HAngle=0° VAngle=0°
    Reflection WEdge=30cm HEdge=50cm
  ...

```

## Parameter

MeasRadiator	Keyword
'...'	Identifier
Def='...'	Reference to the associated Def_MeasRadiator definition. Exactly the same sequence of characters have to be entered between quotation marks ('.' or '..') as for the name of Def_MeasRadiator. No distinction is made between upper and lower case characters.
Node=s=t	The connection terminals of the driver are between nodes <b>s</b> and <b>t</b> . The phase position of the radiated sound is inverted if nodes <b>s</b> and <b>t</b> are exchanged.

**See also chapter 'Introduction / Position of radiators, / Radiation environment':**

x=...m	Horizontal position.
y=...m	Vertical position.
z=...m	Axial position.
HAngle=...°	Horizontal position angle (-180°...180°)
VAngle=...°	Vertical position angle (-90°...90°)
Reflection	Optional. Takes part at evaluation of reflections.
WEdge=...m	Optional. Diffraction. Width of baffle.
HEdge=...m	Optional. Diffraction. Height of baffle.
dEdge=...m	Optional, alternative. Diameter of baffle.
NoRad	Optional. No radiation takes place.
NoDir	Optional. Radiates as a point source.

## Reflection and sound diffraction

It is, of course, only meaningful to enter the parameter `Reflection`, `WEdge=` etc. if the loudspeaker has been measured under baffle conditions.

## Mounting position

### Scalar measurement of SPL:

In contrast to the vector measurement the reference point of the mounting is not clearly defined. Because scalar measurement curves of the sound pressure have undergone the Hilbert transformation, the output parameter of a `MeasRadiator` is always free of delay. Although the reference level refers to a distance of one meter, the phase position relates to the so-called acoustic center of the radiator, on the assumption that the driver of the radiator itself is all-pass free. In most cases of diaphragm radiation this is true. `x=`, `y=`, `z=`, `HAngle=` and `VAngle=` thus relate to the acoustic center of the radiator. If there are standing wave patterns, reflections or the like the response will not be allpass-free (Waveguides, deep cone diaphragm etc.). In these cases the `MeasRadiator` produced by scalar sound pressure measurement should be used with great care.

With conical diaphragms, the acoustic center shifts backwards along the perpendicular to the diaphragm surface towards the dust cap. As a compromise, half the cone depth can be used, so that, for example,  $z = 2.5 \text{ cm} + z_0$  if the cone depth  $t_{D1} = 5 \text{ cm}$  and  $z_0$  is the rest of the mounting height in the z-direction.



With horn loudspeakers, the horn mouth is not anymore the reference point for mounting as in other cases, but the position of the diaphragm. If the distance between the diaphragm and the baffle is 32cm, then for  $z=-32\text{cm}$  is entered.

For the mounting angle,  $\text{HAngle=}$  and  $\text{VAngle=}$ , similar considerations apply. They rotate the radiator about the acoustic center.

### Vector measurement of SPL:

It is best to use vector measurement as 'level and phase' or 'real and imaginary parts' as sound pressure curve to create the `Def_MeasRadiator` file. There is usually a very high quality of reproduction even with horn-loudspeakers. The acoustic center is in this case defined by the measurement and part of the transmission.

See also:

`Def_MeasRadiator`, Tools/Generate MeasRadiator File

## Speaker

*Dialog: Net/ Transducer/ Speaker*

`Speaker` creates a network element whose parameter are described in the associated `Def_Speaker` definition. The `Speaker` element is used to describe only the position in the network and the radiation. The link to the definition is made by means of the name given to the definition. As many `Speaker` elements as required can refer to the same definition.

### ☒ Example

The element refers to the `Def_Speaker` definition with the name 'Sp1'. The positive pole of the driver is connected to node  $s=1$  and the other to ground (node  $t=0$ ). The center point of the diaphragm (suspension height) of the driver is displaced 10cm to the left (seen from the baffle), 20cm downwards and 5cm backwards. The diaphragm radiates forwards. In addition, any `Def_Reflector` defined has an effect on the reproduction of this `Speaker` element. If `Reflection` is entered, then the diffraction parameter also has to be entered.

```
Speaker  Def='Sp1'  Node=1=0
         x=10cm  y=-20cm  z=-5cm  HAngle=0°  VAngle=0°
         Reflection  WEdge=30cm  HEdge=50cm
```

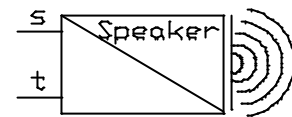


Fig. 62 Circuit diagram of the `Speaker` element

## Parameter

<code>Speaker</code>	Keyword
<code>'...'</code>	Identifier
<code>Def='...'</code>	Reference to the associated <code>Def_Speaker</code> definition. Exactly the same sequence of characters have to be entered between quotation marks ('.' or '..') as for the name of <code>Def_Speaker</code> . No distinction is made between upper and lower case characters.
<code>Node=s=t</code>	The connection terminals of the driver are between nodes <b>s</b> and <b>t</b> . The phase position of the radiated sound is inverted if nodes <b>s</b> and <b>t</b> are exchanged.

**See also chapter 'Introduction / Position of radiators, / Radiation environment'**

<code>x=...m</code>	Horizontal position.
<code>y=...m</code>	Vertical position.
<code>z=...m</code>	Axial position.
<code>HAngle=...°</code>	Horizontal position angle (-180°...180°)

VAngle=...°		Vertical position angle (-90°...90°)
Reflection	Optional.	Takes part in evaluation of reflections.
WEde=...m	Optional.	Diffraction. Width of baffle.
HEde=...m	Optional.	Diffraction. Height of baffle.
dEdge=...m	Optional,	Alternative. Diameter of baffle.
t1=...m	Optional.	Displacement of diaphragm within the baffle.
NoRad	Optional.	No radiation takes place.
NoDir	Optional.	Radiates as a point source.
See also:	Def_Speaker, Bassunit, Driver	

## Bassunit

*Dialog: Net/ Transducer/ Bassunit*

Bassunit creates an element in the network whose parameter are described in the associated Def\_Bassunit definition. With the element Bassunit only the position in the network and the radiation are described. The link to the definition is formed by means of the name given to the definition. Any number of Bassunit elements can refer to the same definition.

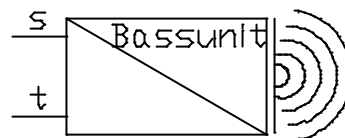


Fig. 63 Circuit diagram of the Bassunit element.

### ☒ Example

The element relates to the definition Def\_Bassunit with the name 'Bu1'. The positive pole of the driver is connected to node 1 and the other to ground. The center point of (suspension height) the diaphragm is, seen from the baffle, displaced 10 cm to the left and 20 cm downwards and 5 cm backwards. The diaphragm radiates forwards. In addition, any reflector defined Def\_Reflector) acts on the reproduction of this Bassunit element. If Reflection is entered, the diffraction parameter also has to be specified.

```
Bassunit      Def='Bu1'
Node=1=0
x=10cm  y=-20cm  z=-2cm  HAngle=0°  VAngle=0°
Reflection  WEde=30cm  HEde=50cm
```

## Parameter

Bassunit	Keyword
'...'	Identifier
Def='...'	Reference to the associated Def_Bassunit definition. Exactly the same character sequence as for the name of Def_Bassunit has to be entered between quotation marks ('...' or '...'). No distinction is made between upper and lower case characters.
Node=s=t	The connection terminals of the driver are between poles <b>s</b> and <b>t</b> . The phase position of the radiated sound is inverted if the poles <b>s</b> and <b>t</b> are exchanged.

**See also chapter 'Introduction / Position / Radiation environment':**

x=...m	Horizontal position
--------	---------------------

<code>y=...m</code>		Vertical position
<code>z=...m</code>		Axial position
<code>HAngle=...°</code>		Horizontal mounting angle (-180°...180°)
<code>VAngle=...°</code>		Vertical mounting angle (-90°...90°)
<code>Reflection</code>	Optional.	Taking part on reflections.
<code>WEdge=...m</code>	Optional.	Width of baffle.
<code>HEdge=...m</code>	Optional.	Height of baffle.
<code>dEdge=...m</code>	Optional,	Alternative. Diameter of baffle.
<code>tl=...m</code>	Optional.	Displacement within baffle.
<code>NoRad</code>	Optional.	No radiation takes place.
<code>NoDir</code>	Optional	Radiates as a point source.

### → Note

If `Def_Bassunit`, the definition forming part of the `Bassunit` element, describes a driver in the reflex enclosure, the entry for the position of the reflex vent is missing here. The reason is that the simplified model of `Def_Bassunit` uses the volume velocity inside the enclosure for radiation calculation, which - as can be demonstrated - is equal to the sum of the radiation of diaphragm and vent (within the scope of the simplified model). If the vent, measured at the wavelength, is distant from the diaphragm or if more accurate analyses are to be carried out, then the reflex enclosure is built up from the `Driver`, `Enclosure` or `Duct`'s and `Radiator`.

Please also note the effect of the `Bassunit` high-pass filter as described in chapter 'Def/Def\_Bassunit'.

See also: `Def_Bassunit`, `Speaker`, `Driver`

# Electrical Network Components

## Capacitor

*Dialog: Net/ Electric/ Capacitor*

Capacitor implements the model of a technical capacitor in the network (Fig. 64).  $C$  is the capacitance of the capacitor. You can optionally specify  $R_p$ ,  $R_s$  and  $L_s$  independently of one another. Although  $R_s$  and  $L_s$  are connected in series with the capacitance  $C$ , you do not require any additional network nodes.

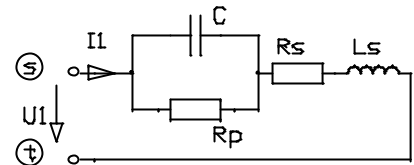


Fig. 64 Equivalent circuit diagram of the Capacitor element

Using  $R_p$ ,  $R_s$  and  $L_s$ , the Capacitor can be used to build up resonant circuits. Since, however,  $C$ ,  $R_p$ ,  $R_s$  and  $L_s$  are in this case combined in one component, you cannot investigate the internal currents and voltages. If you need to do this, you have to specify  $R_p$ ,  $R_s$  and  $L_s$  with their own nodes.

### ☒ Examples

1.: No losses

```
Capacitor      'C1'
Node=1=2      C=10uF
```

2.: With losses in the dielectric

```
Capacitor      'C1'
Node=1=2      C=10uF      Rp=50Mohm
```

3.: As the second example, but additionally with losses and stray inductance in the feeder line (resonant circuit).

```
Capacitor      'C1'
Node=1=2      C=10uF      Rp=50Mohm      Rs=0.5ohm      Ls=4uH
```

## Parameter

Capacitor		Keyword
'...'		Identifier
Node=s=t		Capacitor is between the poles <b>s</b> and <b>t</b>
C=...F		Capacitance in farad [F]
Rp=...ohm	(Optional)	loss resistance in parallel with C Unit: ohms [ohm]
Rs=...ohm	(Optional)	Loss resistance in series with the parallel circuit of C and Rp Unit: ohms [ohm]
Ls=...H	(Optional)	Stray inductance in series with the parallel circuit of C and Rp Unit: henry [H]
See also:		Coil, Resistor

# Coil

*Dialog: Net/ Electric/ Coil*

`Coil` implements the model of a technical coil in the network (Fig. 65).  $L$  is the inductance and  $R_s$  the loss resistance of the coil. Although  $R_s$  is in series with the inductance  $L$ , no additional network node is required. Since, however,  $L$  and  $R_s$  are combined in one component, you cannot investigate the internal currents and voltages. If you need to do this, you have to specify  $R_s$  with its own node.

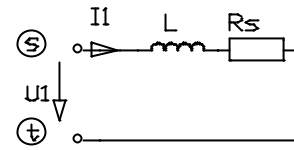


Fig. 65 Equivalent circuit diagram of the Coil element

You can simulate any stray coil capacitance present by connecting an appropriate capacitor in parallel. Since connecting in parallel does not increase the number of nodes, this option was dispensed with in the `Coil` element.

## ☒ Examples

1.: No losses

```
Coil 'L1' Node=1=2 L=5mH
```

2.: Losses due to the wire resistance and possibly due to the core material

```
Coil 'L1' Node=1=2 L=5mH Rs=2ohm
```

## Parameter

<code>Coil</code>		Keyword
<code>'...'</code>		Identifier
<code>Node=s=t</code>		Coil is located between the nodes <b>s</b> and <b>t</b>
<code>L=...H</code>		Inductance in henry [H]
<code>Rs=...ohm</code>	(Optional)	Loss resistance in series with the inductance $L$ Unit: ohm [ohm]

See also: Capacitor, Resistor

# Resistor

*Dialog: Net/ Electric/ Resistor*

`Resistor` implements an electrical resistance in the network.

## ☒ Example

```
Resistor 'R1' Node=1=2  
R=10kohm
```

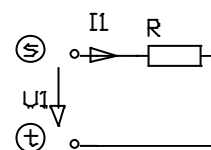


Fig. 66 Circuit diagram of the Resistor element

## Parameter

Resistor	Keyword
'...'	Identifier
Node=s=t	Resistance is between poles <b>s</b> and <b>t</b>
R=...ohm	Resistance value in ohm [ohm]
See also:	Coil, Capacitor

## Transformer

Dialog: Net/ Electric/ Transformer

This network element creates a transformer with two windings in the network. The primary winding is located at the poles **s** and **t** and the secondary winding at the poles **u** and **v**. The transmission characteristics of the transformer are given by:

$$\begin{bmatrix} U1 \\ U2 \end{bmatrix} = \begin{bmatrix} s \cdot L1 + R1 & s \cdot k \cdot \sqrt{L1 \cdot L2} \\ s \cdot k \cdot \sqrt{L1 \cdot L2} & s \cdot L2 + R2 \end{bmatrix} \cdot \begin{bmatrix} I1 \\ I2 \end{bmatrix}$$

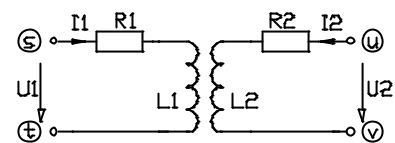


Fig. 67 Equivalent circuit diagram of the Transformer element

$U1$  and  $U2$  are the voltages across the primary and secondary windings.  $I1$  and  $I2$  are the respective currents to the transformer.  $L1$  and  $L2$  are the winding inductances and  $R1$  and  $R2$  their internal resistances, which you can optionally specify.  $s$  is the complex frequency variable. The two coils are connected together via the coupling factor  $k$ .

The coupling factor is given in per cent in this case. You cannot simulate an ideal transformer using the **Transformer** element. If you need ideal coupling, please, use the **Coupler** element. The coupling factor may therefore only have values in the range  $0 < k < 100\%$ . In practice, values between 90% and 99% can be reached.

The expression  $k \cdot \sqrt{L1 \cdot L2}$  is equal to the so-called mutual inductance  $M$ . The transmission factor of an ideal transformer ( $k=100\%$ ) is equal to:

$$U1/U2 = \ddot{u} = \pm \sqrt{L1/L2} = \pm n1/n2$$

where  $n1$  and  $n2$  are the primary and secondary windings respectively.



### Examples

- 1.: The primary winding is between nodes **s=2** and **t=0** and the secondary winding is between nodes **u=3** and **v=0**. The primary winding has an inductance of  $L1=10\text{mH}$  and the inductance of the secondary winding  $L2=1\text{mH}$ . The coupling factor is  $k=95\%$ .

```
Transformer 'Tr1'
Node=2=0=3=0    L1=10mH  L2=1mH  K=95%
```

- 2.: In a similar manner to the first example, Entry in the script as provided by the 'transformer' dialog (see below). The primary winding has an internal resistance of  $R1=10\text{ ohm}$  and the secondary winding an internal resistance of  $R2=0.1\text{ ohm}$ .

The entry  $A1=100$  is not used in the simulation but in the dialog as a reminder of the value.

```
Transformer  'Tr1'
Node=2=0=3=0  | Turns: n1=316, n2=100, n1/n2=3.16
L1=10mH  R1=10ohm  k=95%  A1=100
L2=1mH   R2=0.1ohm
```

## Parameter

Transformer	Keyword
'...'	Identifier
Node=s=t=u=v	The primary winding is at poles <b>s</b> and <b>t</b> and the secondary winding at poles <b>u</b> and <b>v</b> . The input and output current here flow into the transformer.
L1=...H	Inductance of the primary winding. Unit: henry [H].
L2=...H	Inductance of the secondary winding. Unit: henry [H].
R1=...ohm (Optional)	Internal resistance of the primary winding. R1 is in series with L1. Unit: ohm [ohm].
R2=...ohm (Optional)	Internal resistance of the secondary winding. R2 is in series with L2. Unit: ohm [ohm].
k=...%	Coupling factor of the two inductances L1 and L2. k is a measure of the directivity of the magnetic field. The few lines of force of one coil penetrate the other, the smaller k is. Because of the nature of the analysis, the coupling must not be total ( $k < 100\%$ ). With shell cores, annular cores and the like, and especially if the windings are superimposed, the value is in the range $k = 95\%..99\%$ .

## OpAmp

Dialog: Net/ Electric/ OpAmp

OpAmp creates an operational amplifier in the network. Fig. 68 shows the equivalent circuit diagram. An OpAmp amplifies the voltage via the non-inverting (+) and inverting (-) inputs.

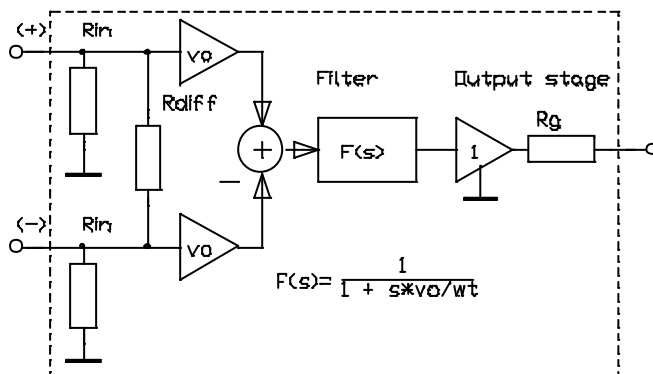


Fig. 68 Equivalent circuit diagram of the OpAmp element

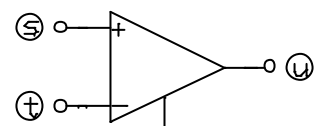


Fig. 69 Circuit diagram OpAmp

## Generator resistance Rg

The analysis used here requires the internal resistance  $R_g$  of the OpAmp output stage (generator resistance).  $R_g$  usually has a very low value (1ohm...500ohm). If the data sheet does not quote it directly, you can usually obtain it from the diagrams or measure it: While retaining a fixed input voltage, you measure the output voltage first without the load resistance  $U_o$  and then the output voltage  $U_L$  with a resistance of  $R_L=1\text{kohm}...2\text{kohm}$ . You can then obtain the generator resistance from the two voltages according to the formula  $R_g = R_L \cdot (U_o/U_L - 1)$ .

If you do not specify  $R_g$  the program uses the default value  $R_g=1\text{ohm}$ .

## Open-loop gain vo and transition frequency ft

The open-loop gain  $v_o$  is frequency dependent and has a low-pass character  $F(s)$ . The -3 dB cut-off frequency is usually about 10Hz. The amplitude decay is at first 6dB/oct and at very high frequencies 12dB/oct. OpAmp operates reliably in the frequency range in which the gain is greater than one and the phase rotation is less than 180°. This frequency range is called unity gain bandwidth or transition frequency  $f_t$ . If the transition frequency  $f_t$  is specified, the model used here takes into account a pole point of the low-pass filter inherent in OpAmp, i.e. the range with 6dB cut-off of the open-loop gain.

The default value for  $v_o=10^6$  and the specification of the transition frequency  $f_t$  is optional.

## Input resistance

There are two types of input resistance. Firstly that between the inverting and non-inverting ( $R_{diff}$ ) and the resistance of an input to ground ( $R_{in}$ ).  $R_{in}$  is usually much larger than  $R_{diff}$ . The data sheets usually only give the input resistance  $R_e$  that occurs when an input is connected to ground  $1/R_e = 1/R_{diff} + 1/R_{in}$ . For high frequencies, the input resistances become complex. You can simulate this using appropriate capacitances. The specification of the input resistances is optional.

## Using the definition Def\_OpAmp

When you use a lot of OpAmp's of the same type, it is convenient to enter the OpAmp-parameter only once in the definition Def\_OpAmp and relate to it with the help of the Def='...' parameter (see chapter Def/ Def\_OpAmp). The network list of the active filter synthesis tool uses this technique. For example:

```
Def_OpAmp  'Op1'
  vo=1e5  Rg=100ohm  Rin=100kohm  ft=3MHz

System  'S1'
  ...
  OpAmp  'Op11'  Def='Op1'  Node=100=101=101
  ...
  OpAmp  'Op12'  Def='Op1'  Node=200=201=201
  ...
```

## No parameter given

In many cases or initially only a generic type of OpAmp (or Def\_OpAmp) can be used. Then, when only the name and the node numbers are specified, the program sets default values for the amplification and for the generator resistance:  $v_o=1e6$  and  $R_g=1\text{ohm}$ . For example:

```
OpAmp  'Op11'  Node=100=101=101
```

### ☒ Examples

- 1.: FET-OpAmp (TL071) with open-loop gain of  $v_o=200 \cdot 10^3$ , a generator resistance of  $R_g=250\text{ohm}$ . The transition frequency (unity-gain bandwidth) is  $f_t=3\text{MHz}$  (at  $f_t$  the modulus of the open-loop gain  $v_o=1$ ). The resistance



between the inverting and non-inverting input  $R_{diff}$  and the resistance of the two inputs with respect to ground  $R_{in}$  is  $1\text{Tohm}$  ( $=10^{12}\text{ohm}$ ).

```
OpAmp    'Op1'
Node=3=4=5  vo=200e3  Rg=250ohm
ft=3.0MHz  Rdiff=1Tohm  Rin=1Tohm
```

2.: Transistor-OpAmp (NE5532) with feedback-free gain of  $vo=50 \cdot 10^3$ , a generator resistance of  $R_g=0.5\text{ohm}$ . The transition frequency is  $ft=10\text{MHz}$ . The differential driving-point resistance is  $300\text{kohm}$ .

```
OpAmp    'Op1'
Node=3=4=5  vo=50e3  Rg=0.5ohm
ft=10MHz  Rdiff=300kohm
```

3.: In this example, a normal output amplifier is defined as OpAmp. Since such amplifiers have internal feedback, you cannot use the transition frequency in this case. For output amplifiers, the so-called damping factor is often given, from which you can calculate  $R_g$ : Attenuation factor =  $R_L/R_g$ , where  $R_L$  is the nominal load resistance (e.g.  $R_L=8\text{ohm}$ ),  $R_g$  is the OpAmp generator resistance.

```
OpAmp    'Op1'
Node=3=4=5  vo=10  Rg=0.1ohm
Rdiff=10kohm
```

4.: Circuit of an integrator with the integration elements  $R_1$ ,  $C_1$  at the inverting OpAmp input ( $t=2$ ). The non-inverting input (+) is connected to ground via the resistor  $R_2$ .  $R_2=R_1$  to minimize the offset voltage.

```
Resistor  'R1'  Node=1=2  R=10kohm
Capacitor 'C1'  Node=2=4  C=16nF
OpAmp     'Op1' Node=3=2=4  vo=100e3  Rg=10ohm
Resistor  'R2'  Node=3=0  R=10kohm  |to Ground
```

5.: Circuit of a non-inverting amplifier stage with a gain of one. The inverting input (-) ( $t=2$ ) is connected to the output stage ( $u=2$ ).

```
OpAmp    'Op1'  Node=1=2=2  vo=100e3  Rg=10ohm
```

## Parameter

OpAmp	Keyword
'...'	Identifier
Def='...'	Reference to the associated Def_OpAmp definition. Exactly the same character sequence as for the name of the definition has to be entered between quotation marks ('...' or '...'). No distinction is made between upper and lower case characters.
Node=s=t=u	Pole <b>s</b> is non-inverting (+), pole <b>t</b> is inverting (-) and pole <b>u</b> is the OpAmp-output.
vo=	Open-loop gain at very low frequencies (without feedback). Default value: $vo=1e6$
Rg=...ohm	Open-loop generator resistance, i.e. the internal resistance of the OpAmp output stage. Default value: $R_g=1\text{ohm}$ Unit: ohm [ohm]

$f_t = \dots \text{Hz}$	(Optional)	Transition frequency of the open-loop gain $v_o$ . At $f_t$ , $v_o=1$
$R_{diff} = \dots \text{ohm}$	(Optional)	Resistance between non-inverting and inverting input Unit: ohm [ohm]
$R_{in} = \dots \text{ohm}$	(Optional)	Resistance of non-inverting and inverting input to ground Unit: ohm [ohm]

## Slew rate, large-signal bandwidth or power bandwidth

The slew rate is the maximum rate of rise of the output voltage. Since the internal capacities of the OpAmp can only be recharged with limited currents, the output voltage cannot change at an arbitrarily fast rate. The OpAmp responds to a discontinuous input voltage of any magnitude with a ramp-shaped output voltage, whose gradient is the slew rate SR. The same effect limits the linear gain of sinusoidal signals with maximum amplitude  $U_{max}$  towards high frequencies. The frequency  $f_p$  at which a sinusoidal signal with  $U_{max}$  can just be transmitted without distortion is called the large-signal bandwidth or power bandwidth.

$$f_p \approx \frac{SR}{2\pi \cdot U_{max}}$$

If the frequency  $f$  exceeds the power bandwidth  $f_p$ , the amplitude  $U$  has to be reduced according to the relationship:  $U \leq U_{max} \cdot f_p/f$  so that the slew rate.[FlI] is not exceeded.

In the simulation, you can easily observe the output voltages of the OpAmps. In filter circuits, in particular, considerable increases in voltage occur in the vicinity of the pole frequencies of the filter blocks. These voltage increases have to be smaller than  $U_{max}$ . They are proportional to the quality of the filter block.

## Transistor

*Dialog: Net/ Electric/ Transistor*

Small-signal Hf-model of a transistor in the network. Apart from the bipolar transistor, this model also represents the field effect transistor (FET) and the valves. No separate identifiers are provided for the FET and the valves. You always enter the parameter via the identifiers of the bipolar transistor.

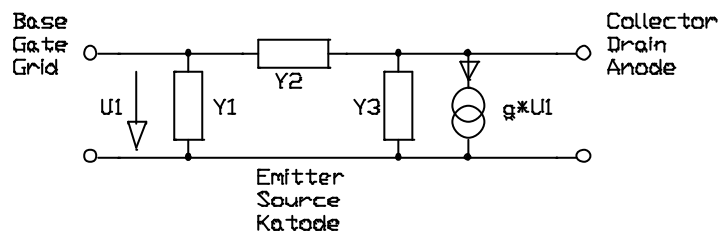


Fig. 70 Equivalent circuit diagram of the bipolar transistor, field-effect transistor and the valves

In the equivalent circuit diagram shown in Fig. 71,  $Y_1$  is the input admittance, which tends to zero in the case of a FET and valves in the NF range.  $Y_2$  is the so-called reverse transfer admittance. You can usually neglect  $Y_2$  in the NF range.  $Y_3$  is the output admittance.  $g$  is the forward transconductance.

### No parameter given

In many cases or initially only a generic type of transistor can be used. Then, when only the name and the node numbers are specified, the program sets a default value for the  $h_{21e}$  parameter:  $h_{21e}=100$ . For example:

```
Transistor 'Q11' Node=100=0=101
```

### Using the definition

When you use a lot of Transistor's of the same type it is more convenient to enter the transistor -parameter only once in the definition Def\_Transistor and relate to it with the help of the Def='...' parameter. For example:

```
Def_Transistor 'Q1'
  h11e=3.5kohm  h12e=2e-4  h21e=270  h22e=32uS
  ft=200MHz     Ccb=3.5pF

System 'S1'
...
Transistor 'Q11'  Def='Q1'  Node=100=101=101
Transistor 'Q12'  Def='Q1'  Node=200=201=201
...
```

### Mixed specification

In some cases it is very practical to be able to overriding some of the parameter of the Def\_Transistor, for instance for testing purpose, etc. For example:

```
Def_Transistor 'Q1'
  h11e=3.5kohm  h12e=2e-4  h21e=270  h22e=32uS
  ft=200MHz     Ccb=3.5pF

System 'S1'
...
Transistor 'Q11'  Def='Q1'  Node=100=101=101
  h21e=100          |overriding h21e
```

## Bipolar transistor

Pole **s** is the base, pole **t** the collector and pole **u** the emitter of the transistor. The hybrid four-pole matrix in the emitter circuit **He** determines the frequency-independent parameter of the transistor:

$$\begin{bmatrix} \partial U_{BE} \\ \partial I_C \end{bmatrix} = \begin{bmatrix} h_{11E} & h_{12E} \\ h_{21E} & h_{22E} \end{bmatrix} \cdot \begin{bmatrix} \partial I_B \\ \partial U_{CE} \end{bmatrix}$$

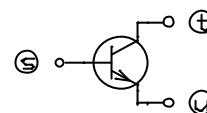


Fig. 71 Circuit diagram of the bipolar transistor

The frequency-dependent characteristic of the transistor is simulated using the transition frequency  $f_t$  of the differential current gain  $h_{21E} = \beta$  and the collector-base capacitance  $C_{cb}$ .

The parameter form the admittances of the equivalent circuit diagram of Fig. 71:

$$Y_1 = \frac{1}{h_{11e}} + s \cdot C_D \quad \text{where } C_D = \frac{h_{21e}}{2\pi \cdot f_t \cdot h_{11e}} \text{ (diffusion capacitance)}$$

$$Y_2 = -\frac{h_{12e}}{h_{11e}} + s \cdot C_{cb} \quad \text{where } C_{cb}: \text{ collector-base capacitance}$$

$$Y_3 = h_{22e} = 1 / r_{CE} \quad \text{where } r_{CE}: \text{ output resistance}$$

$$g = \frac{h_{21e}}{h_{11e}} \quad \text{where } g: \text{ forward transconductance in siemens [S]}$$

$s$ : complex frequency variable

☒ Example

Transistor BC107A in the collector circuit (collector to ground, nodes t=0).

Operating point:  $U_{CE}=5V$ ,  $I_C=2mA$ ,  $f=1kHz$

```
Transistor 'Q1' | BC107A
Node=1=0=2
h11e=8.7kohm h12e=1.5e-4
h21e=220 h22e=18uS
ft=200MHz Ccb=3.5pF
```

## Parameter

Transistor		Keyword																
'...'		Identifier																
Def='...'		Reference to the associated Def_Transistor definition. Exactly the same character sequence as for the name of the definition has to be entered between quotation marks ('...' or '...'). No distinction is made between upper and lower case characters.																
Node=s=t=u		<table><tr><td>Pole:</td><td><b>s</b></td><td><b>t</b></td><td><b>u</b></td></tr><tr><td>bipolar:</td><td>base</td><td>collector</td><td>emitter</td></tr><tr><td>FET:</td><td>gate</td><td>drain</td><td>source</td></tr><tr><td>valves:</td><td>grid</td><td>anode</td><td>cathode</td></tr></table>	Pole:	<b>s</b>	<b>t</b>	<b>u</b>	bipolar:	base	collector	emitter	FET:	gate	drain	source	valves:	grid	anode	cathode
Pole:	<b>s</b>	<b>t</b>	<b>u</b>															
bipolar:	base	collector	emitter															
FET:	gate	drain	source															
valves:	grid	anode	cathode															
h11e=...ohm		<p>Equal to the differential base-emitter resistance <math>r_{BE}</math>.</p> $h_{11e} = r_{BE} = \frac{\partial U_{BE}}{\partial I_B} = \frac{h_{21e}}{g}$ <p>If <math>I_C</math> is the collector current at the operating point, then:</p> $h_{11e} = \frac{h_{21e} \cdot U_T}{I_C} \text{ where } U_T \approx 25mV \text{ 'temperature voltage'}$ <p><math>h_{11e}</math> is a resistance and you have to specify it for the bipolar transistor. In the case of the FET and the valves, <math>h_{11e}</math> is symbolically set to zero or not specified. Unit: ohm [ohm].</p>																
h12e=...	(Optional)	<p>Equal to the voltage reaction <math>A_r</math>. The absolute value of <math>h_{12e}</math> is often less than <math>10^{-4}</math> and you can usually neglect it (<math>h_{12e}=0</math>). At higher frequencies, however, the reaction increases. You can take account of this phenomenon by specifying the collector-base capacitance <math>C_{cb}</math>.</p>																
h21e=...		<p>Equal to the differential current gain <math>\beta</math>.</p> $h_{21E} = \beta = \frac{\partial I_C}{\partial I_B} \bigg _{U_{CE} = \text{const}}$ <p>In the case of the FET or the valves, <math>h_{21e}</math> is equal to the forward transconductance <math>g</math> standardized to one siemens. <math>h_{21e}</math> is unitless and you have to specify it.</p>																
h22e=...S	(Optional)	<p>Approximately equal to the differential output admittance:</p> $h_{22E} \approx \frac{1}{r_{CE}}$																

$$r_{CE} = \frac{\partial U_{CE}}{\partial I_C} \bigg|_{U_{BE}=\text{const}} = \frac{U_Y}{I_C}$$

where  $U_Y$  is the so-called early voltage

( $U_Y=80\ldots 200\text{V}$  for npn transistors,

$U_Y=40\ldots 150\text{V}$  for pnp transistors).

$I_C$ : collector current at the operating point.

$h_{22e}$  is an admittance and is usually very small compared with the external admittances, so that you can neglect it. Unit: siemens [S].

$f_t = \dots \text{Hz}$	(Optional)	Transition frequency of the differential current gain $\beta = h_{21e}$ . At $f_t$ , $h_{21e}$ has a value of one. The data sheets sometimes also give the -3 dB cut-off value of $h_{21e}$ : $f_t \approx h_{21e} \cdot f_{-3\text{dB}}$ . In the case of the FET or of the valves, $f_t$ is the transition frequency of the forwards transconductance $g$ . Unit: hertz [Hz].
$C_{cb} = \dots \text{F}$	(Optional)	Collector-base capacitance. In the case of FET, $C_{cb}$ represents the value of the drain-gate capacitance and in the case of the valves, the value of the anode-grid capacitance. Unit: farad [F].

#### → Note

The internal base-bulk resistance in the equivalent circuit diagram of Giacoleto is not taken into account here. If the transistor is driven from a low-resistance source, you may have to insert this externally, so that the frequency characteristic can be correctly reproduced from the diffusion capacitance.

## Field-effect transistor

The same is true in principle for the FET or MOSFET as for the bipolar transistor. No separate identifiers are available for describing the field effect transistor. The transistor hybrid parameters are used for data input, even though, from the point of view of the physics, this procedure is not quite correct.

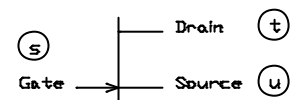


Fig. 72 Circuit diagram of the field-effect transistor

### Connections

The gate is connected to pole **s**, drain to pole **t** and source to pole **u**.

### Driving-point resistance ( $h_{11e}$ )

The parameter  $h_{11e}$  is not used, since the driving-point resistance of a FET tends to infinity.  $h_{11e}$  is not specified in the program or is symbolically set to zero.

### Transconductance ( $h_{21e}$ )

The parameter  $h_{21e}$  is used as the transconductance  $g$ , whose value in this case is standardized to the conductance of one siemens, and is thus unitless.

### Output conductance ( $h_{22e}$ )

In this case  $h_{22e}$  corresponds to the reciprocal of the drain-source output resistance  $r_{DS}$ .

### Frequency dependence

The frequency-dependent characteristic of the FET is simulated with the aid of the transition frequency  $f_t$ , the forwards transconductance  $g$  and the drain-gate capacitance, whose value is assigned to the identifier  $C_{cb}$ .

The transition frequency is:

$$f_t = \frac{g}{2\pi \cdot C_{GS}} \quad \text{where } G_{GS}: \text{gate-source capacitance}$$

### Equivalent circuit diagram (Fig. 72)

$$Y1 = s \cdot C_{GS} \quad \text{where } G_{GS} = \frac{g}{2\pi \cdot f_t} \quad (\text{gate-source capacitance})$$

$$Y2 = s \cdot C_{DG} \quad \text{where } C_{DG}: \text{drain-gate capacitance (corresponds to } C_{cb})$$

$$Y3 = h_{22e} = 1/r_{DS} \quad \text{where } r_{DS}: \text{drain-source resistance}$$

s: complex frequency variable

## Valves

The small-signal behavior of the triodes or pentodes is similar to the FET. No separate identifiers are provided to describe the valve parameter. The transistor hybrid parameter are used for the data input, even though, from the point of view of the physics, this is not entirely correct.

### Connections

The gate of the valve is connected to pole **s**, drain to pole **t** and source to pole **u**.

### Driving-point resistance (h11e)

The parameter h11e is not used, since the driving-point resistance of a valve tends to infinity. h11e is not specified in the program or is symbolically set to zero.

### Transconductance (h21e)

The parameter h21e is used as the transconductance g of the valve, whose value in this case is standardized to the conductance of one siemens, and is thus unitless

### Output conductance (h22e)

In this case h22e corresponds to the reciprocal of the valve resistance  $r_i$ .

### Frequency dependence

The frequency-dependent characteristic of the valve is simulated with the aid of the transition frequency  $f_t$ , the forward transconductance g and the anode-grid capacitance, whose value is assigned to the identifier Ccb.

The transition frequency is:

$$f_t = \frac{g}{2\pi \cdot C_{GK}} \quad \text{where } G_{GK}: \text{mesh-cathode capacitance}$$

### Equivalent circuit diagram (Fig. 73)

$$Y1 = s \cdot C_{GK} \quad \text{where } G_{GK} = \frac{g}{2\pi \cdot f_t} \quad (\text{grid-cathode capacitance})$$

$$Y2 = s \cdot C_{AG} \quad \text{where } C_{AG}: \text{anode-grid capacitance (corresponds to } C_{cb})$$

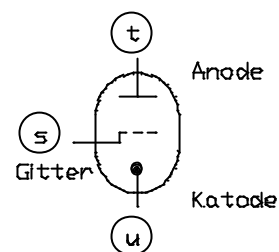


Fig. 73 Circuit diagram of the valves

---

$$Y_3 = h_{22e} = 1/r_i \quad \text{where } r_i: \text{ valve internal resistance}$$

s: complex frequency variable

# Mechanical Network Components

In many cases where more complicate structures are to be modeled, the general component `Impedance` would be also a good alternative (see chapter 'General Network Components').

## MechCompliance

*Dialog: Net/ Mechanic/ MechCompliance*

This two-pole element is used to install a mechanical compliance - i.e. a spring - in the network.

This element is a fundamental mechanical module. It is intended for use in theoretical analyses and experiments or for planning a mechanical structure.

The compliance is treated like a capacitance if, as in the program, the current corresponds to the velocity  $v$  and the voltage to the force  $F$ .

In the frequency domain, the force difference  $F$  across the compliance is equal to

$$F = v / (s \cdot C_m)$$

$s$ : complex frequency

$v$ : velocity in meters per second

$F$ : force in Newton.

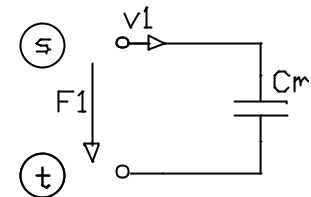


Fig. 74 Equivalent circuit diagram of the MechCompliance element

The compliance is a constant  $C_m = \partial x / \partial F$  with the dimension meter per newton. It is a reciprocal of a spring stiffness.

Note that in contrast to the acoustical compliance the mechanical compliance has not to be grounded because the force is an absolute value and not as the sound pressure related to the absolute pressure.

### ☒ Examples

```
MechCompliance    'Cms'    Node=100=110
Cm=2e-3m/N
```

## Parameter

<code>MechCompliance</code>	Keyword
<code>'...'</code>	Identifier
<code>Node=s=t</code>	The velocity $v$ flows from pole $s$ to pole $t$ .
<code>Cm=...m/N</code>	Value of the mechanical compliance Unit: meter/newton [m/N]
<code>Rm=...Ns/m</code> (optional)	Value of a mechanical series resistance Unit: Newton times second/meter [Ns/m]
See also:	<code>Impedance</code> , <code>MechMass</code> , <code>MechResistance</code>



# MechMass

Dialog: Net/ Mechanic/ MechMass

This two-pole element is used to install a mass in the network.

This element is a fundamental mechanical module. It is intended for use in theoretical analyses and experiments or for planning one's own mechanical structures.

The mass is treated like an inductance if, as in the program, the current corresponds to the velocity  $v$  and the voltage to the force  $F$ .

In the frequency domain, the force difference  $F$  across the mass is equal to

$$F = v \cdot s \cdot Mm$$

$s$ : complex frequency

$v$ : velocity in meters per second

$F$ : force in Newton.

$Mm$ : mass in kilo-gram

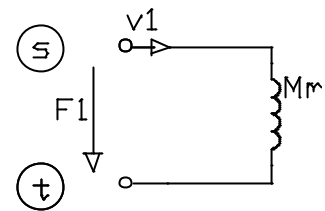


Fig. 75 Equivalent circuit diagram of the MechMass element



## Examples

```
MechMass      'Mms'      Node=100=110
               Mm=50g
```

## Parameter

MechMass	Keyword
'...'	Identifier
Node=s=t	The velocity $v$ flows from pole <b>s</b> to pole <b>t</b> .
Mm=...kg	Value of the mechanical mass Unit: kilo-gram [kg]
See also:	Impedance, MechCompliance, MechResistance

# MechResistance

Dialog: Net/ Mechanic/ MechResistance

This two-pole element is used to install a mechanical resistance in the network.

This element is a fundamental mechanical module. It is intended for use in theoretical analyses and experiments or for planning one's own mechanical structures.

The mechanical resistance is treated like a electrical resistance if, as in the program, the current corresponds to the velocity  $v$  and the voltage to

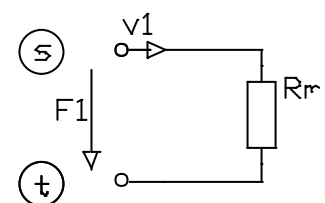


Fig. 76 Equivalent circuit diagram of the MechResistance element

the force  $F$ .

The force difference  $F$  across the resistance is frequency independent and equal to

$$F = v \cdot R_m$$

$v$ : velocity in meters per second

$F$ : force in Newton.

$R_m$ : resistance in Newton second/meter

### ☒ Examples

```
MechResistance    'Rms'    Node=100=110
Rm=1.5Ns/m
```

## Parameter

MechResistance	Keyword
'...'	Identifier
Node=s=t	The velocity $v$ flows from pole <b>s</b> to pole <b>t</b> .
Rm=...Ns/m	Value of the mechanical resistance Unit: Newton times second/meter [Ns/m]
See also:	Impedance, MechCompliance, MechMass

## Note

As with the acoustical resistance the mechanical resistance becomes frequency dependent in many cases. Then the `Impedance` element would be an alternative (see chapter Net/ General/ Impedance).

# Acoustical Network Components

## AcouCompliance

*Dialog/ Acoustic /AcouCompliance*

This two-pole element is used to install an acoustic compliance in the network. This element is used for equivalent circuit diagrams of acoustic structures if the wavelength is very large compared with the dimensions. This element is a fundamental acoustic lumped element. It is intended for use in theoretical analyses and experiments or for planning one's own mechanical or acoustic structures.

The following, somewhat elaborate discussion of 'compliance' is also intended to aid understanding of more compact models that are better suited to practice, for example the elements *Duct*, *Waveguide* and *Enclosure*.

The acoustic compliance is treated like a capacitance if, as in the program, the current corresponds to the volume velocity **V** and the voltage to the pressure **p**.

In the frequency domain, the pressure difference **p** across the acoustic compliance is equal to  $p = V/(s \cdot Ca)$ , in which **s**: complex frequency, **V**: volume velocity of the gas in cubic meters per second, weighted with the flow area.

The acoustic compliance is a constant with the dimension meter<sup>3</sup> per pascal. It is a reciprocal of a spring stiffness based on the square of a surface. To understand the acoustic compliance, imagine an air volume being compressed, but not moved, by a force. In other words, compression without acceleration indicates acoustic compliance. The other case forms part of the acoustic mass (see *AcouMass*).

The classical element representing acoustic compliance is a duct or pipe closed at one end and filled with air or another gas, so that the velocity of the air particles here is zero. In the frequency range in which the diameter of the duct is in the range from  $0.1/\sqrt{f} \leq dD \leq 20/f$ , the acoustic impedance of such a duct is<sup>4</sup>:

$$Z_a = -j \frac{\rho \cdot c}{SD} \cdot \cot(kL)$$

where

$\rho, c$	gas density and velocity of sound
$SD$	cross-sectional area of the duct
$k$	wave factor $k = \omega / c$
$L$	length of the duct

For values of  $kL$  that are not too high, the cotangent can be approximated by the first of the two summands of the series:

$$\cot(kL) = \frac{1}{kL} - \frac{kL}{3} - \frac{(kL)^3}{45} - \dots$$

The acoustic impedance  $Z_a$  then becomes:

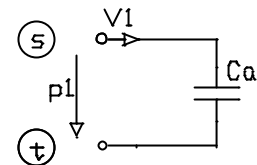


Fig. 77 Equivalent circuit diagram of the AcouCompliance element

<sup>4</sup>This limits are fairly strict. The lower limit is due to viscosity and thermal exchange. The upper frequency is the end of the one-dimensional approach. E.g. a tube of circular cross section has its first higher mode at about  $f_{1,1} = 200/dD$  which is 10 times the value given above.

$$Z_a \approx -j \frac{1}{\omega \cdot (V_b / (\rho \cdot c^2))} + j\omega \frac{L \cdot \rho}{3 \cdot S D} = \frac{1}{j\omega \cdot C_a} + j\omega \cdot M_a'$$

where

$V_b$ : volume of the closed duct  
 $C_a$ : acoustic compliance of  $V_b$   
 $M_a'$ : acoustic mass

This approximation applies for closed ducts with a length  $L < \lambda/7$ . The compliance  $C_a$  is accordingly:

$$C_a = \frac{V_b}{\rho c^2}$$

The source of the sound is then loaded with a mass  $M_a'$ :

$$M_a' = \frac{1}{3} \cdot \frac{L \cdot \rho}{S D}$$

Comparing this mass with that of an open duct  $M_a$  (`AcouMass`), then  $M_a'$  is a third of  $M_a$ . The equivalent circuit diagram is shown in Fig. 78.

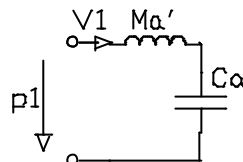


Fig. 78 Equivalent circuit diagram of a closed duct at low frequencies

The `AcouCompliance` element, however, is not intended to simulate acoustic ducts. That function is performed much better by the elements `Duct` and `Enclosure`, both of which apply the formula using the cotangent or tangent function. Fig. 79 compares the reactance of a compliance to a closed duct. At the eigenfrequencies the duct acts like a mass. The diagram shows also the curve of the above mentioned approximation of Fig. 78.

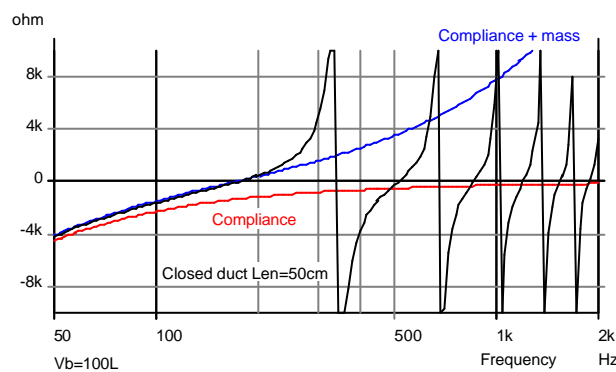


Fig. 79 Comparison of reactances forming enclosures

Acoustic compliances are two-pole elements, one of the poles always being grounded. This concept is derived from the model of the covered pipe. Acoustic compliances can thus not be found floating in an equivalent circuit diagram.



#### Examples

1.: At node 5 there is an acoustic compliance whose volume corresponds to  $V_b = 2 \text{ mm}^3$ . It is calculated with the aid of the equation parser.

---

```
AcouCompliance 'Ca1' Node=5=0
Ca={ Vb=2e-9; roh=1.2; c=344; Vb/(roh*sqr(c)) }
```

2.: The same as in the first example, except that Ca is entered directly here. As you can see, this results in very small numbers. If these fall below numerical range of the interpreter, use the inline equation parser as in Example 1, since the numerical quantity is very much larger in this case.

```
AcouCompliance 'Ca1' Node=5=0
Ca=14.084e-15m3/Pa
```

## Parameter

AcouCompliance	Keyword
'...'	Identifier
Node=s=t	The volume velocity $V$ flows from pole <b>s</b> to pole <b>t</b> , and it should be taken into account that one of the nodes here always has to be grounded.
Ca=...m <sup>3</sup> /Pa	Value of the acoustic compliance $Ca = \frac{V_b}{\rho \cdot c^2}$ where: V <sub>b</sub> : volume, ρ: gas density, c: velocity of sound. Unit: meter <sup>3</sup> /Pascal [m <sup>3</sup> /N]

## Damping material

If the volume of the acoustic compliance is filled with a special material the acoustic compliance  $Ca$  is found to have increased by a factor of 1.2...1.4. What is the reason for this?

If a gas is compressed very rapidly, its temperature increases and vice versa; when it relaxes, its temperature is reduced. For frequencies in the audio range, the air changes its state adiabatically as a result of the compression and decompression by the sound wave. The following relationship exists between pressure  $p$  and volume  $V_b$ :

$$p \cdot V_b^\chi = \text{constant}$$

where  
 $\chi$                       adiabatic exponent (for air:  $\chi=1.4$ )

The equation for the compliance can also be given as:

$$Ca = \frac{V_b}{\chi \cdot p_0}$$

where  
 $p_0$                       air pressure of the normal atmosphere ( $p_0=10\text{kPa}$ )

If the volume  $V_b$  is filled uniformly with a material that can compensate for the rapid temperature changes of the sound wave by absorbing and re-emitting the heat, the changes of state are more or less isothermal. The adiabatic exponent is then reduced to one. Damping material thus has multiple effects. Beside changing the thermal transfer it inserts losses at higher frequencies and risens the effective mass.

See also:                      MechCompliance, Impedance,  
                                     AcouMass, AcouResistance,  
                                     Enclosure

# AcouMass

Dialog: Net/ Acoustic/ AcouMass

This two-pole element is used to install an acoustic mass in the network. Use this element for equivalent circuit diagrams of acoustic structures if the wavelength is large compared with the dimensions.

This element is a fundamental acoustic module. Acoustic masses are found whenever an air volume is displaced. The counterpart is pure compression, which can be described by the element *AcouCompliance*. In modeling acoustic structures, the acoustic mass is mainly used for taking into account the effects of obstructions, bends, etc.

The *AcouMass* element is also intended for theoretical experiments and analyses or for modeling natural mechanical or acoustic structures. The following, somewhat more detailed treatment of the acoustic mass is also intended to aid an understanding of the more compact, and often more practical, models, such as *Duct*, *Waveguide* and *Enclosure*.

The acoustic mass is treated as an inductance, if, as in the program, the current corresponds to the volume velocity **V** and the voltage to the pressure **p**.

In the frequency domain, the pressure difference **p** across the acoustic mass **Ma** is  $p = s \cdot Ma \cdot V$ , where **s**: complex frequency, **V** volume velocity of the gas in cubic meters per second, weighted with the flow area.

The classical element representing an acoustic mass is an open duct or pipe and filled with air or another gas. Its dimensions are significantly smaller than the wavelength. In the frequency range in which the diameter of the duct is in the range  $0.1/\sqrt{f} \leq dD \leq 20/f$ , the acoustic impedance of such a duct is<sup>5</sup>:

$$Z_a = j \frac{\rho \cdot c}{SD} \cdot \tan(k \cdot L)$$

where

$\rho, c$	gas density and velocity of sound
SD	cross-sectional area of pipe
k	wave factor $k = \omega / c$
L	length of pipe

For very low frequencies ( $L < \lambda/16$ ), the tangent series after the first term can be terminated and  $Z_a$  becomes  $Z_a = j\omega \cdot Ma$ , where  $Ma$  is the acoustic mass:

$$Ma = \frac{M}{SD^2} = \frac{\rho \cdot L}{SD}$$

where

M	the air mass in the duct
---	--------------------------

Strictly speaking, the acoustic mass of the duct is somewhat greater. At both ends of the duct, a small portion of air covibrates. This portion is called the air load, and it represents the frequency dependent imaginary component of the radiation impedance which is usually a complicate function depending on the aperture form and on the radiation environment.

At very low frequencies below the directivity frequency  $f_D = c/(\pi \cdot dD)$  of the radiation surfaces of the duct an approximation of the air load of a circular aperture can be given.

The air load for one side is:

$$Ma' = 0.54 \cdot \rho / dD \approx 0.65 / dD \quad \text{The duct terminates at a large baffle.}$$

$$Ma' = 0.39 \cdot \rho / dD \approx 0.5 / dD \quad \text{The duct terminates freely.}$$

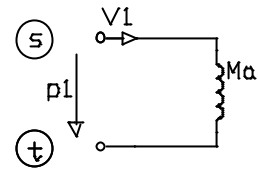


Fig. 80 Equivalent circuit diagram of *AcouMass*

<sup>5</sup>See footnote at *AcouCompliance*.

$Ma'$  should then be included for each side. The procedure is called end correction. In extreme cases the acoustic mass may consist only, or predominantly, of the end correction, for example in the case of a simple hole in a thin wall.

With increasing frequency, the wavelength becomes smaller and the given equations for  $Ma$  lead to a large error. If the acoustic structure represented by  $AcouMass$  is actually tubular, then it is better to replace it with the  $Duct$  element, which calculate the formula for  $Z_a$  using the tangent.

Acoustic masses occur not only in passages and ducts, but also in duct systems if the cross-section changes or obstructions are present. The holes of meshes or insulating materials have, like covibrating material, an acoustic mass.



### Examples

The following formulae are approximations for the frequency range in which the wavelength is considerably greater than the acoustic structure. If, for example, you want to gain an impression of the effect of a constriction, a bend, etc. these examples form a suitable basis for an estimation.

1. Two ducts ( $Duct$ ) with the same cross-section have between them a thin wall with a hole of diameter  $dL$  (Fig. 81). Let the area of the hole be much smaller than the cross-sectional area of the duct. The dimensions of the duct are also much smaller than the wavelength. According to [Mor1] the acoustic mass is given by

$$Ma \approx \rho / dL \quad \text{where } \rho: \text{air density.}$$

Thus, with  $\rho = 1.2 \text{ kg/m}^3$  and  $dL = 2 \text{ cm}$  this results in  $Ma = 60 \text{ kg/m}^4$ . In the equivalent circuit diagram,  $Ma$  is in series with the output of the first duct and the input of the second.

```
AcouMass    'Ma1'    Node=4=5
Ma=60kg/m4
```

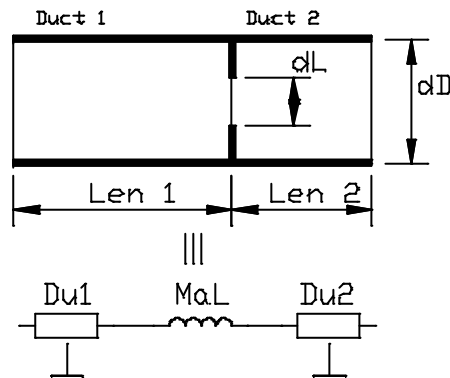


Fig. 81 Acoustic ducts with constriction

2. Two ducts with different cross-sectional area are abutted. The rectangular cross-sectional area enlarges, with both duct parts have the same depth  $t$ . The smaller duct has the height  $H_b$  and the greater the height  $H_a$  (Fig. 82). The acoustic mass for the transition is [Mor1]:

$$Ma = \frac{\rho}{\pi \cdot t} \cdot \left( \frac{(H_a - H_b)^2}{2H_a \cdot H_b} \ln \frac{H_a + H_b}{H_a - H_b} + \ln \frac{(H_a + H_b)^2}{4H_a \cdot H_b} \right)$$

Thus, with  $\rho = 1.2 \text{ kg/m}^3$ ,  $t = 5 \text{ cm}$ ,  $H_a = 5 \text{ cm}$  and  $H_b = 3 \text{ cm}$  this results in  $Ma = 1.9 \text{ kg/m}^4$ . In the equivalent circuit diagram,  $Ma$  is in series with the output of the first duct and the input of the second duct.

```
AcouMass    'Ma1'    Node=4=5
Ma=1.9kg/m4
```



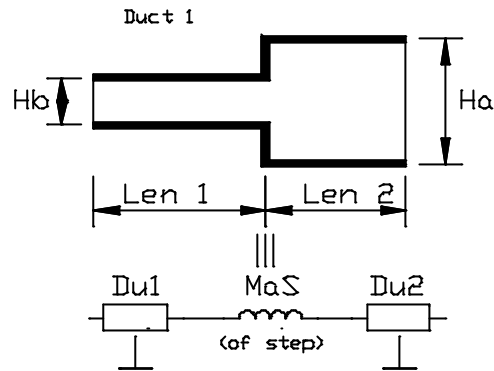


Fig. 82 Acoustic duct with step-shaped enlargement

3. Fig. 83 shows a 90° bend in the duct path. If the dimensions of the duct do not change, i.e. the cross-sectional area before and after the bend are the same, then it is advisable to insert between the two Duct elements an acoustic mass MaB with a value of approximately

$$Ma \approx \frac{1.85}{HD}$$

Where HD is the height of the Duct in meters. The mass is dependent only on the cross-section in the plane of the bend. The acoustic mass increases with the angle, and the sharpness of the bend. In addition there is an acoustic resistance. The above formula gives only a rule of thumb or a good starting point for estimation. As Fig. 83 demonstrates, the duct length is measured only to the beginning of the bend since the acoustical mass is the actual boundary condition for duct 1 and duct 2.

```
...
Duct 'Du1' Node=5=6 WD=20cm HD=5cm Len=30cm
AcouMass 'Ma1' Node=6=7 Ma={ 1.85/0.05 }
Duct 'Du1' Node=7=8 WD=20cm HD=5cm Len=30cm
...
```

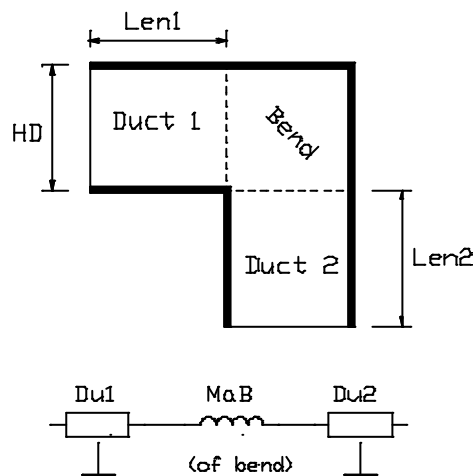


Fig. 83 Acoustic duct with 90°-bent

## Parameter

AcouMass

Keyword

'...'

Identifier

Node=s=t

The volume velocity flows from pole s to pole t.

$Ma = \dots \text{kg/m}^4$

Value of the acoustic mass.

$$Ma = \frac{Mm}{S^2}$$

$Mm$ : mass of the air in the duct

$S$ : cross-sectional area

Unit: kilogram/meter<sup>4</sup>

See also:

`MechMass`, `Impedance`,  
`AcouCompliance`, `AcouResistance`

## AcouResistance

*Dialog: Net/ Acoustic/ AcouResistance*

This two-pole element is used for equivalent circuit diagrams of acoustical structures if the volume velocity vibrates in phase with the sound pressure. Because in practice most of the acoustical resistance is frequency dependent `AcouResistance` will be used only in narrow band analysis or to check the sensibility and effect of a system to a resistive part anywhere in the structure.

This shall also be the place in the manual where the acoustical resistance is introduced and discussed in detail. Instead of implementing many identifiers for each special case of acoustical resistances, AkAbak offers the general and easy to use element `Impedance`. That means if you need to specify more complex models of acoustical impedances then the `Impedance` element would be the best and most flexible tool (see chapter 'General Network Components/ Impedance').

The acoustic resistance is treated like an electrical resistance, if, as in the program, the current corresponds to the volume velocity  $V$  and the voltage to the pressure  $p$  (Fig. 84).

In the frequency domain, the pressure difference  $p$  across the acoustic resistance  $R_a$  is equal to  $p = R_a \cdot V$ , where  $V$ : volume velocity of the gas in cubic meters per second, weighted with the flow area.

The element `AcouResistance` may represent any acoustic element within which the volume velocity vibrates in phase with the sound pressure. In other words, no energy is stored with the sound flow. In practice five basic forms of acoustic resistor are distinguished:

- fine wire or textile meshes
  - thin slits
  - thin ducts (capillaries)
  - porous acoustic material
  - obstructions and bends in sound conductors (Duct, Waveguide)
- meshes are often used in acoustic structures. They provide a resistance value that can be easily measured and are largely free of acoustic mass.
  - slits are used if the value of the resistance is required to be variable.
  - thin ducts produce a very high acoustic resistance. However, these have an acoustic mass.
  - porous acoustic materials act as small Helmholtz resonators. Apart from a high acoustic resistance, they additionally have an acoustic mass or compliance, depending on the level of the frequency.
  - obstructions and bends in sound conductors have an acoustic resistance that depends on the geometry and the sharpness of the corners. Sharp corners also increase turbulence in the flow. In addition, they always generate a greater or lesser acoustic mass.

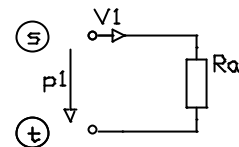


Fig. 84 Equivalent circuit diagram for element `AcouResistance`

In all five forms, the resistance is caused by adjacent layers of gas molecules with different velocities rubbing against one another.

Fig. 85 compares the two opposite sides 1 and 2 of a slit, a duct, or a wire mesh. The sound pressure  $p_2 - p_1$  is generated by the movement of air molecules in the space between walls 1 and 2. Directly at the walls, the molecules do not demonstrate any movement. Maximum movement occurs in the center between the walls. Molecule layers traveling at different rates are close together. This friction is greatest close to the walls and then decreases rapidly towards the center.

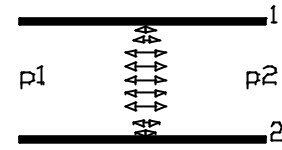


Fig. 85 shows the reduction of the vibration rate of the air molecules in a sound wave in the vicinity of a surface as a consequence of the viscosity of air

The effect of viscosity and thermal exchange in a duct leads to a frequency dependent resistance in its first approximation. The resistance increases with  $Ra(f) \propto \sqrt{f}$ . After Morse [Mor]:

$$Ra(f) = \frac{\rho \cdot c \cdot D \cdot L}{2 \cdot S^2} [k \cdot d_v + (\gamma - 1) \cdot k \cdot d_h]$$

with

$\rho$ : Density [kg/m<sup>3</sup>]

$c$ : Sound velocity [m/s]

$L$ : Length of duct

$D$ : Perimeter of cross-section [m]

$S$ : Area of cross-section [m<sup>2</sup>]

$k$ : Wave-factor  $k = 2\pi f / c$

$\gamma$ : Ratio of specific heat  $\gamma = c_p / c_v$  (air @ 20° C, 101.325 kPa,  $\gamma = 1.402$ )

$c_p, c_v$ : Specific heat capacity at constant pressure/volume [J/(kg·K)]

$d_v$ : Viscosity boundary layer [m]:

$$d_v = \sqrt{\frac{2 \cdot \eta}{\rho \cdot \omega}} = 2.2 \cdot 10^{-6} / \sqrt{f} \text{ [m] @ 20° C}$$

with  $\eta$ : dyn. viscosity [Pa·s] ( $\eta = 18.2 \cdot 10^{-6}$  Pa·s, air @ 20°)

$d_h$ : Thermal boundary layer [m]:

$$d_h = \sqrt{\frac{2 \cdot K}{\rho \cdot \omega \cdot c_p}}, \text{ with } K: \text{thermal conductivity [W/(m·K)]},$$

because  $K \approx 5 / 3 \cdot \eta \cdot c_v \rightarrow$

$$d_h = \sqrt{\frac{10 \cdot \eta}{3 \cdot \gamma \cdot \rho \cdot \omega}} = 2.4 \cdot 10^{-6} / \sqrt{f} \text{ [m] @ 20° C}$$

$Ra(f)$ : Acoustical resistance

In many cases the thickness of the boundary layers of viscosity and of thermal exchange can set to approximately the same values and the viscosity formula for the resistance becomes:

$$Ra(\omega) = \frac{D \cdot L \cdot \eta}{S D^2} \cdot \sqrt{\eta \cdot \rho \cdot \omega}$$

Whenever the room in which the sound wave propagates is not significantly larger than  $2d_v$ , the acoustic resistance increases dramatically. As can be seen in the formula of  $d_v$ , the thickness of the viscosity layer is diminished if the frequency increases.

If there is curving in the sound guide and turbulence, the effect is similar. In this case, too, local air layers of different velocities, as a result of mutual friction, increase the losses, that is to say the acoustic resistance.

## Some practical values of acoustic resistances<sup>6</sup>

### Acoustic resistances of meshes

Number of wires per inch	Wire diameter [mm]	Specific acoustic resistance Rs [Pa s/m]
30	0,33	5,67
50	0,22	5,88
100	0,115	9,1
120	0,092	13,5
200	0,057	24,6

The specific resistance Rs measured still has to be divided by the effective surface area S, to obtain the acoustic resistance Ra:  $R_a = R_s / S$ .

### Acoustic resistance of capillary ducts

In the frequency range in which the radius of the capillary is less than  $0,002 / \sqrt{f}$ , the acoustic impedance is:

$$Z_a = \frac{8\eta \cdot L}{\pi \cdot R^4} + j\omega \cdot \frac{4}{3} \cdot Ma$$

where

L length of the capillary [m]

R radius of the capillary [m]

Ma acoustic mass [ $\text{kg/m}^4$ ], as described for the element `AcouMass`.

The real component of  $Z_a$  is then the acoustic resistance  $R_a$ .

### Acoustic resistance of a thin slit

For the frequency range in which the height of the slit is less than  $H < 0,003 / \sqrt{f}$ , the acoustic impedance is:

$$Z_a = \frac{12\eta \cdot L}{H^3 \cdot W} + j\omega \cdot \frac{6\rho \cdot L}{5W \cdot H}$$

where

L depth of slit [m]

H height of the slit ( $H \ll W$ ) [m]

W width of the slit [m]

The real component of  $Z_a$  is the acoustic resistance  $R_a$ .

#### ☒ Example

Acoustic resistance of a wire mesh as covering of the vent of a reflex speaker. From the table above: 120 wires per inch, 0.092mm wire diameter,  $R_s = 13,5 \text{ Pa s/m}$ .

The vent diameter is  $d_D = 10 \text{ cm}$ .

The vent cross-sectional area is:  $S_D = \pi \cdot d_D^2 / 4$ .

---

<sup>6</sup>After Beranek [Ber]

The acoustic resistance can thus be calculated as:

$$R_a = R_s / S D = 13,5 \text{ Pa s/m} / 0,00785 \text{ m}^2 = 1719 \text{ Pa s/m}^3.$$

The acoustic resistance due to the viscosity of this vent is approximately  $R_a = 375 \text{ Pa s/m}^3$  (length=20cm, frequency=1kHz).

The resistance of the wire mesh is 5 times higher than the resistance due to viscosity.

```
AcouResistance 'Ra1' Node=5=6
Ra=1719 Pa s/m3
```

## Parameter

AcouResistance	Keyword
'...'	Identifier
Node=s=t	The acoustic resistance is between the poles <b>s</b> and <b>t</b> .
Ra=...Pas/m3	Value of the acoustic resistance. Unit in Pascal seconds per meter <sup>3</sup> [Pa s/m <sup>3</sup> ].
See also:	Impedance, AcouMass, AcouCompliance, Duct

## Diaphragm

*Net/ Acoustic/ Diaphragm*

The **Diaphragm** network element describes a passively vibrating diaphragm which is coupled to the acoustical plane via the diaphragm area.

The **Diaphragm** element cannot itself radiate. To do that it must be followed by a **Radiator** element.

Since a **Diaphragm** element does not radiate, it is necessary to free the parameter entered, such as  $f_s$ ,  $Q_{ms}$  etc., of the real and imaginary components of the radiation impedance, because the radiation resistance is given by the elements such as **Radiator**, **Enclosure** etc. connected to front and rear sides of the diaphragm. Unlike the **Driver** element, for example, the program assumes in this case that the parameter are given without radiation impedance. The parameter are here determined by the diaphragm manufacturer or by physical considerations.

On one hand, this element is used in loudspeaker chassis that comprise only the diaphragm and do not have their own motor. They may replace, for example, the vent of a reflex enclosure or be mounted at the end of long ducts.

On the other hand electroacoustic drivers can be built up using **Diaphragm**, for example electrostatic or other drivers.

For the elements of the equivalent circuit diagram in Fig. 87:

$$f_s = \frac{1}{2\pi \cdot \sqrt{M_{ms} \cdot C_{ms}}}, \quad Q_{ms} = \frac{1}{R_{ms}} \cdot \sqrt{\frac{M_{ms}}{C_{ms}}}$$

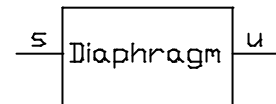


Fig. 86 Circuit diagram of the Diaphragm element

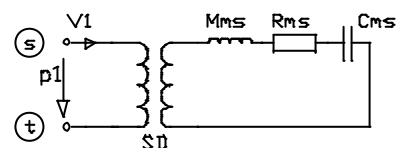


Fig. 87 Equivalent circuit diagram of the Diaphragm element

where (see also parameter):

Mms:	diaphragm mass [kg]
Cms:	mechanical compliance of the diaphragm suspension [N/m]
Rms:	mechanical resistance of the diaphragm suspension [Ns/m]
fs:	resonance [Hz]
Qms:	mechanical quality
SD:	diaphragm surface area [m <sup>2</sup> ]

If you specify the parameter fp, the diaphragm diameter and mass Mms become frequency dependent (see chapter "Introduction/Diaphragm parameter").

### ☒ Example

This Diaphragm element has the name 'PD1'. Pole **s** is at node 40 and pole **t** at node 50. The diaphragm is flat (Piston) and has a diameter of 20cm. The resonance frequency is 20Hz. The mass of the vibrating parts is 150g. The mechanical quality is Qms=3.

```
Diaphragm  'Diaph1'  Node=40=50
           dD=20cm | Piston
           fs=20Hz   Mms=150g  Qms=3
```

## Parameter

Diaphragm	Keyword
'...'	Name of the element. Required if the Radiator element with the identifier Def='...' relates to Diaphragm. Range: Any characters, but at most 20.
Node=s=t	Pole <b>s</b> represents one side of the diaphragm and pole <b>t</b> the other.

### Parameter of transducer see 'Introduction-Transducer'

fs=...Hz	Resonance frequency <b>without</b> airload.
Mms=...kg	Mass of diaphragm assembly <b>without</b> airload.
Vas=...m <sup>3</sup> (Alternative)	Equivalent compliance-volume.
Cms=...m/N (Alternative)	Mechanical compliance of suspension.
Qms=...	Mechanical quality factor <b>without</b> airload.
Rms=...Ns/m (Alternative)	Mechanical resistance of driver suspension.

### Parameter of diaphragm dimensions see 'Introduction-Diaphragm Forms'

dD=...m	Diameter of diaphragm.
WD=...m (Alternative)	Width and height of rectangular diaphragm.
HD=...m	
SD=...m <sup>2</sup> (Alternative)	Area of circular diaphragm.
dD1=...m (Optional)	Diameter of inner diaphragm.
SD1=...m <sup>2</sup> (Alternative)	Area of inner diaphragm.

<code>tDl=...m</code>	(Optional)	Depth or height of diaphragm form.
<code>t1=...m</code>	(Optional)	Displacement of diaphragm within baffle.
<code>fp=...Hz</code>	(Optional)	Mass and area reduction control frequency.
<code>Diffuse=...%</code>	(optional)	Directivity control factor

See also: `Coupler`, `Radiator`

## Note

The `Diaphragm` element is not an element of the mechanical domain because its inherent coupling. To build a mechanical model of a vibrating diaphragm use the mechanical elements `MechMass`, `MechCompliance` and `MechResistance` or for more sophisticated design use the `Impedance` element.

## Radiator

*Dialog: Net/ Acoustic/ Radiator*

`Radiator` is a network element that installs a radiation impedance in the network. and radiates into free space.

The `Radiator` element forms the link between the acoustic elements that guide the sound in closed systems and the free radiation room in which the listening point is located.

The radiation acts at every point in the radiation room, even on the surface of the diaphragm of the radiator. This feedback of the acoustic pressure, together with the diaphragm velocity, gives the radiation impedance. (see chapter 'Introduction/ Radiation').

When the program calculates the radiation and the radiation impedance, it takes into account the shape and effects of the mass reduction of the diaphragm, the sound diffraction at the edges, the baffle and the effect of reflectors.

The `Radiator` element is required whenever a network element does not radiate, for example the `Driver` or `Duct` element.

A typical construction is the `Driver - Radiator - Enclosure` - combination., a loudspeaker in the enclosure. The `Radiator` is connected using the first output node of the four pole `Driver` (node `u`) if the radiation is from the front side of the diaphragm. The enclosure (`Enclosure`) is connected to the reverse side of the diaphragm (node `v`). If `Enclosure` describes a reflex enclosure, there is a second radiation source, namely that of the reflex opening. No `Radiator` element is required here, since the `Enclosure` element already provides all radiation functions for the reflex vent. Other elements with built-in radiation functions are `BassUnit`, `Speaker`, `Enclosure`, `Horn`.

The parameter of a `Radiator` element are the entries for the radiation area and radiation form. You can enter these values directly (e.g. with `dD=` etc.), or by means of a reference.

When entering the values directly, take care that there are no unintended sudden changes in area. This can easily happen after the diaphragm area of the driver or the like has been changed.

If you have indicated the data for the radiation area by means of a reference, the program, while interpreting the script, fetches the data from the element with the name entered after `Def='...'`. This element may be a `Driver`, a `Duct`, a `Diaphragm`, a `Waveguide`, a `Coupler` or else a `Def_Driver`, `Def_TwoCoilsDriver` or `Def_PiezoDriver`. If the diaphragm area or the cross-sectional area of the reference element has changed, the radiation area of the radiator element also changes automatically. It is

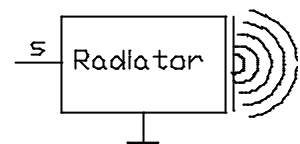


Fig. 88 Circuit diagram of the Radiator element

important that the reference element comes before the `Radiator` element and is installed in the same network.

If both are entered, then the program ignores the direct entry.

### ☑ Examples

1. The typical arrangement, already discussed, of a loudspeaker in the sealed enclosure (Fig. 89).

```
...
System 'S1'
Driver 'D1'      Def='IAK-1'
Node=1=0=2=3
Radiator      Def='D1'
Node=2
x=0 y=0 z=0
HAngle=0 VAngle=0
Enclosure Node=3
Vb=40L Qb/fo=0.1 Lb=20cm
```

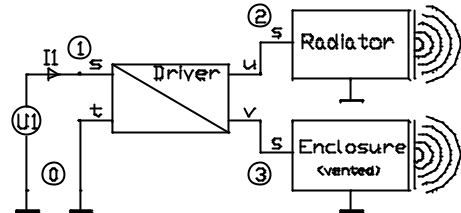


Fig. 89 Circuit diagram for example 1

2. As Example 1, but the loudspeaker radiates forwards through a long acoustic duct. (Fig. 90, see also example Duct).

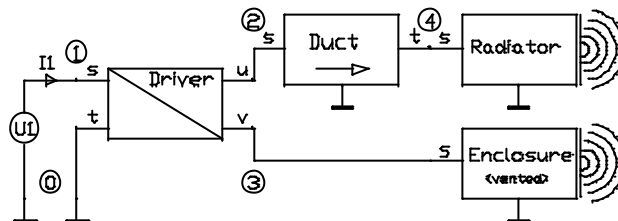


Fig. 90 Circuit diagram for Example 2

```
...
System 'S1'
Driver Def='Drv1' Node=1=0=2=4
Duct 'T1' Node=2=3
dD=17cm Len=50cm
Radiator Def='T1' Node=3
x=0 y=0 z=0 HAngle=0 VAngle=0
dEdge=17cm |Edge diffraction
Enclosure 'E1' Node=4
Vb=40L Qb/fo=0.1 Lb=20cm
```

3. Simulation of a directly radiating cone loudspeaker without baffle and without enclosure, i.e. the loudspeaker radiates as a dipole.

The `Radiator` with the name 'Front' is connected to the diaphragm front of the driver (node `v=2`). The radiator takes the data for the radiation area from the `Driver` element with the name 'D1'.

The `Radiator` with the name 'Back' is connected to the reverse side of the driver (node `u=3`). The perpendicular to the diaphragm surface is rotated backwards by  $H\text{Angle}=180^\circ$ . The acoustic center is displaced 14cm backwards (measured).

A conspicuous feature is the entry for `dEdge=1mm` in this example. If the diameter of the baffle is smaller than that of the radiation surface, `AkAbak` sets `dEdge=dD`, to calculate the sound diffraction. `dEdge=1mm` forces the change over, since the driver diaphragm forms its own baffle here.

Note: In almost all conical loudspeakers, the diaphragm area is somewhat smaller on the reverse side, since the drive occupies some of the area. Usually the difference can be neglected.



```

...
System 'S1'
  Driver 'D1'      Def='Drv1'  Node=1=0=2=3
  Radiator 'Front'  Def='D1'   Node=2
    x=0 y=0 z=0 HAngle=0 VAngle=0
  .. dEdge=1mm
  Radiator 'Reverse' Def='D1'   Node=3
    x=0 y=0
    z=-14cm          |Path of the sound around the driver
    HAngle=180°      |diaphragm reverse radiates backwards
    VAngle=0
  ..dEdge=1mm

```

## Parameter

Radiator	Keyword
'...'	Identifier
Def='...' (Optional)	Reference to an element or a definition, the details of whose radiation surfaces are to be read in. (Driver, Duct, Diaphragm, Waveguide, Coupler, Def_Driver, Def_TwoCoilsDriver Or Def_PiezoDriver). For reference elements that in turn refer to a definition, for example Driver, the entry Def= can refer to the name of the network element or to the name of the definition. Any explicit reference to the radiation surface is disabled by a reference entry. Exactly the same character sequence has to be present between the quotations marks ("..." or '...') as in the name of the reference element; naturally, in this case the reference element has to have a name. No distinction is made between upper and lower case letters.
Node=s	The volume velocity flows through the radiation impedance from node <b>s</b> to ground.

## Diaphragm

See also chapter "Introduction/Diaphragm form parameter" (These entries are ignored if Def= is specified).

dD=...m	Diameter.
WD=...m	
HD=...m (Alternative)	Width or height.
SD=...m <sup>2</sup> (Alternative)	Area

### Only loudspeaker diaphragms

dD1=...m (Optional)	Diameter of inner diaphragm.
SD1=...m <sup>2</sup> (Alternative)	Area of inner diaphragm.
tD1=...m (Optional)	Depth or height of diaphragm form.
t1=...m (Optional)	Displacement of diaphragm within baffle.
f <sub>p</sub> =...Hz (Optional)	Mass and area reduction control frequency.
Diffuse=...% (optional)	Directivity control factor

## Radiation

See also chapter "Introduction/Position of radiators, /Radiation environment":

<code>x=...m</code>		Horizontal position.
<code>y=...m</code>		Vertical position.
<code>z=...m</code>		Axial position.
<code>HAngle=...°</code>		Horizontal position angle (-180°...180°)
<code>VAngle=...°</code>		Vertical position angle (-90°...90°)
<code>Reflection</code>	(Optional)	Takes part at evaluation of reflections.
<code>WEdge=...m</code>	(Optional)	Diffraction. Width of baffle.
<code>HEdge=...m</code>	(Optional)	Diffraction. Height of baffle.
<code>dEdge=...m</code>	(Optional)	Alternative. Diameter of baffle.
<code>NoRad</code>	(Optional)	No radiation takes place.
<code>NoDir</code>	(Optional)	Radiates as a point source.

### Only loudspeaker diaphragms

<code>t1=...m</code>	(Optional)	Displacement of diaphragm within baffle. <code>t1</code> can only be applied on loudspeaker diaphragms. This parameter does not work together with the apertures of <code>Duct</code> - or <code>Waveguide</code> -elements. The parameter <code>t1</code> can be entered twice: 1. together with diaphragm form parameter 2. together with mounting position parameter. The program adds the given values internally.
----------------------	------------	--

See also: `BassUnit`, `Speaker`, `Enclosure`, `Horn`

## Enclosure

*Dialog: Net/ Acoustic/ Enclosure*

The `Enclosure` element is used to install closed or vented cabinets in the network.

On the one hand `Enclosure` is used to implement standard cabinets which are met most frequently in practice. With just a few parameter you are able to describe the most important features of your cabinet.

On the other hand `Enclosure` can be used everywhere where you encounter a cavity. Together with the elements `Duct`, `Waveguide` and `Impedance` you are able to describe more complicate structures.

The enclosure is usually rectangular. The entry for the enclosure depth is used to calculate the standing waves of the basic mode in at least one dimension. (Fig. 92).

For non-rectangular enclosures, (Fig. 95) it is not possible to calculate the mode of vibration. In this case the enclosure is regarded as a linear acoustic spring force. The imaginary component of the radiation impedance (air load) of the enclosure is, in the non-rectangular case, provided by the entry for the acoustic mass over the cross-sectional area  $S_b$ .



Fig. 91 Circuit diagram of the `Enclosure` element: left closed, right vented cabinet



Fig. 92 Rectangular sealed enclosure

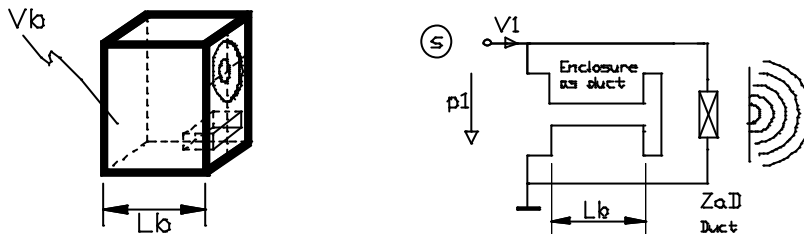


Fig. 93 Rectangular 'deep' reflex-enclosure, the reflecting wall lying opposite the source and vent.

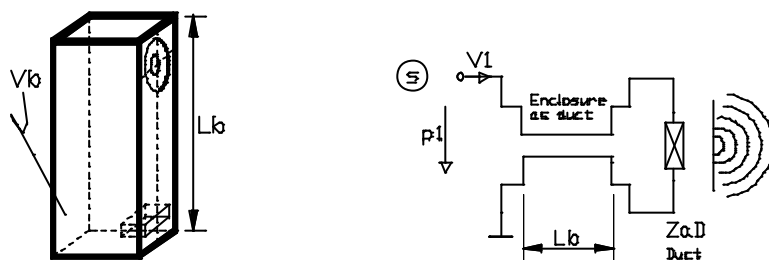


Fig. 94 Rectangular 'long' reflex enclosure, the reflecting wall and vent lying opposite the source

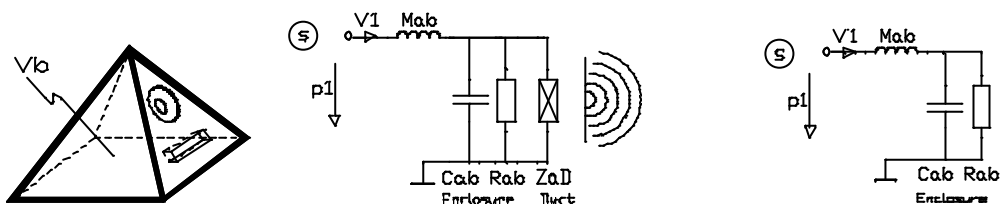


Fig. 95 Non-rectangular vented and sealed enclosure

### ☑ Examples

- 1.: Rectangular closed cabinet with a volume of  $V_b=60\text{L}$  and a depth (interior) of  $L_b=25\text{cm}$ . The enclosure loss, at  $Q_b/f_o=0.1$ , is typical for conventional enclosure shapes.

Enclosure 'E1' Node=4  
 $V_b=60\text{L}$   $Q_b/f_o=0.1$   $L_b=25\text{cm}$

- 2.: As in the first example. However, the enclosure is not rectangular (Fig. 95). The enclosure presents to the connected sound sources an air load corresponding to an area of  $S_b=2400\text{cm}^2$ .

Enclosure 'E1' Node=4  
 $V_b=60\text{L}$   $Q_b/f_o=0.1$   $S_b=2400\text{cm}^2$

3.: Rectangular reflex enclosure, as in Example 1 and Fig. 93. The Helmholtz resonance is  $f_b=30\text{Hz}$  and the vent diameter is  $d_D=5\text{cm}$ . The vent outlet is displaced 20 cm downwards on the baffle ( $y=-20\text{cm}$ ). Infinite baffle conditions are assumed. Tunnel-viscosity is taken into account by default.

```
Enclosure 'E1' Node=4
Vb=60L Qb/fo=0.1 Lb=25cm
fb=30Hz dD=5cm
x=0 y=-20cm z=0 HAngle=0 VAngle=0
```

4.: The vent outlet is on the reverse side of the enclosure ( $z=-55\text{cm}$ ,  $\text{HAngle}=180^\circ$ ) and is displaced downwards by 20cm ( $y=-20\text{cm}$ ). The dimensions of the baffle are  $\text{WEdge}=35\text{cm}$ ,  $\text{HEdge}=55\text{cm}$ . The vent opening participates in the reflection if `Def_Reflector` has been entered and not switched off. The vent is rectangular in this case with a width of  $\text{WD}=2.5\text{cm}$  and a height of  $\text{HD}=2\text{cm}$ .

The 'Helmholtz' dialog (Def/Helmholtz menu) is used to calculate a vent length of approx. 6.6cm. If the wall thickness of the box is additionally deducted, then the vent inlet is located in the interior of the cabinet, more to the reverse side of the box, i.e. opposite the sound source. That is why the keyword `Long` is entered (Fig. 94).

Instead of tunnel-damping by viscosity an absorption-quality factor is specified.

```
Enclosure 'E1' Node=4
Vb=60L Qb/fo=0.1 Lb=50cm Long
fb=30Hz WD=2.5cm HD=2cm Visc=0 QD/fo=0.1
x=0 y=-20cm z=-55cm HAngle=-180° VAngle=0
WEdge=35cm HEdge=55cm Reflection
```

## Parameter

Enclosure	Keyword
'...'	Identifier
Nodes	Connection pole of the enclosure. The other pole is always ground, i.e. the enclosure wall.
Vb=...m <sup>3</sup>	Effective volume of the enclosure, i.e. the cubic volume minus the volume occupied by the driver and other equipment in the enclosure. The volume is also increased if insulating material is present (see <code>AcouCompliance</code> element). Unit cubic meters [m <sup>3</sup> ], cubic inch [in <sup>3</sup> ] or liter [Liter, L].
Qb/fo=... (Optional)	Enclosure quality divided by the frequency at which Qb has been measured, i.e. usually the Helmholtz resonance $f_b$ . Its value is $Q_b/f_o=0.1$ if, for example, at a Helmholtz resonance of $f_b=50\text{Hz}$ , a quality of $Q_b=5$ has been measured. $Q_b/f_o$ can be entered for sealed enclosures and reflex enclosures. (see also note 'enclosure losses'). If no value is specified, <code>AkAbak</code> sets $Q_b/f_o=1000$ which stands for the lossless case.
Lb=...m (Optional)	Rectangular enclosure. When <code>Lb=</code> is entered, the enclosure is rectangular. The program calculates the standing waves in one dimension (Fig. 91 to Fig. 94). If nothing else is entered, the standing waves between the wall on which the driver (or sound source) is mounted and the opposite wall of the enclosure are calculated (Fig. 94, Fig. 93). The radiation impedance providing the standing wave is also responsible for the air load (imaginary component) that,

for example, a driver experiences when it is connected to the housing.  
 In the case of reflex housings, there are two modes: A 'deep' (Fig. 93) and a 'long' (Fig. 94) enclosure. See the description of the keyword `Long` below.  
`Lb=` is thus the distance between the wall with the sound source and the opposite wall of the enclosure.  
 Unit meter [m] or inch [in].

`SB=...`m<sup>2</sup> (Optional) For the approximate determination of the air load `Mab` (Fig. 95) that a non-rectangular enclosure presents to the connected sound sources:

$$Mab = \frac{8}{3} \cdot \frac{\rho}{\pi^2} \cdot \sqrt{\frac{\pi}{SB}}$$

where  $\rho$ : air density

`Mab` is equivalent to the air load on one-side of a diaphragm in a baffle. It is very difficult to determine `Mab` accurately in non-rectangular enclosures, especially if insulating material is present.

Usually `SB` is the area of the baffle in the interior of the box. If the enclosure is, for example, pyramidal, then the area of one side of the pyramid should be given for `SB`. If insulating material is present, this area is reduced by, for example, a quarter to take into account the increase in mass due to the insulating material (`Mab` increases as the area decreases. At first sight, that appears paradoxical, but acoustic parameter always relate to an area).

Unit in square meter [m<sup>2</sup>] or in square-inch [in<sup>2</sup>]

### Following parameter are only valid for vented cabinets

`Len=...`m Length of vent. When `Len` is entered, `Enclosure` describes a vented cabinet, otherwise a closed cabinet.  
 Alternative to the Helmholtz resonance `fb=`.  
 For vented cabinets the vent cross-section area has to be specified.

`fb=...`Hz Reflex enclosure, Helmholtz resonance.  
 When `fb` is entered, `Enclosure` describes a vented cabinet, otherwise a closed cabinet:

$$fb = 1 / \sqrt{2\pi \cdot MaD \cdot Cab}$$

`MaD`: acoustic mass in the vent [kg/m<sup>4</sup>]

`Cab`: acoustic compliance of the enclosure volume [m<sup>5</sup>/N]

For vented cabinets the vent cross-section area has to be specified.

The program calculates the vent length from `fb=`.

Unit hertz [Hz]

`QD/fo=...` (Optional) Reflex enclosure. Vent quality divided by the Helmholtz quality `fb`. Frequency independent damping along the transmission of the vent:  
 Range: 0.00001 ... 1000, Typ.: 0.1 ... 2  
 (see note 'losses in the reflex housing' below).

`Visc=...` (Optional) Reflex enclosure. Factor to switch off calculation of vent-damping due to viscosity or to modify its strength.  
 Off: `Visc` = 0  
 Default: `Visc` = 1  
 Decrease of damping: `Visc` = 0 ... 1  
 Increase of damping: `Visc` > 1

`Long` (Optional) Reflex enclosure. Keyword for a so-called 'long' reflex enclosure (Fig. 94).  
 If this is entered, the enclosure is similar to an acoustic duct and the vent inlet in the enclosure interior is opposite the sound source.  
 If this is not entered, it is assumed that the vent inlet and sound source have an opposite wall in common (Fig. 94).

The configuration usually found in practice exactly corresponds to neither of these cases. It is unfortunately difficult to simulate the acoustic conditions in the interior of an acoustic body. If this has to be done, however, it requires considerably more geometrical data than are necessary here. Since, however, this has a relatively small effect on the degree of accuracy sought here, a more detailed simulation has been dispensed with. Therefore, we estimate from case to case into which category the particular design falls and enter the `Long` keyword as necessary.

<code>dD=...</code>		Mandatory for reflex enclosures. Diameter of the vent. If <code>fb=</code> or <code>Len=</code> is not given, then <code>dD=</code> is ignored. Unit meter [m] or inch [in].
<code>WD=...</code>	(Alternative)	Reflex enclosure. width ( <code>WD=</code> ) and height ( <code>HD=</code> ) of a rectangular vent area. <code>WD=</code> and <code>HD=</code> have to be entered together Alternative to the diameter of the vent <code>dD=</code> . Unit meter [m] or inch [in].
<code>HD=...</code>		
<code>SD=...</code>	(Alternative)	Reflex enclosure. Vent area. alternative to the diameter <code>dD=</code> or rectangle <code>WD=</code> and <code>HD=</code> . Unit square meter [ $\text{m}^2$ ] or square inch [ $\text{in}^2$ ].

### See also chapter "Introduction - Position of radiators - Radiation environment"

The parameter of position are only valid for vented enclosures and are ignored otherwise. The reference point of mounting is the center of the vent outlet.

<code>x=...</code>		Horizontal position.
<code>y=...</code>		Vertical position.
<code>z=...</code>		Axial position.
<code>HAngle=...</code>		Horizontal position angle (-180°...180°)
<code>VAngle=...</code>		Vertical position angle (-90°...90°)
<code>Reflection</code>	(Optional)	Takes part at evaluation of reflections.
<code>WEdge=...</code>	(Optional)	Diffraction. Width of baffle.
<code>HEdge=...</code>	(Optional)	Diffraction. Height of baffle.
<code>dEdge=...</code>	(Optional)	Alternative. Diameter of baffle.
<code>NoRad</code>	(Optional)	No radiation takes place from the vent.
<code>NoDir</code>	(Optional)	Vent radiates as a point source.

## Enclosure losses

In the simple equivalent circuit diagram of the model of `Enclosure`, the acoustic resistance `Rab` represents losses in the enclosure (Fig. 95).

The quality as a function of frequency  $Q_b/f_0$  has the following relationship with `Rab`:

$$Q_b/f_0 = 2\pi \cdot C_{ab} \cdot R_{ab}.$$

$f_0$  is the Helmholtz frequency at which  $Q_b$  has been measured. If you want to measure  $Q_b/f_0$  for a closed enclosure, install a vent temporarily to produce a reflex enclosure. Measure the Helmholtz frequency  $f_b$  and the quality  $Q_b$  of this reflex enclosure.  $Q_b/f_0$  is then  $Q_b/f_b$ .

`Rab` primarily represents losses due to leakages in the enclosure. A further loss resistance, representing absorption, is in series with the compliance of the enclosure `Cab` (not shown in the illustrations). Losses due to

absorption of the sound energy in the enclosure wall occur in the bass range as a result of the wall vibrating in sympathy, and at higher frequencies by cavities. Since, if the loudspeakers are relatively sturdy in construction, this resistance is negligible, in the model it is comprised in  $R_{ab}$ .

If air flow in the reflex vent is not hindered, the additional resistance of the vent is usually also negligible here, and is reflected in  $R_{ab}$ .  $R_{ab}$  thus not only represents leakages, but also other losses in the enclosure and the vent, if they are not predominant.

If no  $Q_b/f_o=$  is specified the program sets  $Q_b/f_o=1000$  internally. By this the lossless case is described.

## Vented cabinet

**Enclosure** becomes a reflex enclosure if the tunnel-length  $L_{en}=$  or the Helmholtz frequency  $f_b=$  is specified. A duct and a radiation element for the vent is installed automatically. Enter also the vent cross section diameter  $dD=$  or area  $SD=$ .

## Radiation

The radiation element has to be positioned on the baffle with the normal position parameter ( $x=$ ,  $y=$ , ...). The parameter for sound diffraction ( $W_{Edge}=$ ,  $H_{Edge}=$ ) and reflection ( $Reflection$ ) also can to be specified. Diffraction and reflection both influence the radiation resistance of the port aperture.

## Helmholtz resonance

If you specify the Helmholtz resonance  $f_b=$  the program will evaluate the vent length  $L_{en}=$ . In this case the length is evaluated with the Helmholtz-formula with a one-sided end correction of a duct mounted in an infinite baffle. The other side is loaded by the enclosure. Usually there is a good approximation of the inner loading due to the fact that standing waves are calculated ( $L_b=$ ).

The 'Helmholtz' dialog assists in the design of the reflex enclosure. Since the calculation does not take many boundary conditions into account, the results should be viewed as just guide values (menu "Def/Helmholtz", input field "fb...", see chapter "Introduction/Enclosure").

## Multiple vents

The program starts out from a single vent, even though it is by all means possible to connect several vents in parallel. The acoustic mass in the vent is reduced as the vent cross-sectional area becomes larger, and the Helmholtz resonance  $f_b$  therefore increases. The losses in the vent increase with the number of vents, since the viscosity increases and more turbulence is produced. If the effect of several vents is to be investigated in detail, the reflex enclosure has to be built up from discrete elements ( $Duct$ , ...).

## Measuring vented cabinets

There are some very practical and easy to measure parameter to control a reflex enclosure. These standard formulae are derived from the simple model which does not include standing waves and the influence of the voice coil.

Before you mount the driver into the reflex enclosure determine the motor parameter of the free-radiating driver:  $f_s$ ,  $Q_{ms}$ ,  $Q_{es}$ ,  $R_e$  and  $V_{as}$  (Tools/Dyn. driver parameter).

When you measure the electrical impedance curve of the mounted loudspeaker you will see the typical curve with two amplitude peaks and four zeros in the phase curve (Fig. 96).

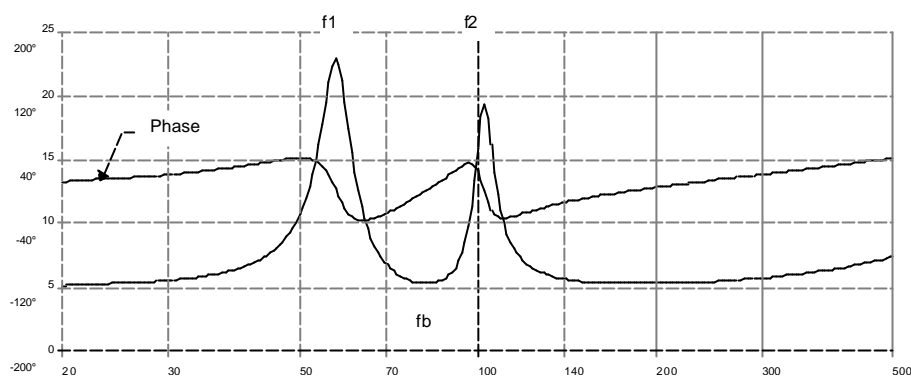


Fig. 96 Magnitude and phase of electrical impedance of a driver mounted in a vented enclosure

From this curve we need the frequencies  $f_1$  and  $f_2$  where the amplitude is maximum and the phase is zero ( $f_1 = 57.5$  Hz,  $f_2 = 102.5$  Hz). Further we measure the Helmholtz-frequency  $f_b$  at amplitude minimum and also zero phase but with positive gradient. Here we measure also the magnitude of the amplitude  $Z_b$  (in the diagram  $Z_b = 5.36$  ohm @ 81 Hz).

With these values some interesting parameter can be calculated:

1. Mass-load factor  $m_b$  = due to mounting the driver into the enclosure.

$$m_b = f_1 \cdot f_2 / (f_s \cdot f_b)$$

2. Relation between  $V_{as}$  and  $V_b$

$$V_{as}/V_b = (f_2^2 - f_b^2) \cdot (f_b^2 - f_1^2) / (f_1^2 \cdot f_2^2)$$

3. Enclosure quality factor  $Q_b/f_0$

$$\frac{Q_b}{f_0} = \frac{V_b}{f_{sb} \cdot V_{as}} \cdot \left[ \frac{1}{\sqrt{Q_{esb}^2 \cdot \left( \frac{Z_b}{R_e} - 1 \right)^2 - \left( \frac{f_{sb}}{f_b} - \frac{f_b}{f_{sb}} \right)^2}} - \frac{1}{Q_{msb}} \right]$$

with

$$f_{sb} = m_b \cdot f_s$$

$$Q_{msb} = Q_{ms}/m_b$$

$$Q_{esb} = Q_{es}/m_b$$

$$f_0 = f_b$$

4. There is a practical relationship between the vented and closed enclosure:

$$f_b^2 = f_1^2 + f_2^2 - f_c^2$$

with  $f_c$ : resonance frequency of the driver in a closed enclosure.

## Losses in the vent

### Constant damping

Apart from the radiation resistance and the loss due to viscosity, you can also enter your own loss - frequency independent - value for the vent of the reflex enclosure.

The loss of the vent is expressed by the vent quality  $Q_D$  divided by the Helmholtz-frequency  $f_b$ . If you specify this quality factor, the transmission of the reflex-duct includes absorption:



$$\frac{p_2}{p_1} = e^{-\alpha \cdot (z_2 - z_1)}$$

where

$p_2, p_1$ : pressure at distance  $z_2$  and  $z_1$

$$\alpha = \frac{2\pi}{c \cdot (QD / f_0)}, \quad c: \text{velocity of sound}$$

$Q_b$  is relatively easy to measure (see above),  $QD$  generally not.  $QD$  was only introduced here to demonstrate the effect of an obstruction in the vent.

For example, you can simulate your system using values of, e.g.  $QD/f_0 = 0.05 \dots 2$ , and ascertain in which manner, and especially with which sensitivity, the reproduction curve varies in the vicinity of  $f_b$ .

Small provides valuable observations on this topic [Sma2].

### Viscosity effect in the vent

In addition to the radiation resistance and  $QD/f_0$ , a further loss resistance is also calculated in the vent, namely that due to the viscosity of air. The program reads the parameter required for this from the entries for the vent area and for the length. As was discussed for the `AcouResistance` and `Duct` elements, the losses in the vent increase if velocity differences occur along the vent. The losses increase, for example, in the case of a small cross-section if the cross-sectional area is slit-shaped, if there are bends or kinks in the vent, if, instead of one vent, there are several smaller ones, or if the vent cross-section changes suddenly, or meshes or the like are installed.

For a more detailed explanation of the factor `Visc=` and the viscosity formula used in the program see chapter `AcouResistance`.

### Note

In order to merge the model and analysis so that a simple, closed mathematical solution is possible, as has been demonstrated by Small [Sma] and many others, note the following: In the closed analysis (i.e. via a transfer function) of the closed enclosure, it is advisable to regard the loss resistance  $R_{ab}$  as in series with the compliance of the enclosure  $C_{ab}$ . The resulting error is generally small. This fine distinction can be clearly seen, however, in the analysis of the diaphragm excursion. (compare the diaphragm excursion of a closed enclosure shown by the diagram option in the `Def_Bassunit` Dialog with the equivalent circuit diagram illustrated here.) This can also be regarded analytically: The diaphragm excursion of the closed enclosure is proportional for very low frequencies:

$$X(f \rightarrow 0) \equiv C_{ms} \quad R_{ab} \text{ is parallel to } C_{ab} \text{ and } R_{ab} \ll 1/(w \cdot C_{ab}).$$

$$X(f \rightarrow 0) \equiv C_{mt} \quad R_{ab} \text{ is in series with } C_{ab} \text{ and } C_{ms}$$

where  $C_{ms}$  is the mechanical compliance of the driver diaphragm and  $C_{mt}$  is the series circuit of  $C_{ms}$  with  $C_{ab}$ .

In the parallel circuit used here (script analysis), the diaphragm excursion at very low frequencies is thus independent of the enclosure shape, if serious losses, i.e. leakages, are present, which are indeed represented by  $R_{ab}$ .

See also:

`AcouCompliance`, `Duct`, `BassUnit`

# Duct

*Dialog: Net/ Acoustic/ Duct*

**Duct** creates an acoustic vent in the network. **Duct** is a waveguide (or transmission line) - a channel with a uniform cross-sectional area.

Most of what has been said about the **AcouMass**, **AcouCompliance** and **AcouResistance** elements also applies to the **Duct** element. The difference is that not an approximation of the tangent or cotangent function is used here, but the function itself.

**Duct** is usually four-poled. The program calculates the transmission behavior from the input port to the output port. Depending on the acoustic termination of the two ports and on the wavelength of the sound, reflections occur in the longitudinal direction of the duct. The higher vibration modes are not taken into account. The dimensions of the cross-section are small compared with the wavelength to be transmitted. The diameter of the duct should be in the range of<sup>4</sup> :

$$0.1/\sqrt{f} \leq dD \leq 20/f$$

where

$f$  frequency [Hz]  
 $dD$  duct diameter [m]

Since one of the poles of the ports always has to be grounded, **Duct** is always given with two poles here. The other poles are grounded internally.

If one of the two other poles is grounded (=0), **Duct** acts like a closed (covered) duct or like the **Enclosure** element if the cabinet is closed and rectangular (Fig. 99), otherwise **Duct** represents an acoustic mass at very low frequencies (Fig. 98).

Losses in ducts are caused by viscosity, which - depending on the design - is automatically taken into account in the calculation. It is possible to switch this calculation off or modify its strength. Further damping can be specified.

A pressure chamber can also be installed at the entrance to the **Duct**, as is produced for example in the case of horn drivers or acoustic low passes (Caf in Fig. 98 and Fig. 99).

The **Duct** element can be used to analyze very many acoustic problems more accurately, or to simulate exotic designs. If, for example, the **Enclosure** element does not represent a reflex enclosure accurately enough, then it can be built up from a "closed"-**Enclosure**, **Duct** and **Radiator** elements. The procedure depends on the geometry of the system to be simulated and on the ratio between the individual components and the wavelength. Further examples can be found, for example, horn designs, sound dampers, in particular bass loudspeakers, etc.

## ☑ Examples

1.: **Duct** with a diameter of 17cm and a length of 10cm. The name is 'D1'. The input is at node **s=2** and the output at node **t=3**. At the duct output is a **Radiator** element, which retrieves the data for the radiating surface ( $dD=17\text{cm}$ ) from the **Duct** element using  $\text{Def}='D1'$ .

```
Duct   'D1'   Node=2=3
      dD=17cm Len=10cm
```

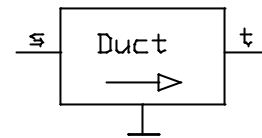


Fig. 97 Circuit diagram of the **Duct** element

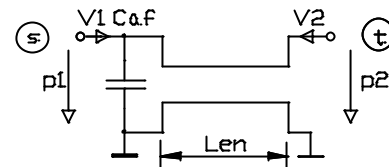


Fig. 98 Equivalent circuit diagram of an acoustic duct

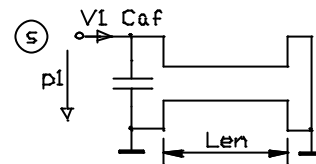


Fig. 99 Equivalent circuit diagram of an acoustic duct closed at one end

<sup>4</sup> see footnote **AcouCompliance**

```
Radiator Node=3=0 Def='D1'
x=0 y=0 z=0 HAngle=0 VAngle=0
```

2.: As Example 1, except that here a pressure chamber ( $V_f=0.3L$ ) is installed. It is assumed that a `Driver` element comes in front, whose diaphragm diameter  $d_D=17\text{cm}$  is.

```
Duct 'D1' Node=2=3
dD=12cm Len=10cm Vf=0.3L
Radiator Node=3=0 Def='D1'
x=0 y=0 z=0 HAngle=0 VAngle=0
```

3.: As the above examples but with further damping specified ( $QD/f_o=0.1$ ).

```
Duct 'D1' Node=2=3
dD=12cm Len=10cm Vf=0.3L QD/fo=0.1
```

4.: As the above examples but with switched off calculation of viscosity.

```
Duct 'D1' Node=2=3
dD=12cm Len=10cm Vf=0.3L Visc=0
```

## Parameter

Duct		Keyword
'...'		Identifier. Required if the <code>Radiator</code> element with the identifier <code>Def='...'</code> relates to <code>Duct</code> .
Node=s=t		<code>Duct</code> is usually four-poled, however for convenience you have to enter only two poles. The sound enters the duct via pole <b>s</b> and leaves it via pole <b>t</b> . If no pressure chamber is installed ( $V_f=0$ ), <code>Duct</code> is symmetrical, otherwise the pressure chamber is on the side of pole <b>s</b> to ground. If one of the two poles <b>s</b> or <b>t</b> is grounded ( $=0$ ), then <code>Duct</code> describes a closed duct (closed pipe).
Len=...m		Length of the duct. Unit meter [m] or inch [in].
$V_f=...m^3$	(Optional)	Size of the pressure chamber volume at the inlet to the duct (from pole <b>s</b> to ground). A pressure chamber is always produced when the cross-sectional area is reduced. For example when the diaphragm of a driver radiates into a duct with smaller cross-sectional area. As in the case of the horn driver or the acoustic low pass. It is important that the diameters of the pressure chamber are much smaller than the wavelength. Unit cubic meter [ $m^3$ ], cubic inch [ $in^3$ ] or liter [Liter, L].
$QD/f_o=...$	(Optional)	Frequency independent damping along the transmission of the duct. Range: 0.00001 ... 1000, Typ.: 0.1 ... 2
$Visc=...$	(Optional)	Factor to switch off calculation of damping due to viscosity or to modify its strength. Off: $Visc = 0$ Default: $Visc = 1$ Decrease of damping: $Visc = 0 \dots 1$ Increase of damping: $Visc > 1$

**Parameter of cross section see also chapter "Introduction-Diaphragm form parameter":**

$dD = \dots m$  Diameter of the duct.

$WD = \dots m$

$HD = \dots m$  (alternative) Width or height of the duct if the cross-sectional area is rectangular.

$SD = \dots m^2$  (alternative) Area of the duct with circular cross section.

**Several ducts connected in parallel and in series**

**Duct** elements can be connected in parallel or in series. For example, the transmission behavior of the phase correction element of a horn driver can be studied using ducts connected in parallel. At the entrance to the ducts, the same sound pressure prevails everywhere. The cross-sectional area and length of the individual ducts vary (e.g. the outer ducts become somewhat longer than the inner ones).

Bends in sound channels (e.g. in the case of the Y-fitting, when two drivers radiate into one horn) can also be represented using horns connected in parallel. The channel itself is subdivided into several smaller channels which together have the same cross-sectional area. The bend is then simulated by means of appropriate length parameter of the **Duct** elements.

Note here, however, that the program automatically includes the losses due to viscosity of air in the duct. The narrower the ducts, the greater the damping along the duct. Since, however, any change in the sound conduction also entails similar losses, this effect is usually welcome. If you do not want absorption insert  $Visc=0$  which prevent **AkAbak** to take into account viscosity losses.

It is also possible to connect ducts in series, and this does not pose any problems if the cross-sectional areas of the individual **Ducts**, or of the sum of several ducts connected in parallel, are identical or only slightly different. If that is not the case, we must reach further into our bag of acoustic tricks to carry out a very accurate analysis. Here, at the transitions, there are additional acoustic masses and compliances that depend, among other things, on the order of magnitude and shape of the transition. The program does not install such elements automatically. See the description of the **AcouMass** and **AcouResistance** elements.

**Constant damping QD/fo**

A specification of the quality factor  $QD/fo$  induces frequency independent damping along the transmission of the duct:

$$\frac{p_2}{p_1} = e^{-\alpha \cdot (z_2 - z_1)}$$

$p_1$

where

$p_2, p_1$ : pressure at distance  $z_2$  and  $z_1$

$$\alpha = \frac{2\pi}{c \cdot (QD / fo)}, \quad c: \text{velocity of sound}$$

**Viscosity**

From the input data the program calculates the losses due to the viscosity of air. The viscosity of the gas always becomes significant at narrow points by increasing the damping along the vent. Compare, for example, the simulations of **Duct** elements with the same cross-sectional area. In one case let the cross-section be square or circular and in the other case let the height be, e.g.,  $HD=1cm$  and the width be  $WD=30cm$ . The damping in the second case is much greater. That means that the program not only takes into account the area of the cross-section, but also its shape.

Taking the formula given by Morse [Mor] which is given in chapter **AcouResistance** and setting the boundary layer of viscosity and thermal exchange to equal values the acoustical frequency dependent resistance is approximately:

$$Ra(\omega) = \text{Visc} \cdot \frac{D \cdot L_{\text{en}}}{SD^2} \cdot \sqrt{\eta \cdot \rho \cdot \omega}$$

with

$$Ra = \frac{\rho \cdot c}{SD} \cdot \alpha \cdot L_{\text{en}}$$

$$\alpha = \text{Visc} \cdot \frac{U}{c \cdot SD} \cdot \sqrt{\frac{\eta \cdot \omega}{\rho}}$$

with

$\rho$ :	Density [kg/m <sup>3</sup> ]
$D$ :	Perimeter of cross-section [m]
$\eta$ :	Dyn. viscosity [Pa·s] ( $\eta=18.2 \cdot 10^{-6}$ Pa·s, air @ 20°)
$\alpha$ :	Damping factor (see last paragraph)
Visc:	Control parameter to modify the effect

## End correction

The entry for the vent length,  $L_{\text{en}}$ , has to include any end corrections that have to be taken into account. End correction, in this context, means taking into account the air load at the ends of the vent. Air load is an acoustic mass and is caused by radiation impedance when the vent radiates into a relatively large room, for example if it terminates in an enclosure or continues into a vent with larger cross-section (see above).

If a `Radiator` element is mounted at the vent outlet, no end correction must be applied for this component, since `Radiator`, after all, installs a corresponding radiation impedance.

Since, for low frequencies, the vent represents a mass that depends on the length, the end correction is taken into account simply by specifying a somewhat greater length. The magnitude of this additional length  $L_+$  depends on the radiation conditions and the ratio of the wavelength to the dimensions:

$$L_+ = 0.425 \cdot dD \quad \text{One end. Vent terminates in a baffle.}$$

$$L_+ = 0.306 \cdot dD \quad \text{One end. Vent terminates freely.}$$

$L_+$  is added to the geometrical length of the vent and produces the length  $L_{\text{en}}$ . If an end correction is to be applied to both ends, the corresponding  $L_+$  has to be added for both ends. As stated: If the cross-sectional area only changes insignificantly, or if a `Radiator` element is connected,  $L_+$  is not applied for that end.

The effect of the air load is negligible in many cases. To estimate it means raising the question again of how sensitively the output parameter (e.g. the sound pressure level) reacts to a change of a parameter.

## Radiation and Referencing

If a vent is to radiate sound, a `Radiator` element has to be installed at the output. This `Radiator` element must have the same radiation area as the `Duct`. To facilitate this operation, `Radiator` can refer to `Duct` via the identifier `Def='...'`. In this case, `Duct` has to be provided with a name and must come before `Radiator` in the script.

The radiation from duct aperture into free space is identically to that of a flat piston like diaphragm.

## Recessing in the baffle

In practice, the diaphragm is often recessed in the baffle. Such recessing has a significant effect on the transmission and should not be neglected in the simulation. If such a structure is described with the elements `Driver-Duct-Radiator`, the simulation lacks the calculation of the radiation form, since the radiation from of the `Duct` element provides only for a flat diaphragm.

### Small recessing

For diaphragms with a mounted vent that has the same diameter as the diaphragm and is not longer than the diaphragm radius, the diaphragm shape parameter  $t_{1=}$  should be used. A `Duct` element does not need to be installed then and the radiation is carried out via the `Radiator` element.

### Deep recessing

If, however, the vent is longer than the diaphragm radius, if other acoustic structures follow, or if the vent cross-section is smaller than the diaphragm area, then the appropriate element is `Duct`, which has to be connected in front of the diaphragm.

### Diaphragm cones as Duct

If a diaphragm cone radiates via a pressure chamber into an acoustic duct or a horn, the diaphragm cone should be modeled by a `Duct` element (Fig. 100).

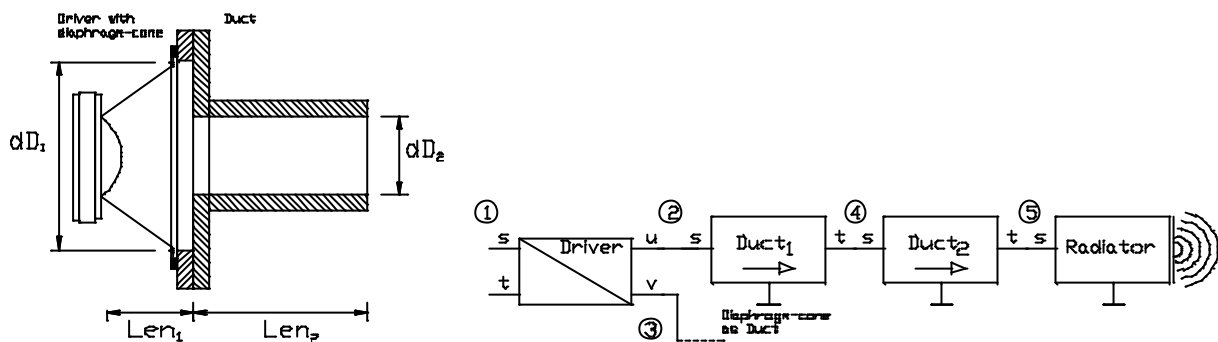


Fig. 100 Diaphragm cone radiating across a pressure chamber in an acoustic duct

### ☑ Example

```

Def_Driver "D1"
  dD=30cm  dD1=14cm  tD1=5cm  fp=800Hz
  fs=40Hz  Mms=80g
  Qms=3    Qes=0.4  Re=5ohm  Le=5mH

System 'S1'
  Driver Def="D1"  Node=1=0=2=3
  Duct "Du1"
    Node=2=4  Len=7cm  dD=22cm  | cone + mounting baffle
                                | mean value (dD+dD1)/2
  Duct "Du2"
    Node=4=5  Len=30cm  dD=12cm  | waveguide
    Vf={pi*0.12*(0.3-0.12)/2*0.03} | transition compliance
  Radiator Def="Du2"
    Node=5      x=0  y=0  z=0    | radiation from duct 2

```

The script describes a driver with a diaphragm cone of  $dD=30\text{cm}$  diameter and a depth of  $tD1=5\text{cm}$ . The thickness of the mounting board is  $2\text{cm}$ . Duct 1 models the diaphragm cone with mounting board. The length of this duct is therefore  $Len=7\text{cm}$ . The diameter of 'Du1' is equal to the average of the diaphragm diameter  $dD=30\text{cm}$  of the driver and its inner diaphragm  $dD1=14\text{cm}$ .

Duct "Du2" represents the waveguide with a diameter of  $dD=12\text{cm}$ . The radiation element `Radiator` takes the information for the cross-sectional area from Duct "Du2".

The Duct "Du2" is also preceded by a compliance  $V_f$ . The volume  $V_f$  is used to take into account a region of the pressure chamber in which the air particles are not displaced but compressed. Imagine water flowing through a channel with the above shape. The water is pent up on both sides upstream of the Duct "Du2". It

behaves in a similar manner in the acoustic pressure chamber. The volume of this "calm water" has to be estimated.

The inline formula describes an annular volume.

See also: Radiator, Enclosure, Waveguide, Horn, Coupler

## Horn

*Dialog: Net/ Acoustic/ Horn*

**Horn** is a compact network element with in-built radiation. A horn is a waveguide with a specific length, a throat surface and mouth surface. The cross-sectional area of the horn increases constantly from the horn throat to its mouth. At the horn throat is mounted the sound source. This source may be a *Driver*, *Duct*, *Waveguide*, or an other acoustic element. It is loaded with the acoustic input impedance of the horn. The sound is radiated at the horn mouth. The simulated radiation characteristic depends not only on the mouth area, but also on the entire horn geometry.

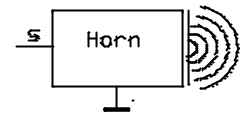


Fig. 101 Circuit diagram of the Horn element

The horn only has a noticeable effect on the sound source if its geometry fits. For the radiation of a surface without a horn, the radiation resistance increases with frequency up to the directivity frequency. From that point, it converges to a constant value. Similar conditions apply for radiation via a horn, the increase in radiation resistance depends on the frequency, the geometry and the rough size of the horn. The frequency from which the radiation resistance starts to converge to a constant value is the horn frequency: another value that depends on the geometry. The constant value to which the radiation resistances converge is the same with or without a horn. By skillfully shaping the horn, the designer tries to reduce the horn frequency to below the directivity frequency of the sound source. In an ideal case, the source is thus loaded with a constant radiation resistance in the whole frequency range that is of interest.

In the type of design most frequently encountered, the sound source is an electrodynamic transducer with a resonance frequency  $f_s$  and its qualities  $Q_{es}$  and  $Q_{ms}$  (see *Def\_Driver*). With free radiation, the curve of the radiation resistance and that of the diaphragm velocity of the driver complement one another in the frequency range between  $f_s$  and the directivity frequency such that the sound reproduction is to some extent constant. If the radiation is via an appropriate horn that loads the driver with a constant radiation resistance even below  $f_s$ , the curve of the reproduction is equivalent to that of the diaphragm velocity. It has a band-pass character. The mid frequency is the resonance of the driver  $f_s$ . The bandwidth of the reproduction is all the greater the smaller the qualities of the driver  $Q_{es}$  and  $Q_{ms}$ . Usually we attempt to reduce the electrical quality by reinforcing the motor using huge magnets. This increases not only the reproduction band, but also, of course, the overall efficiency of the driver.

The efficiency of the system can be further increased if a pressure chamber is installed between the diaphragm and horn throat. It is produced automatically if the area of the horn throat is smaller than that of the diaphragm. Its volume is  $V_f$ . This, thus, produces a velocity transformation.  $V_f$ , however, installs an acoustic low pass at the upper transmission end, so that, although the level increases, the bandwidth decreases. The acoustic low pass thus produced is at least 2nd order. That means that the transmission characteristic can be adjusted. In practice, one of the distinguishing features of a good horn designs is how well this effect has been exploited to increase the transmission bandwidth.

In addition to the pressure chamber, any further chambers and duct systems determined by the design are parallel to the horn inlet. For accurate simulation, reproduce these using the *Duct* and *Enclosure* elements.

The equilibrium of sound pressure and velocity in the horn and with respect to frequency are of course described by the wave equation. The problem is that they cannot be solved - at any rate not precisely and not for every horn shape. There are therefore restrictions. Any horn shape transmits sound. However, the means for describing, and therefore simulating this are extremely complicated.

A horn is a hybrid as regards the radiation problems in housings and radiation into free space. The former gives us a limited set of eigenfunctions to describe the sound field inside, e.g., a rectangular cavity. In the case of a horn, there are infinitely more sets of eigenfunctions - for each position inside the horn you have to determine a

new set of resonances. These natural frequencies, together with some other effects, are the cause of the often-mentioned 'horn sound'.

The higher modi of a waveguide are more strongly forced if there is an additional pressure gradient towards the horn walls. This may be caused by the velocity distribution at the horn throat, by obstacles inside the horn, by non-rigid horn walls and by the type of horn flare.

A horn flare that is not constant or horn-forms which differ from circular cross section enhance the natural frequencies of the waveguide. Because the equipotential lines are more distorted, the directivity of the horn will not be constant at high frequencies. The wave travels through the waveguide along several zigzag or helical paths.

## Horn function

In the current version of the program AkAbak evaluates the radiation and loading of horn forms that obey the Webster equation which describes the transmission in the basic horn mode along the axis. These horn forms are described by the horn function, which provides the cross sectional area at a distance from the horn throat [Sal, Morl]. It is assumed that the shape varies continuously. Other horn forms, with segments described by different horn functions, have to be modeled by cascading *Waveguide* elements. The horn function is a combination of exponential functions. The Salmon factor T controls their characteristics:

$$S(z) = S_{Th} \cdot [\cosh(k_0 \cdot z) + T \cdot \sinh(k_0 \cdot z)]^2 \quad (\text{Equ. 1})$$

where

z:	Argument co-ordinate of horn from horn throat [m]
S(z):	Cross sectional area at point z [m <sup>2</sup> ]
S <sub>Th</sub> :	Cross sectional area of horn throat [m <sup>2</sup> ]
ko:	Wave factor of horn frequency fo: $k_0 = 2\pi \cdot f_0/c$ , where c: velocity of sound
T:	Salmon factor

Equation 1 can also be rewritten as:

$$S(z) = S_{Th} \cdot [t_1 \cdot e^{k_0 \cdot z} + t_2 \cdot e^{-k_0 \cdot z}]^2 \quad (\text{Equ. 2})$$


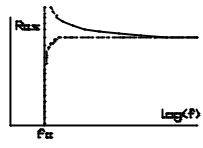
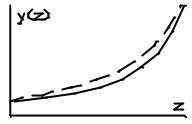
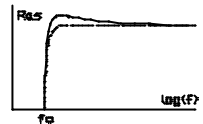

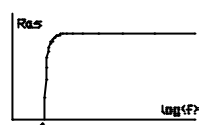
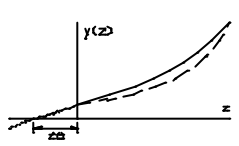
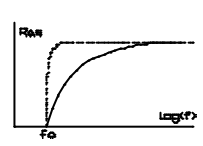
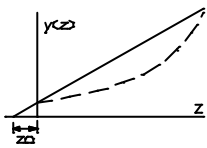
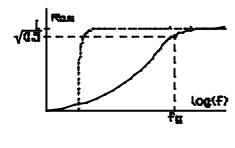
where

$$t_1 = \frac{1+T}{2} \quad \text{and} \quad t_2 = \frac{1-T}{2}$$

Depending on the value of the factor T, typical horn functions result. The following diagrams show the typical profile of the cross-section along the horn length, as well as the acoustic radiation resistance at the horn throat, on a logarithmic frequency scale. The dotted graph is in each case that of the purely exponential horn as a comparison. The curves represent reflection-free horns (infinitely long). In practice, amplitude fluctuations are superimposed as a result of the non-constant transition to the outside world at the horn mouth.

Ras is the acoustic power radiated from the horn. The imaginary component at the horn throat is also highly frequency dependent. It corresponds to a mass that loads the sound source. The maximum of the radiation reactance is at horn frequency fo, and its value is approximately equal to the mass of the air in the horn.



Name	T	Cross section $y(z) = \sqrt{S(z)}$	Radiation resistance $R_{as}$ as a function of frequency (log)
catenoid	0		
cosh	$0 < T < 1$		
exponential	1		
sinh	$1 < T < \infty$		
conical	$\infty$		

### Horn frequency $f_0$ for combined exponential horns

As can be seen from the radiation resistance, for exponential horns  $R_{as}$  rapidly becomes zero at  $f_0$ . In an infinitely long horn, no more sound is radiated below  $f_0$ . In this case all the energy is used to move the air mass in the horn. With normal horns, the reproduction above  $f_0$  ripples slightly, below it the level decreases rapidly and the diaphragm movement increases.  $f_0$  is an important parameter for designing horns. For a purely exponential horn, for example,  $f_0$  is set approximately one octave below the lowest operating frequency of the horn system. The program calculates  $f_0$  from the given dimensions of the horn. Solving equation 2 for  $f_0$ :

$$f_0 = \frac{c}{2\pi \cdot \text{Len}} \cdot \ln \left[ \frac{1}{2 \cdot t_1} \cdot \sqrt{\frac{S_{Mo}}{S_{Th}}} \cdot \left( 1 + \sqrt{1 - 4 \cdot t_1 \cdot t_2 \cdot \frac{S_{Th}}{S_{Mo}}} \right) \right]$$

where

Len: horn length [m]

$S_{Mo}$ : cross sectional area of the horn mouth [m<sup>2</sup>]

## Special horn functions

### Catenoid (T=0)

Catenoid horn shapes are characterized by an extremely high radiation resistance  $R_{as}$  slightly above the horn frequency  $f_0$ . They are typically used for signal horns or sirens, whose fundamental tone is matched to the maximum of  $R_{as}$ . Many wind instruments, for example trumpets, also have a catenoid horn shape.

The cross-sectional area first increases slowly and then very rapidly to the horn mouth. In this case, equation 2 can be simplified to:

$$S(z) = S_{Th} \cdot \cosh^2(k_0 \cdot z), \text{ where } t_1 = t_2 = 0,5$$

### Exponential (T=1)

The purely exponential horn is the horn shape most widely used for loudspeaker systems. Not only because of the simple horn function, as is shown by equation 2 with  $T=1$ , but also because of the effective reproduction behavior with, at the same time, smaller dimensions compared with other shapes. When  $T=1$ , the reproduction characteristic is maximally flat. Purely exponential horns remind us of the so-called Butterworth filter in filter theory.

$$S(z) = S_{Th} \cdot e^{2k_0 \cdot z}, \text{ where } t_1 = 1, t_2 = 0$$

The horn frequency  $f_0$  here is:

$$f_0 = \frac{c}{4\pi \cdot L_{en}} \cdot \ln \left[ \frac{S_{Mo}}{S_{Th}} \right]$$

### Conical (T = ∞)

The conical horn is the simplest horn shape. The cross-section increases linearly. The horn function Eq. 2 is simplified to:

$$S(z) = S_{Th} \cdot \left( \frac{z}{z_0} + 1 \right)^2$$

where

$z_0$ : distance from cone tip to horn mouth [m]

The transition from equation 2 to this shape is achieved by the critical value  $k_0 \rightarrow 0$ . That means, conical horns have no horn frequency. The radiation impedance is similar to that of the diaphragm. As can be seen from the diagram of the curve for  $R_{as}$ , a different frequency is defined in this case.  $f_g$  is a critical frequency at which the radiation resistance has fallen by 3dB:

$$f_g = c / (2\pi \cdot z_0)$$

$z_0$  is the design constant - the distance from the cone tip to the horn mouth, if the cross-sectional area is circular. For  $f_g$  to become low, the value of  $z_0$  must be high enough. As the diagram of the horn contour shows, the opening angle of the horn is thus reduced.

Compared with exponential horns, conical horns have a precisely defined radiation characteristic. They belong to a group for which the one-dimensional wave equation can precisely describe the states in the horn [Ged1]. It is frequently used as the shape for so-called 'constant directivity' horns, which manifest a homogeneous radiation characteristic over a wide frequency range.

Another advantage is that the cross-sectional area in the vicinity of the horn throat increases more rapidly than for exponential horns. This thus reduces the sound pressure in the horn as well as distortions due to the compressibility of air.

The only disadvantage of this horn shape is its size.

**Cosh ( $0 < T < 1$ )**

This shape is called the hyperbolic cosine horn, since in this range the horn function (Eq. 1) is more described by the cosh component. Depending on the value of the factor  $T$ , a horn is obtained with the characteristics of a catenoid horn ( $T=0$ ) or with that of the pure exponential ( $T=1$ ).

**Sinh ( $1 < T < \infty$ )**

If the factor  $T$  is greater than 1, the horn is known as hyperbolic sine. Its characteristics are between those of the purely exponential ( $T=1$ ) and those of the conical horn ( $T \rightarrow \infty$ ). The horn function intersects the abscissa at a distance  $z_0$  from the horn throat (see diagram) at:

$$z_0 = \frac{c}{4\pi \cdot f_0} \cdot \ln \left( \frac{T+1}{T-1} \right)$$

Above a value of 10 to 20 of the horn factor  $T$ , the horn behaves virtually like a conical horn.

The sinh and cosh horn functions are often helpful for approximating a given horn contour.

**Radiation cone**

*Dialog: Net/ Acoustic/ Horn rad. cone*

As you can imagine, it is not a trivial matter to describe the radiation of horns into free space. The velocity distribution at the horn mouth is not at all evenly distributed across the mouth area. In addition, there are diffraction problems at the mouth edges.

At low frequencies the radiation of a flat diaphragm with the form and size of the horn mouth is in many cases not too bad an approximation. If you want to use this model instead of the `Horn` model, you should cascade a `Waveguide` and a `Radiator` element. The latter is most appropriate to model horn-loudspeaker in the low frequency range.

For the simulation of the radiation in the mid and high frequency range of horns to be useful, you also have to take into account the flux in the direction of the two other dimensions. To a first approximation, AkAbak works with the 'radiation cone' of the horn: The `Horn` element is similar to the `Waveguide` element, but includes a special radiator. This radiator consists of multiple subradiators, each with the directivity of a rectangular aperture. These sub-radiators are placed on a sphere with vertical and horizontal convexities. The convexity of this radiation sphere can be entered manually or calculated automatically.

In many cases, the radiation cone should be refined subsequently. In future versions of AkAbak, this model will be improved. Until then, this approximation will be adequate and will correctly show causes and effects with just a handful of parameter.

The program calculates the radiation cone of a horn with exponentially flared walls automatically taking the point in the horn at which a ray starting from the center of the horn throat meets the horn wall tangentially (Fig. 102). Consider a cone whose tip lies in the center of the cross-sectional area of the horn throat and whose apex angle is equal to the angle of view when one looks into the horn from the horn throat. This apex angle can, of course, differ in width and height, depending on the horn shape.

The convexity of the radiation cone is calculated from the curvature of the horn wall at the point at which the radiation cone contacts the wall (arc between  $W_c$  and  $H_c$  in Fig. 102). The surface is subdivided into numerous small rectangular areas. Each of these areas possesses the radiation characteristics of a rectangular surface. The greater the curvature of the radiation surface, the more radiation surfaces are installed. From this, the radiation characteristics can be ascertained. If the apex angle of the radiation cone is relatively narrow in width or height, or in both directions, and the curvature of the horn wall at this point is small, then the radiation characteristics of the horn resemble that of a single, small, rectangular surface or diaphragm. If the curvature of the horn wall at this point is greater, the radiation surface is more greatly curved. The radiation then resembles more a small dome diaphragm. If, on the other hand, the apex angle is large, and the curvature of the horn wall is small, the radiation characteristics resemble more those of a large rectangular diaphragm, so that their directivity frequency is at lower frequencies. If the horn wall also has a greater curvature at this point, then

AkAbak installs more radiation elements again. The radiation characteristics then resembles those of a relatively large dome diaphragm.

The radiation cone only forms the radiation characteristics and has no influence on the efficiency or on the position of the acoustic center of the horn.

The program only simulates horns with a rectangular cross-sectional area, although you can also enter the mouth area with a diameter  $d_{Mo}$ . In this case not a round horn, but one with a square cross-section is simulated.

### Manual entry of radiation cone

As soon as you enter the parameters of the radiation cone in the script, the radiation elements are arranged according to these details. It is advisable to enter these parameters by hand whenever the horn shape does not have a width and a height corresponding to the simple horn-forms which are supported by the program. A horn which is constructed with the help of multiple waveguides can have a Horn element which models the mouth part. In this case the Horn elements models only a small segment and the radiation cone must be entered manually. Note, that in this case, a situation can occur where the cone is fitted anywhere but in the Horn element itself. The radiation cone only forms the radiation characteristics and has no influence on the efficiency or on the position of the acoustic center of the horn.

You can edit and display the evaluated parameter of the radiation cone if you place the script cursor in the lines of the Horn-element and open the Net/ Acoustic/ Horn Radiation Cone-dialog.

You can enter each position individually. Units: meter [m] or inch [in].

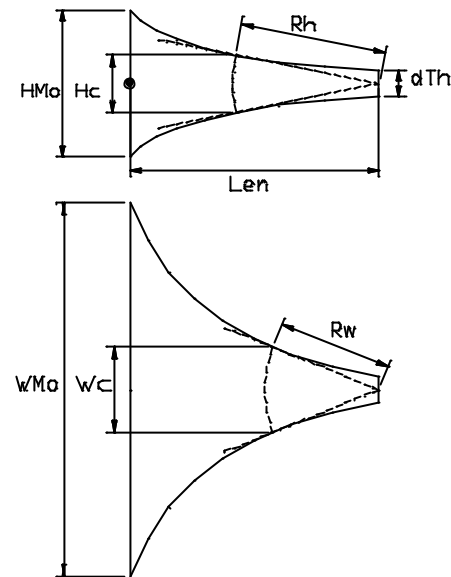


Fig. 102 radiation cone of the Horns

$R_w = \dots m$  Horizontal cone-radius. Has to be entered together with  $w_c$ .

$w_c = \dots m$  Horizontal cone-width. Has to be entered together with  $R_w$ .

$R_h = \dots m$  Vertical cone-radius. Has to be entered together with  $H_c$ .

$H_c = \dots m$  Horizontal cone-width. Has to be entered together with  $R_h$ .

With radial horns, the radiation cone is rotated through  $90^\circ$  by the keyword *Vertical*.

## Horn designs

There is naturally an infinite variety of horn models. But most of them greatly resemble one another and three types can be distinguished that the program can simulate:

- continuous horns
- radial horns
- slit horns

### Continuous Horns

An example is drawn in Fig. 103. The cross-sectional area given by the horn function is in this case mapped onto a cross-section that is rectangular, or round at the horn throat, so that no steps or slots are produced, and so that the convexity of the wavefront in the horn is curved approximately equally in the height and breadth. In

practice, that means that the ratio of width to height of the horn mouth ( $W_{Mo}/H_{Mo}$ ) must not be too extreme. If one side is substantially larger than the other (for example  $W_{Mo}/H_{Mo} > 3$  or  $W_{Mo}/H_{Mo} < 0.3$ ), the horn behaves more like a radial horn, and should be entered as such.

The program constructs this design for calculating the radiation cone by default, and assumes that the ratio  $W_{Mo}/H_{Mo}$  continues throughout the horn. If the horn shape departs greatly from this, then you should enter the data for the radiation cone ( $R_h$ ,  $R_w$ ,  $H_c$ ,  $W_c$ ) manually (see also slit horns).

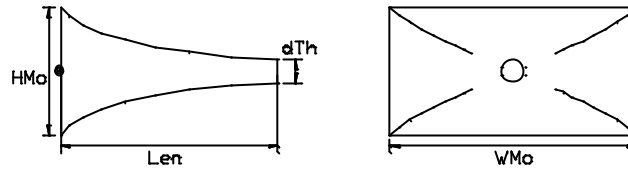


Fig. 103 Rectangular continuous exponential horn

## Radial Horns

From a certain point, these horns expand linearly in the horizontal direction and exponentially in the vertical direction, as illustrated in Fig. 104. Here, radial horns must have an exponential horn function. Conical horns are either constant or slit horns.

Exceptionally, in the case of radial horns, their cross-section is curved horizontally. It follows the curve of the horn mouth. With most designs, the radial horn consists of two parts. Up to the length  $LenTh=$  it enlarges constantly from the predominantly round horn throat to a rectangular cross-sectional area. To this is connected the so-called horn bell. The height of the exponentially profiled sides is given by the area calculated from equation 1 or 2, divided by the arc length. There is often first a tapering in the height, since the arc length increases faster than the horn function.

To calculate the radiation cone, the program first requires the length of the horn  $Len=$  and the width  $W_{Mo}=$  and height  $H_{Mo}=$  of the horn mouth flange. The horn mouth flange is the rectangle formed by the height of the exponential side and the chord of the circular section of the horn bell. For the program, the center point of the horn mouth flange is the reference point of mounting that is used for positioning the horn.

The geometry of the horn bell results from, among other factors, the arc height  $H_{Arc}=$  and the length of the horn extension  $LenTh=$ . If you enter  $H_{Arc}=$ , the horn automatically becomes a radial horn, even if you enter the radiation cone manually. The program tests the data entered for plausibility.

At very high frequencies, only the horn extension is still active. The radiation characteristic then corresponds predominantly to that of a rectangular diaphragm with the same area and shape as the end of the horn extension. The program does not take this fact into account.

In the case of a radial horn, the broad side  $W_{Mo}=$  is always the linear or radial side. The height  $H_{Mo}=$  is used to indicate the exponential side. If the radial horn is installed rotated through  $90^\circ$  (rotation about the z-axis), so that the linear walls lie at the top and bottom, you have to enter the keyword *Vertical*.

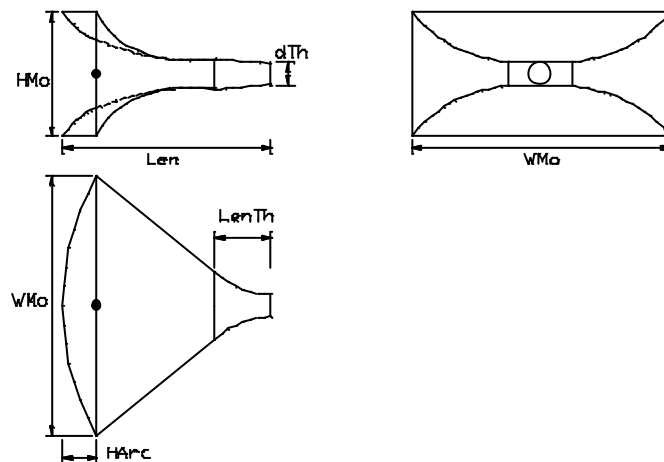


Fig. 104 Radial-horn

## Slit horns

This class of horn shapes includes all horns whose cross-sectional shape changes suddenly, without departing from the horn function to be described. Otherwise you have to cascade several `Waveguide` elements with their individual horn functions. The horn function in this case may be exponential or conical, as is drawn, for example, in Fig. 105.

The slit has a diffracting effect. The narrower the slit, the broader the radiation characteristics can be made. The directivity is formed by the following horn bell. At very high frequencies the directivity is similar to a rectangular surface with width  $w_{s=}$  and height  $H_{s=}$ .

In the case of exponential horns, you have to enter the radiation cone manually. The tip of the cone then lies not in the center of the horn throat, but where the slit is located.

For conical horns, there are two extra parameter:

$w_{s=}$  and  $H_{s=}$  (slit width and height)

The program then automatically constructs the radiation cone from this.

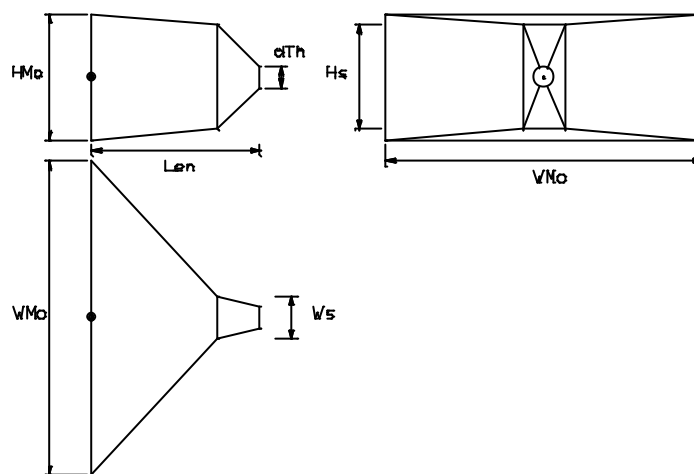


Fig. 105 Example of a conical slit horn

### ☒ Examples

#### 1. Typical case of a horn-driver combination.

The horn driver contains the keyword `Meas_DoNotModify`, so that the program does not deduct the radiation impedance at resonance. The diaphragm of 2g has a diameter of 5cm. The diaphragm type is of a concave dome. The depth of dome is  $tD1=-5\text{mm}$ . The mass and area reduction of the diaphragm starts at  $f_p=4500\text{Hz}$ . Together with the suspension and the compliance of the enclosure volume behind the diaphragm, it resonates at 800Hz. The driver has a powerful motor, so that a high bandwidth is ensured by a low quality factor.

In the network, the `Driver` element connects the driver to nodes 1 and 0 at the electrical side and to nodes 2 and 0 at the acoustic side. The reverse side of the diaphragm is connected to ground (node 0), since the effect of the enclosure here is already included in the parameter.

An acoustic duct (`Duct`) closed at one end should be taken into account between the diaphragm or voice coil and magnet in the horn driver. From an acoustic point of view, the transition between the diaphragm or voice coil and the magnet in the horn input is very complicated. Depending on the design of the horn driver, it may be necessary to attach further acoustic elements here (`Duct` and `Enclosure`, see exercise).

The horn is also at node 2. As  $T=1$  is specified, the horn function is purely exponential. The horn throat has a diameter of one inch. The cross-sectional area enlarges constantly along the horn length of 30cm to a rectangular shape at the mouth of  $40\text{cm} \times 20\text{cm}$ . Between the driver diaphragm and the horn throat, there is a pressure chamber with a volume of  $1\text{cm}^3$ .

The horn is positioned at the origin of the baffle coordinate system.

```

Def_Driver      'HornDriver'
  Meas_DoNotModify
  dD=5cm  tDl=-5mm  |Concave  dome
  fp=4500Hz
  fs=800Hz  Mms=2g
  Qms=3.5  Qes=0.5  Re=12ohm  Le=70uH

System
  Driver  Def='HornDriver'  Node=1=0=2=0
  Duct  Node=2=0          | cavities in horn driver
    WD={pi*5e-2}          | circumf. of magnet gap with formula parser
    HD=1mm                | magnetic gap width
    Len=3cm
  Horn  Node=2  T=1
    dTh=1in  WMo=40cm  HMo=20cm
    Len=30cm  Vf=1cm3
    x=0  y=0  z=0  HAngle=0  VAngle=0  WEdge=40cm  HEdge=20cm

```

2. Similar to the first example, except that it has a larger horn with a radial shape. The horn mouth flange is 60cm wide and 12cm high. It is 11 cm to the arc of the horn bell. The horn extension has a length of 11cm. The horn is 50cm long.

```

Horn  Node=2  T=1
  dTh=1in  WMo=60cm  HMo=12cm
  Len=50cm  Vf=1cm3
  HArc=11cm  LenTh=11cm  |radial horn bell
  x=0  y=0  z=0  HAngle=0  VAngle=0
  WEdge=60cm  HEdge=12cm

```

3. The same dimensions as in the second example, except that the horn function is conical.

```

Horn  Node=2  Conical
  dTh=1in  WMo=60cm  HMo=12cm
  Len=50cm  Vf=1cm3
  x=0  y=0  z=0  HAngle=0  VAngle=0
  WEdge=60cm  HEdge=12cm

```

## Parameter

Horn	Keyword								
'...'	Identifier								
Node=s	Network nodes of the horn. The volume velocity flows from pole <b>s</b> to ground.								
T=	<p>Parameter of horn function (Eqs. 1,2), as described above. If not entered, the program sets T=1, thus specifying a purely exponential horn function. If the keyword <b>Conical</b> has been entered, the value of T is ignored.</p> <p>range: <math>0 \leq T \leq 100</math></p> <table> <tr> <td>T= 0</td><td>catenoid</td></tr> <tr> <td><math>0 &lt; T &lt; 1</math></td><td>cosh</td></tr> <tr> <td>T= 1</td><td>exponential (default)</td></tr> <tr> <td><math>1 &lt; T &lt; 100</math></td><td>sinh</td></tr> </table>	T= 0	catenoid	$0 < T < 1$	cosh	T= 1	exponential (default)	$1 < T < 100$	sinh
T= 0	catenoid								
$0 < T < 1$	cosh								
T= 1	exponential (default)								
$1 < T < 100$	sinh								
Conical	(Optional) If entered, the horn function describes a conical horn profile (see above). If the parameter T has been entered, it is ignored.								

Len= . . . m		Length of the horn center line from the horn throat to the horn mouth. For radial horns, <i>Len</i> extends from the horn throat to the arc height of the horn bell. If the horn is operated from a large pressure chamber, an end correction may have to be applied to the horn throat. In most cases this can be neglected, however. Unit meter [m] or inch [in].
HArc= . . . m	(Optional)	Radial horn. Arc height as shown in Fig. 104. When <i>HArc</i> is entered, the horn becomes a radial horn. Range: <i>HArc</i> > 0, but at most so high that the apex angle of the horn bell is not greater than 180°. Unit meter [m] or inch [in].
LenTh= . . . m	(Optional)	Radial horn. Length of the horn extension for radial horns. <i>LenTh</i> can be zero, so that the horn bell starts directly at the horn throat. Unit meter [m] or inch [in].
Vertical	(Optional)	Radial horn. Since radial horns are not symmetrical as regards their cross-section, the keyword <i>Vertical</i> is entered to install the horn rotated through 90°. The linear walls are then at the top and bottom.
Ws= . . . m Hs= . . . m	(Optional)	Conical horns. As shown in Fig. 105, <i>Ws</i> and <i>Hs</i> are the width and height respectively of the slit produced when the shape of the cross-sectional area changes suddenly. These entries, together with the mouth shape are used to calculate the radiation cone. <i>Ws</i> and <i>Hs</i> only apply to conical horns and are otherwise ignored. Unit meter [m] or inch [in].
Vf= . . . m <sup>3</sup>	(Optional)	Size of the pressure chamber volume at the entrance to the horn (from node <b>s</b> to ground). A pressure chamber is produced whenever the cross-sectional area is reduced. For example, when the diaphragm of a driver radiates into a horn with a smaller cross-sectional area. It is important that the dimensions of the pressure chamber are smaller than the wavelengths. Unit cubic meter [m <sup>3</sup> ], cubic inch [in <sup>3</sup> ] or liter [Liter, L].

### Throat cross section

See also chapter "Introduction/Diaphragm form parameter".

The specifications of throat cross section are calculated to a squared form by the program.

dTh= . . . m		Diameter of throat.
WTh= . . . m		
HTh= . . . m	Alternative	Width and height of throat.
STh= . . . m <sup>2</sup>	Alternative	Area of throat.

### Horn mouth cross section

See also chapter "Introduction/Diaphragm form parameter"

The specifications of mouth cross section are calculated to a squared form by the program.

WMO= . . . m		
HMO= . . . m		Width and height of the horn mouth. The width is horizontally, the height is vertically. For radial horns, <i>WMO</i> and <i>HMO</i> are, as with other horn shapes, the width and height of the horn mouth flange, where <i>WMO</i> is equal to the chord of the arc



section of the linear side. If the radial horn is rotated so that the linear horn walls are at the top and bottom, the keyword `Vertical` has to be entered. The horn mouth area has to be larger than the throat area. Unit meter [m] or inch [in].

<code>dMo=...m</code>	Alternative	Diameter of mouth. Circular cross section forms are calculated to squared forms by the program.
<code>SMo=...m2</code>	Alternative	Area of a mouth with squared form.

## Radiation

See also chapter "Introduction/Position of radiators, /Radiation environment"

The reference point of mounting is for all horn forms the center of the horn mouth flange (small dot in Fig. 103 to Fig. 105).

<code>x=...m</code>		Horizontal position.
<code>y=...m</code>		Vertical position.
<code>z=...m</code>		Axial position.
<code>HAngle=...°</code>		Horizontal position angle (-180°...180°)
<code>VAngle=...°</code>		Vertical position angle (-90°...90°)
<code>Reflection</code>	(Optional)	Included in the evaluation of reflections.
<code>WEde=...m</code>	(Optional)	Diffraction. Width of baffle.
<code>HEde=...m</code>	(Optional)	Diffraction. Height of baffle.
<code>dEdge=...m</code>	(Optional)	Alternative. Diameter of baffle.
<code>NoRad</code>	(Optional)	No radiation takes place.

See also: `Waveguide`, `Radiator`

# Waveguide

*Dialog: Net/ Acoustic/ Waveguide*

`Waveguide` is a network element that does not radiate, an intermediate between `Duct` and `Horn`, i.e. an acoustic duct whose cross-sectional area changes, or a horn that does not radiate. It is usually used to simulate the transmission behavior of composite waveguides with varying cross-section and horn functions. The same applies to the `Waveguide` element as to the `Horn` element, except for the components concerned with radiation.

A `Waveguide` element is a horn with a certain length, a throat area and a mouth area. The cross-sectional area of the waveguide increases constantly from the horn throat to its mouth. At the horn throat there is mounted the sound source. This source may be a `Driver`, `Duct`, `Waveguide` or `Coupler`, or other acoustic element. At the waveguide mouth another acoustic element is connected. The radiation is usually performed by a `Horn` element at the end of the chain. Of course, it is also possible to install a `Radiator`, and thereby to obtain a horn. This, then, does not take into account radiation behavior of the curved wavefront in the waveguide. In the case of long horns with a small apex angle, the difference from the `Horn`-elemnt is not very great, and this fact can be used to speed up the computation.

A `Waveguide` element does not concern itself with the shape of the funnel cross-section. Only the areas of the horn throat and mouth are therefore used. Unlike with the `Horn` element, therefore, no distinction is made between the various designs.

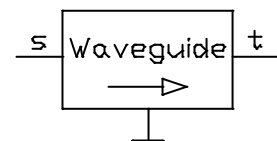


Fig. 106 Circuit diagram of `Waveguide` element

The waveguide cross-section expands according to horn function, as is the case with the `Horn` element. It is also possible to operate the `Waveguide` element in reverse, so that the wave travels to the tapered horn throat.

`Waveguide` is actually a four pole element. Since, however, two connections are always at ground, the entry for the network nodes only requires two nodes, as with the `Duct` element. Pole `s` is the waveguide throat and pole `t` the waveguide mouth. At the input, as with the `Horn` and `Duct` element, there is applied a compliance `Caf` from pole `s` to ground to allow a pressure chamber with the volume `Vf` to be installed if necessary. If the waveguide is required to taper, the input is at pole `t`. The pressure chamber is in this case at the output of the `Waveguide` element.

If the `Waveguide` elements follows a horn for the radiation, the automatic calculation of the radiation cone in the horn is only approximately correct if the throat area of the horn is sufficiently small. If, therefore, a large proportion of the entire horn is described using `Waveguide` elements, the horn cross-section will have already been increased. In this case you should enter the radiation cone of the horn manually (Fig. 107). It is best to make a drawing of the entire horn and then fit the radiation cone into the total horn construction, as described under `Horn`.

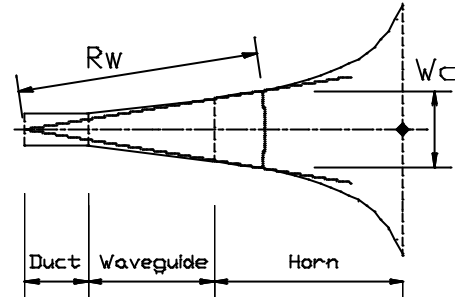


Fig. 107 Radiation cone of a composite horn

### ☑ Example

Example of a composite horn as drawn in Fig. 107. A driver first radiates into a 10cm long acoustic duct with a pressure chamber of 1cm<sup>2</sup>. This is followed by a conical waveguide of 20cm length. The radiation is from a `Horn` element with a purely exponential profile. You should enter the radiation cone manually, as drawn in Fig. 107.

The horn driver contains the keyword `Meas_DoNotModify`, so that the program does not deduct the radiation impedance at resonance. The diaphragm of 2g has a diameter of 5cm. The diaphragm type is of a concave dome. The depth of dome is `tD1=-5mm`. The mass and area reduction of the diaphragm starts at `fp=4500Hz`. Together with the suspension and the compliance of the enclosure volume behind the diaphragm, it resonates at 800Hz. The driver has a powerful motor, so that a high bandwidth is ensured by a low quality factor.

In the network, the `Driver` element connects the driver to nodes 1 and 0 at the electrical side and to nodes 2 and 0 at the acoustic side. The reverse side of the diaphragm is connected to ground (node 0), since the effect of the enclosure here is already included in the parameter.

An acoustic duct (`Duct`) closed at one end should be taken into account between the diaphragm or voice coil and magnet in the horn driver. From an acoustic point of view, the transition between the diaphragm or voice coil and the magnet in the horn input is very complicated. Depending on the design of the horn driver, it may be necessary to attach further acoustic elements here (`Duct` and `Enclosure`, see exercises).

The horn is positioned at the origin of the baffle coordinate system.

```
Def_Driver      'HornDriver'
  Meas_DoNotModify
  dD=5cm  tD1=-5mm  |Concave dome
  fp=4500Hz
  fs=800Hz  Mms=2g
  Qms=3.5  Qes=0.5  Re=12ohm  Le=70uH

System  'S1'
  Driver  Def='HornDriver'  Node=1=0=2=0
  Duct    | cavities in horn driver
    Node=2=0
    WD={pi*5e-2}  | circumference magnet gap with formula parser
    HD=1mm        | magnetic gap width
    Len=3cm
  Duct    |Begin of horn
    Node=2=3  dD=1in  Len=10cm  Vf=1cm3
```

```

Waveguide
  Node=3=4  Conical
  dTh=1in  WMo=10cm  HMo=10cm  Len=20cm
Horn
  Node=4  T=1
  WTh=10cm  HTh=10cm  WMo=40cm  HMo=20cm  Len=30cm
  Rw=40cm  Wc=11.8cm  Rh=40.5cm  Hc=10.7cm
  x=0  y=0  z=0  HAngle=0  VAngle=0

```

## Parameter

Waveguide		Keyword
'...'		Identifier
Node=s=t		Network node of the Waveguide. The volume velocity at the entrance - the funnel throat - 'flows' from pole <b>s</b> to ground and at the outlet - the funnel mouth - from pole <b>t</b> to ground.
T=	(Optional)	<p>Parameter of horn function (Eqs. 1,2), as described for the element Horn. If T is not entered, the program sets T=1, thus specifying a purely exponential horn function. If the keyword Conical has been entered, the value of T is ignored.</p> <p>range: <math>0 \leq T \leq 100</math></p> <p>T= 0                  catenoid</p> <p><math>0 &lt; T &lt; 1</math>        cosh</p> <p>T= 1                  exponential (default)</p> <p><math>1 &lt; T &lt; 100</math>     sinh</p>
Conical	(Optional)	If entered, the horn function describes a conical horn profile (see above). If the parameter T has been entered, it is ignored.
Len=...m		<p>Length of the horn center line from the horn throat to the horn mouth.</p> <p>If the horn is operated from a large pressure chamber, an end correction may have to be applied to the horn throat. In most cases this can be neglected, however.</p> <p>Unit meter [m] or inch [in].</p>
Vf=...m3	(Optional)	<p>Size of the pressure chamber volume at the entrance to the horn (from node <b>s</b> to ground). A pressure chamber is produced whenever the cross-sectional area is reduced. For example, when the diaphragm of a driver radiates into a horn with a smaller cross-sectional area. It is important that the dimensions of the pressure chamber are smaller than the wavelengths.</p> <p>Unit cubic meter [m<sup>3</sup>], cubic inch [in<sup>3</sup>] or liter [Liter, L].</p>

### Horn throat cross section

See also chapter "Introduction - diaphragm form parameter"

dTh=...m		Diameter of throat.
WTh=...m		
HTh=...m	Alternative	Width and height of throat.
STh=...m2	Alternative	Area of throat.

**Horn mouth cross section**

See also chapter "Introduction - diaphragm form parameter"

WMO= . . . m

HMO= . . . m

Width and height of the horn mouth.

dMO= . . . m

Alternative

Diameter of mouth.

SMO= . . . m<sup>2</sup>

Alternative

Area of a mouth.

See also:

Horn, Duct

# Filter

## Filter Element

Dialog: *Filter/ Filter dialog*

*Ctrl + F*

The `Filter` elements total transfer function consists of two parts. The first is the rational transfer function and the second term is any function.

## Rational function

The `Filter` element describes a rational transfer function of the output voltage  $U_a$  to the input voltage  $U_e$  in the form:

$$TF(s) = \frac{b_m \cdot s^m + b_{m-1} \cdot s^{m-1} + \dots + b_1 \cdot s + b_0}{a_n \cdot s^n + a_{n-1} \cdot s^{n-1} + \dots + a_1 \cdot s + a_0} \cdot e^{-to \cdot s}$$

where

$TF(s)$

$U_a/U_o$

$b_i$

coefficient of the numerator, denominator polynomials

$m, n$

degree of the numerator, denominator

$to$

delay time [s]

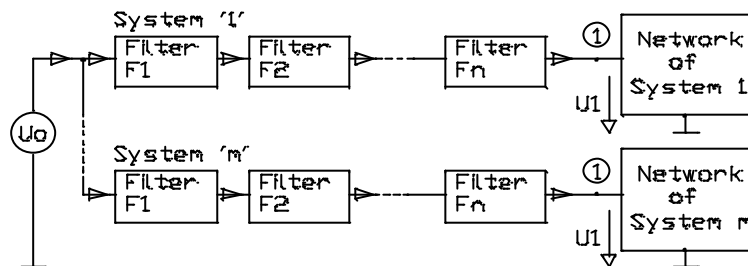


Fig. 108 Cascading of the filter elements in each system

The `Filter` element is not a network element, nor is it positioned by entering a node. The `Filter` element itself is free of feedback, i.e. the load at the output side is not reflected at the filter input.

`Filter` elements weight the input voltage  $U_1$  of the following current network, which is applied across nodes 1 and nodes 0 (Fig. 108). As soon as a `Filter` is entered in a system, the input voltage  $U_0$  of the system is multiplied by the transfer function of the filter. If several filters are entered, their transfer functions are multiplied (except when a `Feedback` formula is present, see below). The total transfer is thus  $U_1/U_o = F_1 \cdot F_2 \cdot F_3 \dots$ , if  $F_1$ ,  $F_2$  etc. are the names of the filters.

## Filter before the systems

Usually all `Systems` are switched in parallel and are driven by the source voltage  $U_{in}$  which in turn is specified by the simulation control dialogs. By this the `System` input voltage  $U_0 = U_{in}$ .

It is also possible to weight the source voltage  $U_{in}$  with a filter cascade (not shown in the figure). In this case  $U_0$  of the `Systems` is not equal  $U_{in}$ . These filters have to be listed in the script before the first system. However, it is not possible to use `Feedback` in this case.

## Binary structures

It is not possible to enter binary tree structures of filter circuits (Fig. 109) directly (except for those mentioned in the last paragraph). In this case the filters have to be located before the branch, divided into two identical filters and inserted into the respective systems.

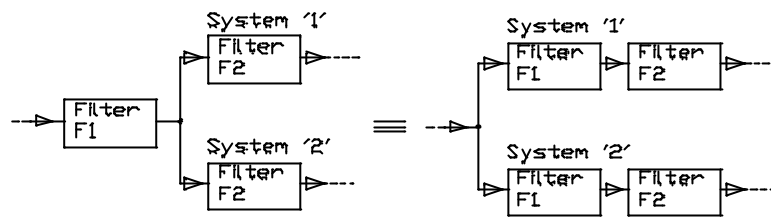


Fig. 109 Resolution of a binary filter structure

## Order of the transfer function

The maximum order of a filter element is limited to  $m=30$  and  $n=30$ . The accuracy of computation decreases with increasing degree of the filter function, so that, depending on the coefficient sensitivity, you should treat results of approx.  $m=20$  and  $n=20$  and above with caution.

## Filter frequency

The coefficients are values normalized to any pole frequency. To be able to use the filter in the range of technical frequencies, enter the denormalization or filter frequency  $f_0$ . This parameter is outside the equation system and its syntax is that of the script interpreter.

If you do not enter  $f_0=$ , AkAbak sets  $f_0=1\text{Hz}$ . You therefore do not need to enter  $f_0=$  if the filter represents only one amplifier. For example:  $\{b_0=-3\}$  for an inverting amplifier with a gain of three.

Apart from the purely rational transfer function, the `Filter` element also accepts entries that correspond to an ideal signal delay ( $t_0$  and  $z_0$ ).

## General formula system

Additionally to the rational transfer function it is possible to enter any complex function system. This formula system follows the same syntax as the formula parser described in chapter "Introduction/Formula Parser". The additional formula system has to follow the transfer function specification and is also enclosed in curly brackets. There are three further reserved identifiers:

$f$	abscissa frequency in [Hz]
$w$	abscissa frequency in cycles [ $s^{-1}$ ]
TF	result of the transfer function

For example:

```
Filter 'F1'
  fo=1kHz
  { b0=1; a2=1; a1=sqrt(2); a0=1; } |rational transfer function
  { exp(jw/500) - TF }              |additional formula system
```

### ☒ Examples

1. 2nd order Butterworth low pass. The filter frequency is 1000Hz. The value of the denominator coefficient  $a_1$  is calculated:  $a_1 = \sqrt{2}$ .

```
Filter 'F1'
fo=1kHz
{b0=1;
a2=1; a1=sqrt(2); a0=1; }
```

2. The same function as in Example 1. In this case a constant is established, with the aid of which the filter can be scaled in the frequency level. With the entered value of  $so=1.5$ , the filter frequency (in the case of Butterworth, the -3dB-Frequency, too) has been scaled to 1500Hz.

```
Filter 'F1'
fo=1kHz
{so=1.5; |Scale factor
b0=1;
a2=1/sqr(so); a1=sqrt(2)/so; a0=1; }
```

3. The same as in Example 2; except that it is the associated high-pass in this case. The denominator coefficient  $a2$  is assigned the value of the numerator coefficient  $b2$ .

```
Filter 'F1'
fo=1kHz
{so=1.5; |Scale factor
b2=1/sqr(so);
a2=b2; a1=sqrt(2)/so; a0=1; }
```

4. Example of an integrator:  $U_a/U_e = 2\pi \cdot 1000/s$ .

```
Filter 'F1'
fo=1kHz
{b0=1; a1=1}
```

5. Example of a 1ms delay element.

```
Filter 'F1'
to=1ms
```

6. Example of a delay element corresponding to a sound path of 10 cm.

```
Filter 'F1'
zo=10cm
```

7. Example of an inverting amplifier with a gain of 3.

```
Filter 'F1'
{b0=-3}
```

8. Example of the use of the additional formula system. Simple digital IIR-lowpass-filter.  $w$  is the abscissa frequency normed to 10kHz. {  $b0=1$  } is the residuum of the rational transfer function which always has to be entered even if you do not use it.

```
Filter
{ b0=1 }
{ z = exp(jw/10e3); z1 = 0.6;
G = (1 - abs(z1))/(z - z1); }
```

9. Example of the use of the additional formula system. Same as before but with an (analog) antialiasing filter expressed by the rational transfer function.  $TF$  is the link to the rational transfer function. This variable contains the result.

Filter

fo=5kHz

```
{ b0=1; a8=1; a7=5.125831; a6=13.137071; a5=21.846151;  
  a4=25.688356; a3=21.846151; a2=13.137071; a1=5.125831; a0=1; }  
{ z = exp(jw/10e3); z1 = 0.6;  
  G = TF*(1 - abs(z1))/(z - z1); }
```



## Parameter

Filter		Keyword
'...'		Identifier
f <sub>0</sub> =...Hz		Filter frequency. The frequency variable of the transfer function is regarded as to be normalized to f <sub>0</sub> . If f <sub>0</sub> is not entered, the program sets f <sub>0</sub> =1Hz. Unit hertz [Hz]
{ b <sub>0</sub> =...; b <sub>1</sub> =...; ...; a <sub>0</sub> =...; a <sub>1</sub> =...; ...; }		<p>Coefficients of the polynomials of the rational transfer function. The coefficients are entered as the formula system (see chapter Introduction/Formula Parser). The formula system is written in braces (curly brackets) and can go over several lines. There is no identifier. The coefficients of the numerator are called b<sub>0</sub>=, b<sub>1</sub>=, b<sub>2</sub>=, ... and those of the denominator a<sub>0</sub>=, a<sub>1</sub>=, a<sub>2</sub>=, .... These function names are reserved here. Their values are entered in the usual way, e.g.: b<sub>0</sub>=1. However, each entry has to be separated by a semicolon. A space is permitted before and after the equal sign. Any coefficients not entered are set to zero by default. They may be entered in any order. In addition to entering numerical values, you may also assign expressions, symbolic constants or previously defined coefficients. You can not use the abscissa frequency here, for this, use the additional formula system described below. Comments inside the curly brackets are also permitted. They commence with   and end either with the same character or at the end of the line. Constants that have previously been computed in the definition Def_Const can also be assigned. The values of several filters can thus be changed centrally. The values calculated for the coefficients have to be real values. That does not mean that they may not have been previously obtained by complex calculation.</p>
{ ... }	(Optional)	<p>Additional complex formula system including the abscissa frequency and the result of the leading rational function system:</p> <p>f        abscissa frequency in [Hz] w        abscissa frequency in cycles [s<sup>-1</sup>] TR       result of the transfer function</p> <p>In the case you want to use just this formula but do not need the rational function, enter {b<sub>0</sub>=1} as a dummy before the formula system. This becomes necessary because otherwise the interpreter can not distinguish both function systems.</p>
t <sub>0</sub> =...s	(Optional)	<p>The entry of a time value at t<sub>0</sub>= leads to a multiplication of the other filter function by e<sup>-s·t<sub>0</sub></sup>, i.e. only the phase is rotated and the amplitude remains constant. As an alternative to t<sub>0</sub>=, you may enter a distance z<sub>0</sub>=. If the filter is only to be a delay, you can dispense with the entries of f<sub>0</sub>= and the coefficients { ... }. Lags are produced by t<sub>0</sub>&gt;0 and leads by t<sub>0</sub>&lt;0. The default is t<sub>0</sub>=0. Unit seconds [s].</p>
z <sub>0</sub> =...m	(Optional)	<p>Alternative to t<sub>0</sub>=. Delay by the distance covered by the sound wave. From this AkAbak calculates t<sub>0</sub> according to: t<sub>0</sub> = z<sub>0</sub>/c, where c: velocity of sound. Delays are produced by z<sub>0</sub>&gt;0 and leads by z<sub>0</sub>&lt;0. Unit meter [m].</p>

## Synthesis of filter circuits

### Using active components

Technically speaking, the `Filter` element is similar to the so-called active filter circuits that have buffer stages at the input and output, so that feedback is minimized. The internal resistance of the output stage is virtually zero (e.g. OpAmp or power amplifier) and that of the input stage has a very high impedance. The filter itself is usually implemented by means of capacitors, resistors and gyrators that simulate the inductances. There are a wide number of methods to accomplish this. Relatively small filters usually use the so-called 'Sallen and Key' circuit. This subdivides the polynomials of the numerators and denominators into blocks of 2nd and 1st order polynomials whose product again forms the original transfer function. Each of the terms is accordingly implemented by means of OpAmp or transistor circuits and connected in a cascade. AkAbak provides a function with the aid of which the numerator and denominator polynomials of a `Filter` element can be decomposed into corresponding `Filter` elements whose highest polynomial degree is two (`Filter/Polynomial` → `product`). The entire process also works in the reverse direction, so that several filters can be combined to form one (`Filter/Product` → `Polynomial`).

AkAbak offers a tool to synthesis some of the most commonly used active filter circuits for filter blocks of first and second order (see chapter `Filter/ Active synthesis`).

### Synthesis with passive components

If the filter function is to be implemented by means of purely passive components such as capacitors, resistors, coils and transformers, the problem becomes generally more difficult than with active implementation. The problem here is the networking of the individual oscillation circuits. It is also often difficult to produce components with the required values.

The program provides a synthesis method for calculating the appropriate network from the transfer function. This process was intended for designing loudspeaker cross overs. It generates ladder networks, which are driven by an ideal voltage source at their input side and are loaded by an resistor at their output side. It is capable of taking into account the losses (dissipation) in the coils. The capacitors are regarded as loss-free. The synthesis process processes all pure low-pass, high-pass and band-pass transfer functions (the numerator is not a summation series. Bridge or transformer circuits are synthesized from all-pass functions (see chapter `Filter/LCR synthesis`).

If you wish to build up a passive filter circuit, remember that AkAbak's synthesis method does not permit functions whose denominators consist of a series of summands (the exception is all-pass). The function has to be inherently capable of implementation (Hurwitz criteria).

### Zeros and poles

Using the menu option '`Filter/Polynomial` → `Zeros and Poles`', AkAbak calculates the zeros of numerator and denominator.

## Filter Dialog

*Dialog: Filter/ Filter dialog*

*Ctrl + F*

This is a non-modal input dialog. With this dialog, filter functions can be edited, generated, scaled and transformed. A diagram option displays the transfer function as a graph in the frequency and time domain. The dialog is arranged to allows you to process two different filters simultaneously and analyze them graphically. For the design of cross overs for loudspeakers, the sum of both functions is shown as a graph.

After the dialog is opened (`Filter/Filter dialog`), the parameter are read in for that `Filter` element or `SynthesisInfo` block on which the script cursor is located. Any formulae present are calculated and the coefficients assigned. If you do not intend to evaluate the formulae and you want to display the formula-expressions as an expression in the dialog, then first mark the entire `Filter` element.

The parameter for pure time delay  $t_0=$  and  $z_0=$  and the additionally runtime formula system can not be edited with the filter-dialog. Please, see the help function (F1) for operating details.

## Group 'Transfer 1'

This group contains the data of the first transfer function. The parameter entered here are formatted when the dialog is closed and either copied into the clipboard so that they can be copied into the script or copied to the 'synthesis of polynomial filters with passive elements' dialog, if it is open. (menu: Filter/LCR synthesis).

The input boxes correspond to the parameter listed above, though  $t_0=$  and  $z_0=$  cannot be entered in the dialog. The large input box contains the coefficients  $b_0, b_1, \dots$  and  $a_0, a_1, \dots$  as described above, but without the curly brackets.

## Group 'Transfer 2'

The functions of this group are used to generate, scale and transform a transfer function. It is also displayed in the Bode diagram if you click on the 'Diagram' button at the top right-hand side of the dialog.

The transfer function is displayed as a rational function in the display panel,  $s$  is the complex, normalized frequency variable:  $s = \sigma + j\omega$ . You can either copy in the transfer function of group 1 or have the function generated. The coefficients cannot be edited in this case.

## Sub-dialog 'Standard low-passes Functions'

When you click on the 'Standard Low-passes Functions...', a sub-dialog appears which generates the various filter functions that are particularly of interest for constructing cross overs. When you click on the 'OK' button, the filter coefficients are calculated and assigned to the low-pass transfer function of group 2 of the 'filter' dialog. The 'order' input box accepts the order of the low-pass transfer function to be generated. The maximum value is 30. If it is to be subsequently transformed into a band-pass function, don't forget that this transformation doubles the order. The filter characteristics can be found in the group in the lower part of the dialog. Only a brief description of the most important properties of these curves is given here. You can find further information in the filter literature under the name used here.

### Butterworth

Butterworth is the 'all-round' setting for all filter operations that require a steep filtering characteristic in the amplitude response. The amplitude response of these functions is as flat as possible, i.e. a maximum bandwidth without excessive amplitude rise. The -3dB cut-off frequency is identical to the filter frequency  $f_0$ . The group delay varies in a rippling manner and the step-function response in the time domain overshoots drastically. High-order filters with this function have high pole qualities that may be technically difficult to produce.

### Bessel

The low-passes have a group delay with a maximally flat shape, i.e. maximum group delay without ripple. The step-function response in the time domain does not overshoot. The transition band of the amplitude delay is very rounded. Although the gradient in the attenuation band increases with order, the rounded curve shape in the transition band is retained.

Bessel characteristics are used in particular for delay elements (see below). The zeroes of the Bessel polynomial are extremely far apart, so that any calculations requiring determination of the zeroes only work up to 18th order.

### Linkwitz-Riley

These characteristics are used predominantly in the construction of cross overs. They are similar to Butterworth characteristics, except that the transition band is rounder. The group delay does not ripple so much and the overshoot of the step-function response is not so pronounced as with the Butterworth function. LR filters are

formed by squaring the Butterworth function. The highest quality is therefore equal to that of a Butterworth function of half the order, and the order of LR function is always even numbered. At the filter frequency  $f_0$ , the level has fallen by 6 dB. Literature reference: [Lin]

### Chebyshev

These functions have an extremely steep transition from the pass band to the attenuation band of the amplitude response. Of course, the asymptotic gradient in the attenuation band again depends only on the order of the filter. The pass band has a uniform ripple, the value of which you have to enter in decibels in the 'Ripple' input box, on the right-hand side next to the switch. The higher the level of the ripple, the more angular the transition band. However, this also increases the resonance quality factors, so that in some circumstances it is difficult to construct the filter. The filter is not suitable for time-critical functions, since the group delay ripples to an extreme degree and the step-function response overshoots strongly in the time domain.

### Bu-LR Compromise

This alignments vary between the Linkwitz-Riley and Butterworth characteristics. If a level of 0dB is entered in the associated input box, the Linkwitz-Riley function is obtained. A level of 3dB produces the Butterworth function. This characteristic class is required for the construction of cross overs. The modulus of the sum of an even-order Butterworth low-pass and high-pass has an overshoot of 3dB at the selectivity frequency. The power sum is frequency-independent.

The modulus sum of the LR filter is constant, but instead the power sum at the selectivity frequency falls by half. Bu-LR compromise characteristics here form a compromise. The value in the 'Ripple' input box indicates the overshoot of the modulus sum in decibels. Literature reference: [Bul1]

### Bu-Thomson

This filter curves vary between Butterworth and Bessel characteristics. If the factor in the associated input box is  $M=0$ , a Butterworth function is generated.  $M=1$  generates a Bessel characteristic. With values of  $0 < M < 1$ , so-called Butterworth-Thomson characteristics are obtained.

Butterworth-Thomson characteristics can only be generated up to 18th order, since the zeroes of the Bessel polynomial have to be determined and the method used cannot determine them at higher orders. Literature references: [Wei, Hor]

## Sub-dialog 'Bessel all-pass delay'

The 'Bessel all-pass delay' button leads into a sub-dialog that generates the Bessel all-pass transfer functions. You can enter the delay time 'to' or the delay path 'zo' as well as the order of the transfer function. The maximum order of the transfer function is 30. This dialog is used for building up delay lines for special filter circuits, for example, or for compensating sound travel paths. For the latter, the input box 'zo' is provided, which gives the delay of a sound path:  $t_o = z_o/c$ , where  $c$  is the velocity of sound. If you click on the 'Evaluate' button, the display panel displays the data of the delay line:

## Filters for cross overs

Cross overs for loudspeakers damp the high frequencies in the bass branch and the low frequencies in the treble branch. In two-way models, therefore, a low-pass and high-pass are connected in parallel. In multi-way models, a band-pass is additionally provided. The output of each filter drives a loudspeaker. At the listening point, the sound pressures of the bass and treble branch are summated by modulus and phase. The overall transfer of the loudspeaker is influenced by the transfer functions of the filters, the geometrical positions of the radiators with respect to one another and with respect to the listening point, and on the radiation characteristics of the diaphragms. In designing cross overs, the skill lies in finding filter characteristics for the low-pass and high-pass that, together with the other elements of the transfer chain, optimize the transfer overall. These aspects are summarized in [Pan2].

The filter dialog can be used to investigate functions connected in parallel. In the design, we first assume that the other elements of the transfer chain do not exert an effect. The diagram of the filter dialog therefore shows either the total voltage or the total power of the transfer functions given in group 1 and group 2.

There are several ways of loading the two groups with the filter functions:

- 1) In group 1, enter the coefficients of, for example, the low pass manually and copy them to group 2 using the 'Copy to 2' button, to carry out the low-pass → high-pass transformation there. The transfer function of group 1 thus represents the low pass and group 2 the associated high pass.
- 2) Using the 'Standard low-pass functions' button, generate a low-pass function for group 2 and, using the 'copy to 1' button, transfer it to group 1. The function of group 2 is subsequently transformed into a high pass.

Some filter functions rotate the phase just enough for the total voltage at the transition frequency from low pass to high pass to be extinguished or negatively affected. To avoid this, invert the phase of one of the two filters. In the filter dialog, just the filter of group 1 can be inverted by reversing the sign of the numerator coefficients  $b_0$ ,  $b_1$  ...

The conventional filter functions for loudspeaker cross overs are:

Butterworth (Bu), Linkwitz-Riley (LR), Bu-LR compromise and possibly Bessel.

The characteristics and effects of these functions are described in [Pan2].

## Diagram

Using the 'Diagram' button in the 'Filter' dialog, the transfer functions are displayed in the Bode diagram. The diagram is displayed in the upper part of the diagram dialog. If two transfer functions are present, the sum of the transfer functions is additionally displayed. Changes made to the transfer functions are only represented in the diagram if the 'diagram' button in the filter dialog is pressed. At the bottom right-hand side of the dialog, the type of display is selected. The left-hand switch list relates to the analysis in the frequency range, and the right-hand list to the time range. If two functions are present, the third graph displays the sum of the output parameter of both filters.

### Note

It is very complicated to calculate the responses in the time domain. The procedure used is efficient and fast, but unfortunately only up to a certain time on the abscissa. Beyond this time, the graph may well 'break loose'. This can be seen by the wild up and down fluctuation of the curve. Since the computational errors only occur when the filter has already settled to a steady state, none of the information content of the curve is lost. If necessary the time interval can be reduced using the input boxes in the 'range' group.

That is why the input range of the abscissa input boxes is limited. Time values in the range from  $t = 0 \dots 10/(2\pi \cdot f_0)$  can be input.  $f_0$  is the filter pole frequency given in the main dialog.

## LCR-Synthesis...

*Dialog: Filter/ LCR-Synthesis...*

The non-modal dialog 'Synthesis of polynomial filters with passive elements' opens. From the transfer function, which is provided in the form of a polynomial, this tool synthesizes a passive network. The latter may be inserted directly into the script and simulated.

LCR synthesis is very complicated in principle. The various processes are oriented to the network structure. The synthesis procedures for networks implemented in AkAbak are used especially for the design of loudspeaker cross overs. These ladder networks are very versatile, however, and they can also be used for many other purposes.

Please, see the help function (F1) for operating details of the dialog.

## Source of the transfer function

When the dialog is opened, that `Filter` element or `SynthesisInfo` block on which the script cursor is currently located is read in. With the 'Get from script' button, this procedure is repeated without the dialog being closed.

The transfer function may also originate from the 'Filter' dialog (Menu Filter/Filter dialog). While the synthesis dialog is open, the 'Copy function 1 to synthesis dialog' button appears in the 'Filter' dialog. This button copies the transfer function of group 1 of the 'Filter' dialog into the synthesis dialog. Since neither of the two dialogs is modal, you can change from one to the other.

The transfer function appears as a rational filter function in the display panel in the upper part of the synthesis dialog. The complex frequency variable is  $s = \sigma + j\omega$ .

The maximum order of the transfer function is  $m=30$ ,  $n=30$ . The computational accuracy decreases as the order increases. It is difficult to find a test criteria. The most suitable one is the comparison of the simulation of the transfer function to test the quality of the synthesis, first as `Filter` element then as synthesized network. It is necessary to take into account that the manufacturing accuracy of the components is much lower than the computational accuracy. If, therefore, the program no longer calculates accurately, it is very unlikely that the filter can be constructed using passive elements.

## Feasibility

Not every transfer function can be implemented in a network using purely passive components. Among others, the so-called Hurwitz criterion has to be satisfied, i.e. all pole points of the transfer function are in the left half of the complex plane. The program carries out the feasibility test automatically. The zeroes of the transfer function have to lie either at zero or infinity, or, in the case of the all-pass, symmetrical to the pole points. This results in the following types<sup>7</sup>:

## Low pass, high pass and band pass

It is only possible to synthesize transfer functions whose numerator is not a series, i.e. only one of the numerator coefficients ( $b_0, b_1, \dots$ ) may have a value not equal to zero (with the exception of all-passes, see below). In this context, low passes are all transfer functions whose numerators consist of a constant ( $b_0 \neq 0$ ). High-passes have a numerator whose highest degree coefficient is not equal to zero. All the other are identical to zero. Band passes are produced by the multiplication of a low pass with a high pass. There is also only one numerator coefficient, whose degree depends on the type of band pass.

## All pass

If the transfer function is an all-pass, the zeroes are on the right-hand side of the complex plane and are symmetrical to the poles. If  $H(s)$  is the characteristic polynomial (denominator) of the transfer function, the all pass has the form  $A(s) = H(-s)/H(s)$ , where  $s$ : complex frequency variable. Since, for the implementation the numerator of the transfer function has to be either even or odd, the function is supplemented with  $H(s)$ . This step doubles the order of the all pass, as is manifested by the number of energy-storing elements. A canonical all-pass function always has twice as many coils and capacitors as the order of the transfer function indicates.

---

<sup>7</sup>Incidentally: If a transfer function is to undergo the Hurwitz test, it can be passed to synthesis process. If this is successful, the denominator is sure to be a stable, characteristic polynomial. If the numerator of this function is a summing series, then set it to one so a low-pass function is passed to the test.

## Synthesis

### Low, high and band passes

If the transfer function is of the low pass, high pass or band pass type, a so-called ladder network is synthesized. The term ladder networks is used in the case of networks grounded at one side. The reactances are alternately in series and parallel. The input of the ladder network in this case is connected directly to the voltage source. It is assumed that the voltage source does not have any generator resistance. The load resistance  $R_L$  is located at the network output. The first reactance is always in series with the voltage source (for examples see Fig. 110, Fig. 111 and Fig. 112).

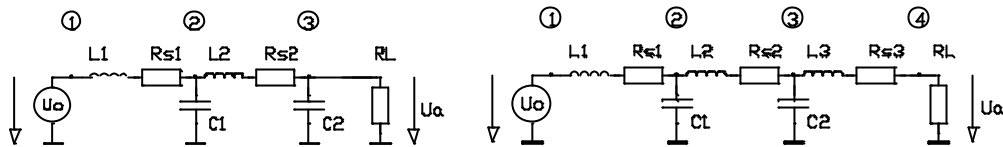


Fig. 110 Low-pass ladder network with dissipative coils. Even order (4), left, and, right, odd order (5)

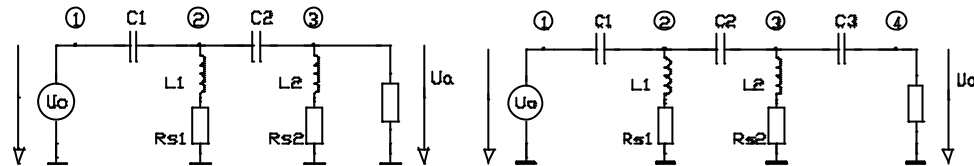


Fig. 111 High-pass ladder network with dissipative coils. Even order (4), left, and, right, odd order (5)

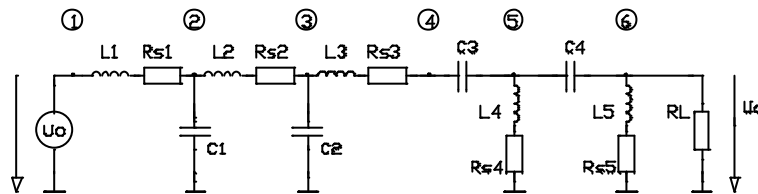


Fig. 112 9th order band-pass ladder network with dissipative coils.

The value of the load resistor is entered in the 'Loading resistor' input box. The filter frequency  $f_0$  is in the 'Filter frequency' box. For the synthesis it is necessary to enter both.

AkAbak's synthesis method takes into account coil losses. The capacitors are always free of loss (dissipation). In the presence of dissipative coils, the synthesis method varies the reactances so that the form of the entered transfer function is as far as possible retained<sup>8</sup>. If, as a result of the series resistors, the network is subject to an insertion loss, this is displayed. Synthesis with dissipative coils is only possible if all coils have the same quality factor. The quality factor can be entered in 'Quality of coils' box. If there is no entry here, a loss-free network is calculated. Empirical values are usually used to estimate the value of the coil quality factor. Typical values are  $Q=10\ldots20$ . The higher the inductance and the thinner and longer the coil wire, the lower the quality of the coil. Core materials have an additional resistance at high frequency. Therefore, enter the quality of the 'worst' coil. So that, in the end effect, each coil has the quality specified, the inductances are supplemented with appropriate series resistors. If  $Q$  is the coil quality at  $f_0$ , the series resistance  $R_s$  of the respective inductance  $L$  is

$$R_s = \frac{2\pi \cdot f_0 \cdot L}{Q} \quad [\text{ohm}]$$

<sup>8</sup>The transmission of high-pass-containing networks is always distorted in the stop band. The stop-band damping decreases. At very low frequencies, a dissipative high-pass behaves as an RC circuit.

The resistor to be connected in series with the coil has a value equal to  $R_s$  minus the loss resistance of the respective coil.

If the network cannot be synthesized with the specified quality, an error report appears at the bottom in the status line of the program. In this case increase the quality factor of the coils by steps. In general: the higher the pole qualities of the transfer function, the higher the value of the coil qualities has to be.

The synthesized network structure is displayed in the left-hand list in the script syntax. The inductances appear as the `Coil` element. The loss resistance that the coil has to have overall is specified as sub-parameter  $R_s$  in the `Coil` element. Therefore, no nodes are drawn between the inductances and resistors in Fig. 110 to Fig. 112. The capacitances use the `Capacitor` element. Since this is regarded as loss-free, only the modulus of  $C$  appears here. The load resistance  $R_L$  is at the end of the list and uses the `Resistor` element.

The synthesis is started by clicking on the 'Network type 1' button. In the display panel above the input boxes, the moduli of the largest inductance and capacitance in the network are displayed in the panel above the input boxes. The modulus of the insertion losses is given at the start of the network list.

The 'Network type 2' button generates an alternative network only for band pass filters. The two alternative networks lead to the same transfer function, however more favorable values for components may exist.

## All pass

All-passes can be synthesized either in symmetrical bridge circuits or as their equivalent transformer circuits. (Fig. 113).

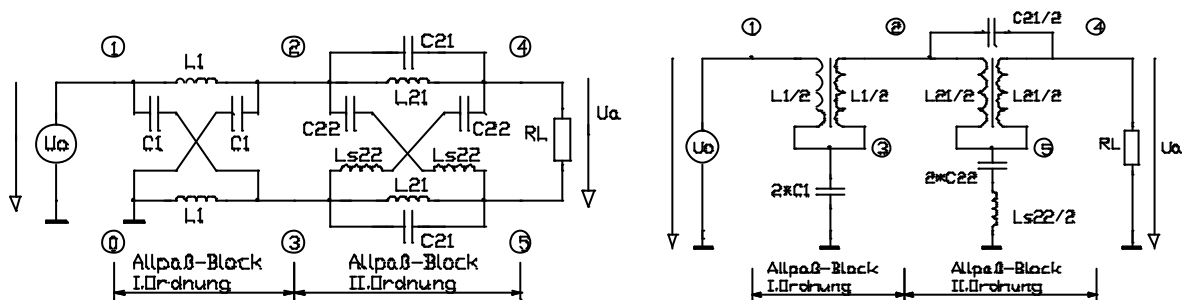


Fig. 113 Example of a 3rd order all-pass bridge circuit. Right, its transformer equivalent.

Passive all-passes are produced by cascading 1st and 2nd order blocks. Thanks to their property of transforming the terminating resistor  $R_L$  at the input of each block, each block is loaded with the resistor  $R_L$ . The example in Fig. 113 shows the network of a 3rd order all pass. A 1st order all-pass block is connected between nodes 1 and 2. Between nodes 2 and 3 is the 2nd order block. The whole is terminated with the load resistance  $R_L$ . If the network is divided at node 2 and the driving-point impedance of block 2 measured, it is found that the impedance is equal to  $R_L$ . The same naturally also applies to the input terminals at nodes 1 and 0.

Component losses cannot be taken into account in all-pass synthesis. The effect of losses, however, should always be tested by the simulation.

## Bridge circuit

The circuit in the upper-hand diagram of Fig. 113 is a lattice circuit (or symmetrical bridge circuit) and is generated with the button 'Lattice network 1', which appears when the transfer function is an all-pass. As Fig. 113 shows, a ground node no longer appears to the right of nodes 1,0. The load resistor  $R_L$  thus has a floating connection.

## Transformer circuit

Without changing the external properties of the network, the bridge circuit can be transformed into an alternative circuit with a transformer (Fig. 113 below). The circuit is generated using the 'Transformer net 2' button. The transformer circuit requires fewer components and lower inductances. The transformer should have a very good coupling and low internal resistances. The two coils are the same except that they are wound in opposite



directions. pot-type cores are usually used for the transformer. The winding data can be calculated using the 'Transformer' input dialog (Menu: E-Net/Transformer...)

In the network list, the individual blocks are separated from one another by commentary lines.

In the 2nd order blocks, the series oscillating circuits are reproduced by the `Capacitor` element, whose sub-parameter  $L_s$  represents the inductance. This makes the entry compacter and gets rid of two nodes.

### Estimation of element-values

All-pass circuits can be very complicated. The following table gives an overview of the complexity.

'Order' means the order of the Bessel all-pass (see above). 'Delay to' means the delay in the chain. The values in the table refer to  $t_0=1\text{ms}$ . The relationship is linear. With the increase in the delay time, the bandwidth is reduced by the same amount. 'Frequency with 1% deviation from to' indicates the bandwidth of the all pass within which it functions as a linear delay line. 'Number of blocks' indicates how many 1st and 2nd order all-pass blocks are required. Although the structure of the individual 2nd order blocks is identical, their component values are not.

Order	Delay to	Frequency with 1% deviation from to	Number of blocks
2	1 ms	180 Hz	II
3	1 ms	380 Hz	I + II
4	1 ms	615 Hz	2 · II
5	1 ms	860 Hz	I + 2·II
6	1 ms	1100 Hz	3·II
7	1 ms	1400 Hz	I + 3·II
8	1 ms	1700 Hz	4·II
9	1 ms	1900 Hz	I + 4·II
10	1 ms	2200 Hz	5·II

### Attenuator

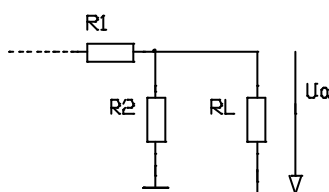


Fig. 114 Attenuator

If the transfer function specifies attenuation, the synthesis process automatically installs two further resistors (Fig. 114). This attenuator takes into account the specified coil losses. The attenuator is always at the network output. It is designed so that, together with the load resistor  $R_L$ , it supplies the same resistance as  $R_L$  to the filter network.

### Compensation network for sealed enclosures

The synthesized network only has the specified transfer function when the load resistance  $R_L$  is to some extent real and constant. If, however,  $R_L$  represents the driving-point impedance of a loudspeaker, the circuit of the filter is networked with that of the driver, its enclosure, or whatever is connected instead of the resistor  $R_L$ . The impedance curve of the network that follows the filter can be linearized by a trick. Connect the network in parallel with its own dual network. The driving point impedance is then real and constant.

AkAbak has a tool for finding the compensation network corresponding to a given impedance curve (see chapter Tools/Impedance Compensation).

## Inserting the network into the script

After the synthesis, the network is copied into the clipboard with the 'Copy and close' button and the dialog is closed. The network can then be inserted into the script.

If the node numbers have to be adapted subsequently, the option in the 'Search/Move Nodes...' menu may help. This function shifts the node numbers of a system from the line on which the script cursor is located.

Since the load resistor  $R_L$  is in most cases replaced by the input resistance of a following network  $R_L$  is not copied by default. If you want to investigate the transfer function of the filter network using the 'Inspect/Voltage...' simulation check the button 'Including  $R_L$ '.

## SynthesisInfo

The copied synthesized network is appended an information paragraph called `SynthesisInfo`. The only function of `SynthesisInfo` is to document the synthesized network.

```
Capacitor  Node=100=101  C=11.526uF
Coil       Node=101=0    L=2.306mH  Rs=0.48ohm
Resistor   Node=101=102  R=5ohm
Resistor   Node=102=0    R=10ohm
Resistor   'RL'  Node=102=0  R=10ohm
SynthesisInfo
  Passive  FirstNode=100  RL=10ohm  QL=30.0
  fo=1kHz  vo=0.5
  {b2=1;
    a2=1;  a1=1.414214;  a0=1;  }
```

## Re-synthesizing

To re-synthesis the filter move the script cursor in the lines of `SynthesisInfo` and issue the 'Search/Current Element...' or press `Ctrl+E`.

## Editing the transfer function

Move the script cursor in the lines of `SynthesisInfo` and issue 'Filter/Filter dialog...' or press `Ctrl+F`. When the Filter-dialog loads the transfer function by this way, the dialog will copy only the transfer function on closing.

# Active Filter Synthesis

*Dialog: Filter/ Active Filter Synthesis*

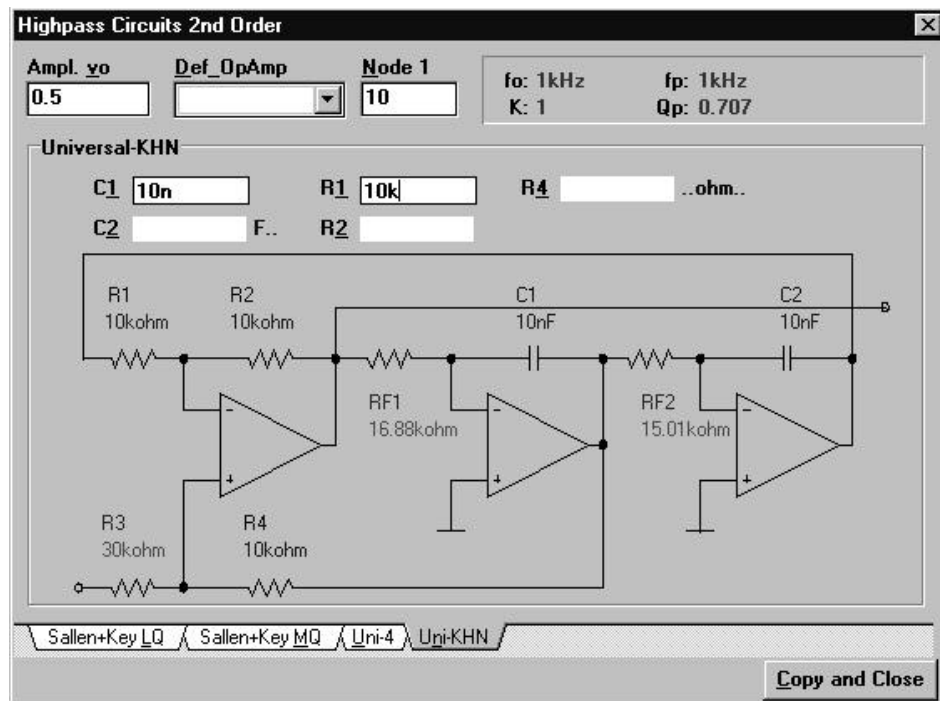
This set of dialogs provides a tool for designing active filter circuits from first and second order filter blocks.

Before an active filter dialog can be opened a `Filter` element of first or second order must be present in the script. Move the cursor in the lines of the `Filter` element and issue `Filter/Active Synthesis`. Instead of a `Filter` element also the `SynthesisInfo` block can be applied as the source. The program interprets the transfer function and opens the appropriate dialog.

In case the transfer function is not of first or second order use the feature 'Filter/Polynomial -> Product' to decompose the transfer function. Transfer functions with two real poles can be realized by either a single second order block or two cascaded first order blocks.

After calculating the components of the active filter press the 'Copy and close' button. The network is formatted ready to be pasted into the script. As with the LCR synthesis a `SynthesisInfo` paragraph is added which saves

the information for further calculations. Potential elements are included at which you can investigate the output level of the active filter with 'Inspect/ Voltage' simulation.



## Dialogs upper part

The dialogs are split into two parts: In the upper part the specification of the Filter element is repeated but in terms of the pole/zero frequency  $f_p/f_z$  and quality factor  $Q_p/Q_z$  of the low pass, high pass, band pass and all pass transfer function.

This notation of the transfer function extracts the constant multiplier  $K$  which is the pass band level. Depending on the circuit this level can not be realized in all cases. Then the form of the transfer function is preserved but with a different level.

Additional amplification ( $vo > 1$ ) or damping ( $vo < 1$ ) factor.  $vo$  is multiplied with the above mentioned factor  $K$ , and again, not all circuits can realize the level  $vo \cdot K$ .

Select from the list of the available, i.e. already defined, Def\_OpAmp definition. If there is no Def\_OpAmp definition present or selected, the program calculates with the default OpAmp-parameter ( $vo = 10^6$ ,  $R_g = 1\text{ohm}$ ).

## Dialogs lower part

The lower part is the component calculator. Usually for a given transfer function there exist a huge variety of possible circuits. Up to now we have implemented the most common active filter circuits.

Click on the tabs to switch to the desired circuit type. There are two circuits with one operational amplifiers. One is for low  $Q$ 's ( $Q < 1.5..2$ ) and the other is for medium  $Q$ 's ( $Q < 5$ ).

The other two circuits are so-called 'state variable filters'. The KHN filter is available in form of an integrated circuit. The latter two circuits are used when high  $Q$ 's at low distortion are specified. Further the amplification, the  $Q$  and the filter frequency can be adjusted independently in contrast to the one-operational amplifier circuits.

## Calculation

The calculation is done when you enter the values in the small number fields. For example enter '10n' in the C2 field and immediately the rest of components is calculated.

Bordered fields must be specified. When not-bordered fields are left blank then the associated parameter is set equal to that of a boarded one.

When no result is displayed the circuit is not realizable. Just change the parameter until the calculation becomes valid.

Note, that you can spin the component values. Enter a start value and then press the cursor keys **up/down**. The spinned values are stepped by the E96 row. When the **Ctrl**-key is pressed additionally the step width is E12 (Beside, this feature works for all number fields in AkAbak).

## Output

The blue values corresponds to the entered data and the red values are calculated.

When the value of 'Possible  $K \cdot v_o$ ' is not equal to the specification  $K \cdot v_o$  then it displays the maximum possible amplification which is possible with the particular circuit.

Some circuits calculate the values to minimize the Gain Sensivity Product (GSP) which is then displayed.

Some circuits implements a voltage divider at the input, for example R11 and R12, when the  $K \cdot v_o$  factor requires a damping. In this case an additional value is displayed which shows the value when no voltage divider is applied, for example  $R1 = R11 \parallel R12$ .

## Close and Copy

Copies the circuit into the clipboard ready to be pasted into the AkAbak script. Appended is a SynthesisInfo block to document and save settings and the transfer function.

Use the Inspect/Voltage and Inspect/Current simulations to analyse the circuit. Move the script cursor into the SynthesisInfo block and issue Search/Current element to modify the active filter circuit when necessary.

For higher order transfer functions the whole procedure must be repeated until all blocks are realized. Usually the blocks have to be modified several times until all currents, amplification factors and so on are optimized.

```
System 'LP1'
Resistor 'R1' Node=1=2 R=15.92kohm
Capacitor 'C1' Node=2=0 C=10nF
Resistor 'R2' Node=4=3 R=10kohm
OpAmp 'Op1' Def='' Node=2=3=4
SynthesisInfo
Active FirstNode=1 Def=''
Sys={10nF 10kohm}
fo=1.0kHz vo=1
{b0=1;
a1=1; a0=1; }
```

## Feedback

*Dialog: Filter/ Feedback*

With this entry a network comprising **Filter** elements and the current network can be described. In the case of complicated filter structures and control circuits, it is helpful to be able to simulate such networks.

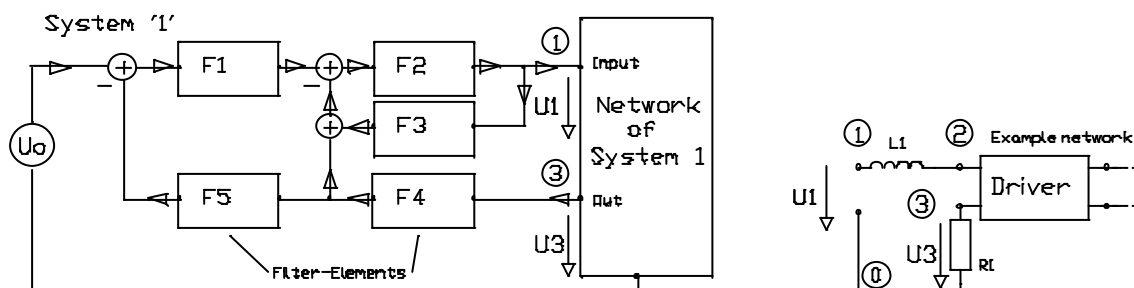


Fig. 115 Left: Example of a filter network, Right: Example of the current network

As the example in Fig. 115 shows, the structure may also include potentials of the current network. It is immaterial here whether the potential corresponds to an electrical voltage, to a force or to an acoustic pressure.

An example of the former is shown in the right-hand side of Fig. 115. Here the current through the voice coil of the loudspeaker (`Driver`) is measured (`U3`) and fed to a control loop (`F4`, `F5`,...).

One example of the feedback of sound pressure is a microphone. The sound pressure across the `Radiator` is taken as potential. The distance from the diaphragm and the transfer characteristic of the microphone are first entered in the following filter.

## Filter name

`Feedback` can only be used within systems and follows the filter elements. Furthermore, all the filters listed have to occur in the feedback equation and be present in the same system.

All filters receive a name of up to 20 characters long, which is repeated in the feedback equation. This name must not be 'U0' or 'U1', since the interpreter could otherwise confuse it with the reserved names of the node potentials U0 and U1. Spaces are permitted within the name.

## Potentials

The `Feedback` formula contains potentials. The basic voltages are U0 and U1. U0 is the input voltage of the `System` and is equal the voltage `Uin` given in the simulation control dialog when no global `Filter` elements are specified at the beginning of the script. Otherwise U0 is weightend by these filters.

If the feedback formula of the `Filter` elements includes potentials of the following network then the network must contain `Potential` elements. It is immaterial whether the potential is an electrical voltage, a force or a sound pressure. The `Feedback` formula relates to the `Potential` element of the same `System` by the name of the `Potential` element. This name can be up to 20 characters long and may contain spaces.

The potential can be a 'floating' potential as for example:  $U34 = U3 - U4$ .

## Generating the equations

There is no method for this. The aim is always to solve the equation for U1, which may crop up on both sides of the equals sign (feedback). U0 always has to be present. U0 is the source voltage. Whether potentials other than U1 are found depends on whether you want to include the potentials of the current network in the structure. Finally, the equation has to be multiplied out to remove the brackets.

### ☒ Example

The feedback equation is generated for the (arbitrary) network shown in Fig. 115. The names of the `Filter` elements are F1, F2, F3, F4 and F5. The source voltage of the network is U0. The input voltage of the current network is U1, U3 is the potential across the resistor R1. In the network a `Potential` element must be implemented:

```
Potential 'U3' Node=3=0
```

$$U1 = (F1 \cdot U0 - (F3 \cdot U1 + F4 \cdot U3) - F1 \cdot F4 \cdot F5 \cdot U3) \cdot F2$$

This multiplies out to the following form, which the script interpreter can evaluate:

$$U1 = F1 \cdot F2 \cdot U0 - F2 \cdot F3 \cdot U1 - F2 \cdot F4 \cdot U3 - F1 \cdot F2 \cdot F4 \cdot F5 \cdot U3$$

The overall transfer function is:

$$U1 = \frac{F1 \cdot F2}{1 + F2 \cdot F3} \cdot U0 - F2 \cdot F4 \cdot \frac{1 + F1 \cdot F5}{1 + F2 \cdot F3} \cdot U3$$

The denominator  $1 + F_2 \cdot F_3$  is thus the characteristic function of the filter network. To this comes the denominator of  $U_3$ , since  $U_3$  is the transfer function of the current network.

Feedback

$$U_1 = \{ F_1 \cdot F_2 \cdot U_0 - F_2 \cdot F_3 \cdot U_1 \\ - F_2 \cdot F_4 \cdot U_3 - F_1 \cdot F_2 \cdot F_4 \cdot F_5 \cdot U_3 \}$$

## Parameter

Feedback

Keyword

$U_1 = \{ \dots \}$

Feedback equation. It describes the structure of the filter network. Its expression is between curly brackets (braces). The identifier is always  $U_1 =$ . The equation is not evaluated by the formula parser but directly by the script interpreter, so that only the -, + and \* operators are permitted. Numerical constants, equations and the minus sign as negator cannot be used here. The interpreter can also not resolve brackets, so that the formula has to be multiplied out. Line breaks and comments are permitted. The variables are the name of the `Filter` elements and of the `Potential` elements of the current network. 'U0' and 'U1' are reserved potential names of the System-input voltage and the network voltage at node 1,0, respectively.

## No current network

If no current network is present,  $U_1$  is simply the output voltage of the filter structure. Current analyses or the like are of course useless in this case. If node potentials are specified here, they are set to zero.

## Stability

The program does not carry out a stability test. But not to worry, your computer will not start to oscillate. If the characteristic function is not a Hurwitz polynomial, the result is simply unusable..

## Polynomial ® Product

Menu: *Filter/ Polynomial → Product*

This command decomposes a given polynomial transfer function  $H(s)$  into the blocks of 1st and 2nd order, as is often required for the synthesis of active filter circuits. The product of the decomposed 1st and 2nd order functions is equal to the transfer function in polynomial form.

$$H(s) = H_1(s) \cdot H_{21}(s) \cdot H_{22}(s) \cdot H_{23}(s) \cdot \dots$$

$H(s)$  transfer function in polynomial form

$$H(s) = \frac{b_0 + b_1 \cdot s + b_2 \cdot s^2 + b_3 \cdot s^3 + \dots}{a_0 + a_1 \cdot s + a_2 \cdot s^2 + a_3 \cdot s^3 + \dots}$$

$H_1(s)$  transfer function of 1st order

$$H_1(s) = \frac{b_0 + b_1 \cdot s}{a_0 + a_1 \cdot s} = K \frac{\omega_n + s}{\omega_d + s}$$

$$\omega_n = b_0/b_1, \omega_d = a_0/a_1, K = b_1/a_1$$

$H_2(s)$  transfer function of 2nd order

$$H_2(s) = \frac{b_0 + b_1 \cdot s + b_2 \cdot s^2}{a_0 + a_1 \cdot s + a_2 \cdot s^2} = K \frac{\omega_n^2 + \omega_n/Q_n \cdot s + s^2}{\omega_d^2 + \omega_d/Q_d \cdot s + s^2}$$

$$\omega_n^2 = b_0/b_2, Q_n = b_2/b_1 \cdot \omega_n$$

$$\omega_d^2 = a_0/a_2, Q_d = a_2/a_1 \cdot \omega_d$$

$$K = b_2/a_2$$

This feature does not have a dialog. It will decompose the transfer function of the `Filter` function in whose line the script cursor is located before the command is called up. If the analysis is successful, the following message is displayed:

'Success! Evaluation error: Numerator=0%, Denominator=0%'

The result is then in the clipboard and can be inserted into the script. 'Evaluation error' is the error of the computation in percent for the numerator and denominator of the transfer function. The error equations are:

$$\text{errN} = 100 \cdot \left| 1 - \frac{\sum_{i=0}^m b z_i^2}{\sum_{i=0}^m b_i^2} \right| \quad \text{errD} = 100 \cdot \left| 1 - \frac{\sum_{i=0}^n a z_i^2}{\sum_{i=0}^n a_i^2} \right|$$

where

errN, errD:	relative error in percent of the numerator or denominator, respectively
m, n:	degree of the numerator or denominator polynomial
b <sub>i</sub> , a <sub>i</sub> :	coefficient of the numerator or denominator polynomial in front of the i-th power of the frequency variable
b <sub>zi</sub> , a <sub>zi</sub> :	coefficient of the numerator or denominator polynomial which has been decomposed and multiplied out again in front of the i-th power of the frequency variable

The denominator coefficients of the polynomial to be decomposed all have to be zero, i.e. poles at zero have to be extracted manually before the analysis. If no denominator polynomial is present, AkAbak sets the constant coefficient to a<sub>0</sub>=1.

The numerator polynomial is arbitrary. The degree may be greater than that of the denominator. In this case zeroes may accumulate at zero in the last term of the analysis, so that the last filter element may in some circumstances have a higher degree than two.

If so-called multiple zeroes occur in the numerator or denominator polynomial, the analysis error may increase drastically, and in some circumstances the analysis may be unusable.

You can now insert the filter system in the script using the 'Edit/Paste' menu command or `Ins`. The result is 2nd order `Filter` elements at most. The filter frequencies `fo` of the individual blocks are equal to the filter frequency of the transfer function in polynomial form. The frequency scaling is implicit in the coefficients.

Real zeroes and poles are analyzed into 1st order filter blocks. If you do not want to do this, you can join the two filters together into a 2nd order block again using the 'Filter/Product → Polynomial' menu option.

The pole frequencies and qualities of the individual blocks are additionally output as a commentary. These parameters are required for many synthesis processes.

<code>f<sub>pN</sub></code> = . . . Hz	pole frequency of the numerator
<code>Q<sub>N</sub></code> = . . .	pole quality factor of the numerator
<code>f<sub>pD</sub></code> = . . . Hz	pole frequency of the denominator
<code>Q<sub>D</sub></code> = . . .	pole quality factor of the denominator

## Pole - zeroes

The pole-zero combination is purely coincidental and if necessary should be checked again before construction of the filter. This does not change the transfer function.

## Constant multiplier

The constant multiplier always appears in the first filter block. In the synthesis of the transfer function, this constant is distributed according to the pass-band gain, the controllability, and the noise behavior of the individual filter stages.

### ☑ Example

A 9th order Chebyshev high-pass filter is decomposed into four 2nd order blocks and one 1st order block.

A point of interest is that a high quality **QD=18.029** occurs in the denominator at a pole frequency of the first 2nd order low-pass filter block of **fpD=985.185 Hz**.

```
Filter
fo=1kHz
{b0=0.006858;
a9=1; a8=0.906118; a7=2.60482; a6=1.812042; a5=2.261823;
a4=1.12861; a3=0.729343; a2=0.223674; a1=0.063868;
a0=0.006858; }
```

```
Filter
fo=1kHz      | fpD=985.185Hz QD=18.029
{b0=0.006858; a2=1; a1=0.054645; a0=0.970589; }
```

```
Filter
fo=1kHz      | fpD=869.595Hz QD=5.527
{b0=1; a2=1; a1=0.157343; a0=0.756195; }
```

```
Filter
fo=1kHz      | fpD=653.99Hz QD=2.713
{b0=1; a2=1; a1=0.241078; a0=0.427703; }
```

```
Filter
fo=1kHz      | fpD=372.609Hz QD=1.26
{b0=1; a2=1; a1=0.295698; a0=0.138838; }
```

```
Filter
fo=1kHz      | fpD=157.355Hz
{b0=1; a1=1; a0=0.157355; }
```

## Product ® Polynomial

*Menu: Filter/ Product → Polynomial*

This command multiplies several transfer function to produce a transfer function in polynomial form. This option is used for network synthesis with passive components (see LCR synthesis), since a multiplied-out transfer function in polynomial form has to be present for this.

The **Filter** elements marked in the script are multiplied out. Each **Filter** element has to be completely marked (see chapter Edit for marking).

The individual **Filter** elements may have different filter frequencies **fo**. After multiplication, a new filter frequency is produced:

$$fo = \sqrt[r]{\prod_{i=1}^r fo_i}$$

where

r: number of filter blocks

foi: the filter frequency of the i-th block



### ☒ Example

A band-pass is formed from a high-pass at  $f_o = 500$  Hz and a low pass at  $f_o = 2$  kHz. When multiplied out, this results in a 4th order band-pass function with the filter frequency  $f_o = 1$  kHz, which can now be implemented as a passive network using the synthesis process.

```
System 'S1'
  Filter
    fo=500Hz
    {b2=1;
      a2=1; a1=1.414214; a0=1; }
  Filter
    fo=2kHz
    {b0=1;
      a2=1; a1=1.414214; a0=1; }

System 'S2'
  Filter
    fo=1kHz
    {b2=4;
      a4=1; a3=3.535535; a2=6.250001;
      a1=3.535535; a0=1; }
```

## Polynomial ® Poles and Zeros

Menu: *Filter/ Polynomial → Poles and Zeros*

This command calculates the poles and zeroes of the transfer function of the `Filter` element in whose lines the script cursor is located.

The result is in the clipboard and can be inserted as a comment.

### ☒ Example

```
Filter
  fo=1kHz
  {b5=0.1; b0=0.112697;
    a5=1; a4=0.899871; a3=1.558227; a2=0.863591;
    a1=0.494225; a0=0.100442; }
```

```
| Zeros of Transfer Function:
| Numerator:
| 0.828591 + j0.602006
| 0.828591 - j0.602006
| -0.316485 + j0.974061
| -0.316485 - j0.974061
| -1.024212
| Denominator:
| -0.278076
| -0.224968 + j0.587785
| -0.224968 - j0.587785
| -85.92992e-3 + j0.951057
| -85.92992e-3 - j0.951057
```

# Simulations

The simulations are carried out in the frequency domain by solving the filter-network and the node-potential matrix of each system-network. Time domain responses can be obtained subsequently by post-processing a spectrum with the help of the FFT (see chapter 'Calc/ Spectrum to time').

All simulations are controlled with the help of the so-called 'Simulation control dialog' which opens whenever you issue one of the commands in the simulation menus 'Sum' or 'Inspect'. The control dialogs are designed to be similar. Some of the entries are actually shared as for example the abscissa range, resolution and the input voltage which can be a rms or a peak value, respectively. The entries are stored in a special header of the script-file. Since the control dialogs are similar the shared control elements are described at the beginning of this chapter.

When you press the 'Ok' button of the control dialog a diagram window is created, the entered data are stored in the script and the simulation is getting started. The graphs are drawn during calculation. You can abort the simulation at any time with **Esc**.

This '**Sum**'-group of simulation displays additive output parameter.


Acoustic Pressure	Acoustic pressure of all radiators at particular test points in the room.
Acoustical Power	Acoustical power of all radiators, incl. directivity factor Q.
Beamwidth	Beamwidth investigation of the directivity vs. frequency.
Directivity Pattern	Directivity pattern of all radiators at specific frequencies.
Sum Voltage	Frequency response of the sum of all filter output voltages.
Driving-Point Impedance	Total driving-point impedance of all systems.
Driving-Point Power	Total input power of all systems.
Driving-Point Current	Total input current to all systems.
Extreme Values	List of extreme values loading the components.
Power Density	Thermal power loading of resistive elements.

This '**Inspect**'-group of simulation displays the individual network parameter.

Voltage	Electrical domain
Current	
Electrical Power	
Force	Mechanical domain
Excursion	
Velocity	
Acceleration	
Pressure	Acoustical domain
Volume Velocity	
Impedance	All domains

## Common Controls of Simulation Control Dialogs

### Bode type

Select the component of the frequency response. All parameters are peak values. Items with '(Phase)' indicate that the phase is available with clicking on  in the diagram or check the box 'Show angle' in the simulation control dialog. When two dimensional curves are displayed the curve of the first component is underlined.

Lp (Phase)	Sound pressure level and phase response. The level relates to the threshold of hearing $p_0=20\mu\text{Pa}$ . $L_p=20\cdot\log(\sqrt{0.5}\cdot p_{\text{peak}}/p_0)$ . For radiation in free space the phase relates to the origin of the baffle coordinate system. i.e. the phase response is made independent from the distance.
Level (Phase)	dB-Level and phase response. The level relates to the constant 1.
Amplitude (Phase)	Modulus and phase response (peak value).
Real	Only real part of response (peak value).
Imaginary	Only imaginary part of response (peak value).
Real and imaginary	Real and imaginary part of response (peak value).
Phase	Unwrapped phase response i.e. a phase which does not rotate within $\pm 180$ degree. The unwrapping works better when you increase the frequency resolution (e.g. 500 points).
Lw	Sound power level. The level is adjusted that it relates to the threshold of hearing $p_0=20\mu\text{Pa}$ .
Power	Sound power in Watts.

### Input voltage Uin

The potential of source that is present at the input of all systems. The potential is the rms or peak value of a sinusoidal signal (check the small box under the Uin-input field if your entered value shall be interpreted as a rms-value).

$1V_p$	$0.707V_{\text{eff}} = \sqrt{0.5}\cdot V_p$	
$4V_p$	$2.828V_{\text{eff}}$	(1W @ 8ohm)
$2.828V_p$	$2V_{\text{eff}}$	(1W @ 4ohm)

The potential might be voltage, force or pressure, respectively, but is entered always with a voltage unit. That means if your network consists only of acoustical components Uin is equivalent  $p_{\text{in}}$ , a pressure.

If you need a current source for a network then use the Gyrator element (see chapter 'General network elements/ Gyrator').

### Abscissa


Since all simulations are performed in the frequency domain the abscissa is the frequency in Hertz. The maximum range which can be specified is: 0.1Hz ... 1000THz. The simulation can be scaled logarithmic or linear. The log-simulation creates dense distribution of points in the lower frequency range and the lin-simulation in the upper frequency range.

'Points' specifies the resolution of the calculations. The maximum is 1024 points. The higher the resolution, the deeper you can 'zoom' into the diagram without re-simulation. However, a high resolution means a long calculation time. The default value is 200 points.

Below the 'Points'-input field the point density is displayed, *log*: number of points within an octave band, *lin*: step size

## Ordinate

The unit of the ordinate is specified by the selected 'Bode type' (see above). You can specify the initial ordinate range in the control dialog, but usually you would change this setting as much as you like subsequently.

When the phase response is the second parameter, like 'Lp (Phase)', it can be hidden by the switch 'Show phase'. You can change this setting subsequently by clicking on  in the diagram.

## Network Lists

The Inspect simulation control dialog has three component lists. Each list belongs to the subsequently displayed graph.

### System, Nodes, Component

The lists display only those System's and elements for which an unique identifier is given, and only those components for which the simulation makes sense. When <clear> is selected this curve is not used.

In the first column the name of the system is given, followed by the node numbers. The last column lists the type and name of the component. When 'Double id:' appears then these elements have the same identifier and can not be selected.

### Operating the lists

Click on the button on the right hand side or press **alt+up/down** to open the list. Select an item with the cursor keys or click on it.

## Distance of the listening point

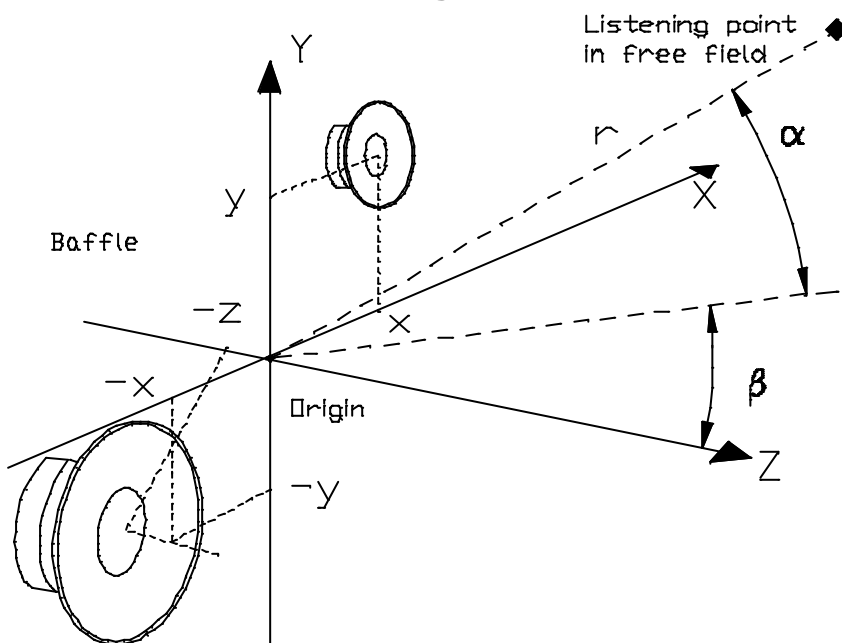


Fig. 116 Baffle coordinate system

$\alpha$ : vertical listening angle,

$\beta$ : horizontal listening angle,

$r$ : distance of listening point

The distance  $r$  between the origin of the baffle coordinate system and the listening point (see Fig. 116) effects the amplitude and phase.

$$p \propto \sum \frac{p_i}{r_i} \cdot e^{-jkr_i}$$

At the listening point the acoustic pressure of the individual radiators is summated according to modulus and phase. The closer the listening point is to the baffle, the more inclined are the acoustic rays from the individual radiators that do not lie at the origin of the baffle co-ordination system.

At very large distances, you must bear in mind that the simulation does not take into account the transmission characteristics of air or other media.

The simulated phase response is re-calculated so that it relates to the origin of the baffle coordinate system.

The lower limit for the distance is  $r=5\text{cm}$ . Use the Inspect-simulation 'Pressure' for a sound pressure simulation directly at the aperture.

## Labels

Labels can control the summation of the sound pressure or sound intensity vectors. For example, when you want to distinguish the total radiation of a vented enclosure and the radiation of the port only. Or, the radiation of a tweeter, the bass channel and the total sum.

Labels must be first specified in the script and can then be selected in the simulation control dialog. System names are also valid labels.

See chapter 'Introduction/ Radiation Environment/ Label' for specification of labels.

All labels specified within the script are listed in the Label-listboxes. Because System-names are also treated as labels they are also listed.

The output is then the radiated sum of all elements with the same label. When a radiator has no label it is not summed, unless <all> is selected. When you have selected a System name then all radiators of this system are summed and displayed.

## Sum/ Acoustic Pressure

*Dialog: Sum/ Acoustic pressure*

*Key: F5*



Simulates the total acoustic pressure of all radiators at up to three different listening points or of three label selections. The transfer function is the impulse response of the total acoustic pressure to the input voltage, including distance, directivity characteristic, diffraction, reflections, networks and filters.

### Listening angles

The listening angle (Fig. 116) indicates the position of a listening point in the free radiation field. At most, three simulations can be carried out from three different listening angles. All other settings, such as distance and the input voltage are the same.

The horizontal angle  $\beta$  lies in the x-z plane of the baffle coordinate system. Positive values rotate the angle towards the positive x axis which is on the left hand side seen from the baffle. The range is limited to  $+180^\circ > \beta > -180^\circ$ .

The vertical angle  $\alpha$  is in the y-z plane. Positive values rotate the angle upwards, negative values downwards. The range is limited to  $+90^\circ > \alpha > -90^\circ$ .

The two switches 'horizontal' and 'vertical' activate the input boxes for three listening angles each, which are assigned to graphs G1 to G3 of the following diagram. If the checkboxes are switched on and the angular values are not equal to zero, the listening points are not in the x-z plane or y-z plane but anywhere in the room. The two angles are their coordinates.

Note, that when no diffraction parameters are specified for a specific radiator there is no radiation behind the infinite baffle.

### Multi-Angles/ Multi-Labels

This switch dispatches the settings to the three possible curves of the diagram.

Multi-Angles simulates a maximum of three listening angles simultaneously. Only one Label can be checked.

Multi-Labels simulates a maximum of three labels simultaneously. Only one listening angle can be checked.

## Mean Window

The mean value (not the intensity) of the sound pressure in a square window around the specified listening point is determined. The angle range specifies the dimensions of the square window in space in terms of angle width. Imagine a sphere centered at the origin of the baffle coordinate system. Then the angle range is the width and high of this patch on the sphere. The maximum is  $180^\circ$  which means  $\pm 90^\circ$  from the listening point which is located in the center of the patch.

The sound pressure values are located equally distributed on the square window, starting with a point on each corner, on each cross point and one in the center. The higher the resolution as more points are included. The distance from one point to the other is reported beside the input box.

## Sum/ Acoustical Power + Directivity Factor

*Dialog: Sum/ Acoustic power*

*Key: F7*

Simulates the acoustical output power, taking into account the filters, the networks and all radiators with their respective directivity characteristics, position, reflection and diffraction.

The acoustical output power is obtained by integrating the acoustic intensity over an envelope surface which contains all the radiators.

When reflectors are present the size of the integrating surface is given by the boundary.

After the simulation is carried out the so called directivity factor  $Q$  is calculated for the first graph and displayed as calculated graph (G6). The directivity factor is the difference of the on-axis level and the sound power level.

## Integration

This type of simulation takes a lot of computation time. Using this input group, you can control the integration process to accelerate the integration process if necessary or increase the accuracy of the calculation.

### 4pi-sr / 2pi-sr

Specifies the range of integration. With '4pi-sr' the integration is performed for the whole sphere, with '2pi-sr' only half of the sphere is taken into account. This switches are disabled when reflection is active since then the boundary walls determine the size of the sphere.

Note, that these switches control only the distribution of the test points. It means not the power radiated into a full or halve space. Calculated is always the total radiated power. But in case of '2pi-sr' this is estimated only from a halve sphere. Thus an omni radiating point source leads to the same power regardless how this switch is set. On the other hand when the radiation takes place from an infinite baffle there is a level difference of 3dB between these two settings since there is no radiation to the rearward side. Since an infinite baffle splits the room, the result reflects the actual power situation, i.e. having double power in the halve room.

### Steps for integration

Select from this list the step size of integration. The first list controls the lower part of the frequency band and the second list controls the upper part. Usually the integration density for the upper part should be higher since here occurs intense interference .

### Cross

You use this setting to accelerate the integration. Only those points lying on a horizontal and vertical equator of the sphere are taken into account. The integration is calculated exactly when the radiator is at the origin of the baffle coordinate system. Although when there are several radiators, an error occurs, this is usually small if the position of the radiator is symmetrical with respect to the integration path. If in doubt, compare the result with the 'area' integration mode (see below). With active reflectors the program switches automatically to the 'area' integration mode.

## Area

This mode of integration scans the entire envelope area. A genuine surface integration is carried out. This setting is necessary if the radiators are positioned asymmetrically on the baffle. If reflectors are active, AkAbak automatically switches to this mode.

## Labels

Controls the summation of the intensity of the radiators. A maximum of three labels can be simulated simultaneously.

# Sum/ Beamwidth

*Dialog: Sum/ Beamwidth*

This simulation calculates the points on the directivity curve where the level is fallen down by a specified value, usually 6dB, with respect to the on-axis level. For each frequency the directivity is calculated and it is searched for this level.

The diagram displays two curves simultaneously. The upper curve is the beamwidth of the left or upper listening point range and the lower curve is the beamwidth of the right or lower range of the listening point (seen from the baffle, left=+x, right=-x, top=+y, bottom=-y).

'Angle range' specifies the ordinate range within the search shall takes place (max.  $180^\circ = +90^\circ$ ).

'Level' specifies the sound pressure level drop relative to the on-axis response (typically -6dB).

'Plane' specifies where the beamwidth shall be calculated.

## Example:

The polar plot of the directivity of Fig. 117 indicates the -6dB points on the main lobe at  $\pm 20.5^\circ$ . To the left is the corresponding beamwidth diagram. At  $f=4\text{kHz}$  the beamwidth is  $\pm 20.5^\circ$ .

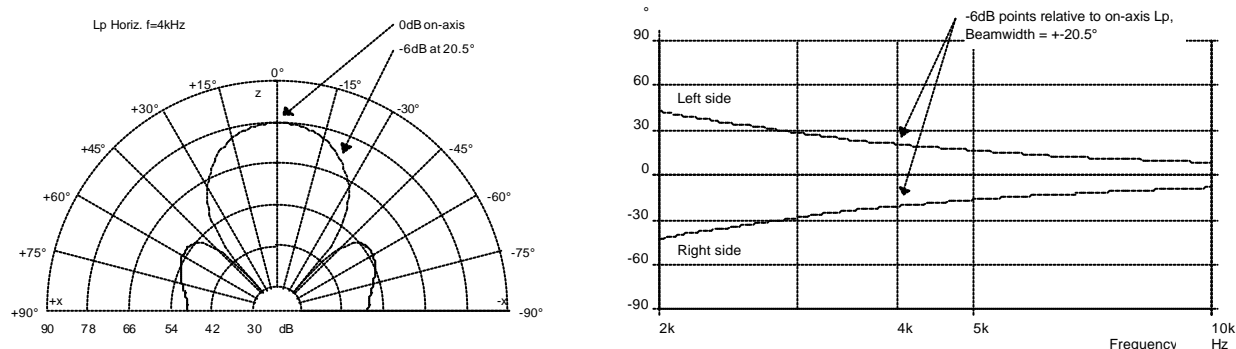


Fig. 117 Polar plot of sound pressure directivity at  $f=4\text{kHz}$  and Beamwidth diagram

# Sum/ Directivity Pattern

*Dialog: Sum/ Directivity, polar*

Key: Shift + F6



*Dialog: Sum/ Directivity, cartesian*

Key: F6

This simulation shows the radiation characteristic of the sound pressure taking into account the filters, the networks, the radiators, reflection and diffraction at a maximum of three frequencies.

The directivity pattern can be displayed in a polar or cartesian plot. In the polar plot only the sound pressure level  $L_p$  can be displayed whereas with a cartesian plot all Bode types are available. This means that also the phase response is plotted vs. the listening angle.

When 'Level normed to 1st graph' checked, the curves G2 and G3, which are usually assigned to different frequencies, are level normed to the on-axis point of curve G1. Otherwise the curves follow the magnitude of the actual response at that frequency.

## Listening angle

The abscissa is the position of the listening point in degrees, either in the horizontal or vertical plane, at the level of the origin of the baffle coordinate system. Note, that the cartesian abscissa starts with negative angles, whereas in the polar plot the negative range is to the right hand side.

In the cartesian plot the angle range can be specified by degree, for example  $300^\circ = \pm 150^\circ$ .

In the polar plot the range can be set to  $180^\circ = \pm 90^\circ$  or  $360^\circ = \pm 180^\circ$ .

For both plots the resolution is fixed to a step size of  $1^\circ$ .

## Frequency parameter

Enter the frequency where the directivity shall be calculated. Each frequency-box is associated with a graph. Use the check boxes to disable a frequency.

## Multi-Frequencies/ Multi - Labels

This switch dispatches the settings to the three possible curves of the diagram. Multi-Frequencies simulates a maximum of three frequencies simultaneously. Only one Label can be checked. Multi-Labels simulates a maximum of three labels simultaneously. Only one frequency can be checked.

# Sum/ Sum Voltage

*Dialog: Sum/ Sum Voltage*

This simulation only refers to the purely electrical network elements and should be regarded as a 'special feature' simulation, originally intended for some aspects of theoretical crossover design. Usually you will obtain better results with the 'Inspect/Voltage' simulation.

Since only the level is displayed here, the input voltage is set fixed to 1V.

The output voltage of a network element is only summated when the keyword 'outport' is included in its script description, for example:

```
Resistor 'R1' Node=345=456 R=123ohm Outport
```

If no network exists in a System, the output voltage of the filter network or the product of the Filter-elements present in a System is added. With the help of this feature it is convenient to simulate the sum of Filter elements (parallel switching). Otherwise, to drive a current-network with a Filter-network which is switched in parallel, the Feedback-formula is used. For example:

```
System 'S1'
  Filter 'F1' ...

System 'S2'
  Filter 'F2' ...
```

is, using this simulation, equivalent to

```
System 'S1'
  Filter 'F1' ...
```



```
Filter    'F2' ...
Feedback  U1={ F1*U0 + F2*U0 }
```

The latter example could also be investigated with the help of the Inspect/Voltage simulation.

If a System contains neither a current network nor a filter network (there is only the term System), the input voltage is added, which, however, may be weighted by a filter cascade in the definition part.

### Power level

Here the modulus square of each voltage is summated and displayed as level in dB. That corresponds to the power in a hypothetical resistor of 1ohm. This type of evaluation is only useful in conjunction with loudspeaker crossovers, since in this case we are dealing with what are, in the end effect, idealized acoustical powers.

## Sum/ Driving Point Parameter

### Impedance

*Dialog: Sum/ Driving Point Impedance*

Total driving-point impedance of all systems with a current network. This simulation accordingly shows loading of the voltage source driving all Systems.

### Power

*Dialog: Sum/ Driving Point Impedance*

Active power which is produced by the voltage source applied to the input of all systems.

### Current

*Dialog: Sum/ Driving Point Impedance*

Current provided by the voltage source which is applied to the input of all systems.

### Note

Please see chapter Def/ Def\_Bassunit for the special treatment of Bassunit's with an optional electrical high pass filter.

## Sum/ Extreme Values

*Dialog: Sum/ Extreme Values*

This simulation does not generate a diagram, but a list. This list contains extreme values that are of particular interest for indicating the loading of the particular components used in the given frequency range and at a given input voltage.

### Abscissa Points

This specifies the resolution of the calculations. The maximum is 1024 points. The higher the resolution, the more exact is the search for the extrema. However, a high resolution means a long calculation time. The default value is 200 points.

## Maximum values of

### Diaphragm excursion of Driver, Bassunit, Speaker

**X<sub>max</sub>** is the maximum amplitude of the diaphragm excursion.

The diaphragms of electroacoustical drivers cannot be moved by any arbitrary amount. In the electrodynamic driver, the diaphragm excursion is provided by, for example, the depth of the magnet gap and the height of the voice-coil development. The half of the difference between these two dimensions is approximately the X<sub>max</sub> of the particular driver. If this excursion is exceeded, intense distortions are produced. The list value can be compared with the loudspeaker data and the maximum input voltage, for example, can be calculated from this.

### Current through Coil and Transformer

**I<sub>max</sub>** is the amplitude maximum in the curve of the current amplitude response. These values are of particular interest if you are dealing with coils having a core and transformers. If the current value is too high, the core material becomes magnetically saturated. The transmission behavior of the inductance then becomes non-linear and the consequence is distortion.

### Voltage across Capacitor

**U<sub>max</sub>** is the amplitude maximum of the voltage-amplitude response. The most important loading limit value of a capacitor is the maximum electrical field strength in the dielectric. A higher voltage damages the dielectric of the capacitor. It is important to compare U<sub>max</sub> with the a.c. specifications of the capacitor. The effective value is often given there, so that you only have to divide U<sub>max</sub> by  $\sqrt{2}$ . However, there is no harm in having a little in reserve.

### Power in Resistor and Driver voice coil

The program searches in the frequency response for the point with the greatest real power. Components are: Electrical resistors and the voice coil resistance of any dynamical driver.

## Sum/ Power Density

*Dialog: Sum/ Power Density*

This simulation calculates the power dissipation in electrical resistors and voice coils based on the power density of special test signals. The handling of this simulation is similar to the Extreme values.

In the Extreme Values simulation, the peak value is sought. If this is exceeded only slightly, distortion, or even damage occurs. In the case of resistors, time is also a factor. Electrical energy that is converted in a resistor or the real part of a coil, such as the voice coil of an electrodynamic loudspeaker, has to be dissipated as heat energy.

A sinusoidal signal will generate a different temperature in the resistor than, for example, a pulsed signal or noise signal. In the first case a voltage with a single frequency is applied continuously across the resistor. In the second case the voltage-signal may consist of a wide spectrum. This raises the question of how much time is available to the resistor to cool down. For example, speech is a signal-mixture which causes a lower temperature in a resistor than - let us say - the spectrum of a distorted electric guitar.

For this reason multiple versions of the power calculation are offered. In all cases the test-function of the input signal is noise with various shapes. Noise has a specific amplitude but a chaotic phase response. See IEC 268-norm for more details.

### Input voltage U<sub>in</sub>

Because with noise as input signal it would make no sense to specify a peak value. In this case U<sub>in</sub> should be the rms-voltage of the noise in the specified frequency range, or in other words the voltage a true rms-voltmeter would measure.

**'Within frequency range'**

Abcissa range in which the simulation and integration is to be carried out. The input limits in these boxes are: 0.1Hz ... 1000THz.

**Points**

This specifies the resolution of the integration. The maximum is 1024 points. The default value is 200 points.

**Noise shapes of driving voltage****Pink Noise**

The level of pink noise drops with frequency by 3dB/Octave. There is more energy at low frequencies than at high frequencies. A band-pass-filters with equal relative bandwidth output the same power at all frequencies when this noise shape is applied. Pink noise is the all-round noise shape for testing.

**White noise**

Here all spectral components in the given frequency range are equally strongly represented.

**IEC 268**

The IEC 268 describes a function with which a pink-noise signal is to be weighted to obtain a power density distribution similar to music. The IEC-paper offers a schematic circuit and a filter function in form of a table.

**RS 426 A (ANSI/EIA)**

Similar to the IEC norm but with a stronger power distribution at higher frequencies.

## Inspect/ Voltage

*Dialog: Inspect/ Voltage*

*Key: F4*

Electrical potential across the specified poles of an electrical network element (peak value). The 'Level (Phase)' type is related to the constant 1V.

**Network**

Lists the available branches and nodes for inspection.

'U1' is the output voltage of the product of all Filter-elements or the Filter-network of this particular System, even when there is no current network present. Thus, use this simulation to investigate the transfer function of abstract filter or filter networks. When there are no Filter-elements and no current-network then U1 is equal the input voltage given in this dialog (it might be weighted by a cascade of globally Filter-elements which are specified in front of the first System). When present, U1 is the driving point voltage of the current network.

**Comments**

In some cases it might be helpful to use the Potential-element to have better access to node-potentials.

## Inspect/ Current

*Dialog: Inspect/ Current*

Electrical current through the specified electrical network element (peak value). If the element is a two pole the flow points from the first to the second node. If the element has more than two poles the simulated flow

direction points always into the element (seen at the first pole of a port). The 'Level (Phase)' type is related to the constant 1A.

### Network

'I1' is the driving point current fed into the network.

## Inspect/ Electrical Power

*Dialog: Inspect/ Electrical Power*

Electrical power in the specified electrical network element. The 'Level (Phase)' type is related to the constant 1W ( $L=10*\log(P/1W)$ ).

### Network

'P1' is the driving point power fed into the network.

## Inspect/ Force

*Dialog: Inspect/ Force*

Mechanical potential across the specified poles (peak value). The 'Level (Phase)' type is related to the constant 1N.

### Network

'U1' is the output voltage of the product of all Filter-elements or the Filter-network of this particular System. When present, U1 is the driving point voltage of the current network.

In the context of the 'Force'-inspection 'U1' is interpreted as driving point force and displayed with the unit Newton [N].

### Driver element

This simulation displays the force between the front and rearward side of the diaphragm.

## Inspect/ Excursion

*Dialog: Inspect/ Excursion*

Key: F8



Diaphragm excursion (peak value). The 'Level (Phase)' type is related to the constant 1m.

### Driver element

The excursion simulation is derived from the flow into the acoustical side of this multi-pole, i.e. the phase is inverted to the flow of those elements which are connected to these poles.

## Inspect/ Velocity

*Dialog: Inspect/ Velocity*

Velocity of diaphragm or sound particle (peak value). If the element is a two pole the flow points from the first to the second node. If the element has more than two poles the simulated flow direction points always into the element (seen at the first pole of a port). The 'Level (Phase)' type is related to the constant 1m/s.

**Network**

'I1' is the driving point flow fed into the network. In the context of the 'Velocity'-inspection 'I1' is interpreted as the mechanical driving point velocity and displayed with the unit [m/s].

**Driver element**

The velocity simulation is derived from the flow into the acoustical side of this multi-pole, i.e. the phase is inverted to the flow of those elements which are connected to these poles.

## Inspect/ Acceleration

*Dialog: Inspect/ Acceleration*

Acceleration of diaphragm or sound particle (peak value). The 'Level (Phase)' type is related to the constant  $1\text{m}^2/\text{s}$ .

**Driver element**

The acceleration simulation is derived from the flow into the acoustical side of this multi-pole, i.e. the phase is inverted to the flow of those elements which are connected to these poles.

## Inspect/ Pressure

*Dialog: Inspect/ Pressure*

*Key: Shift + F8*

Acoustical potential across the specified poles of an acoustic network element (peak value).

The Bode type  $L_p$  (Phase) is also available here which is the sound pressure level in terms of a network potential. This level is not a radiation parameter as calculated in the Sum/Acoustic pressure simulation but it can be used for investigating the  $L_p$  directly at an aperture or within enclosures and to compare the curves with the measurement at these observation points. Note, that the simulation does not take into account the near-field radiation effects of an aperture which may effect the measurement at higher frequencies.  $L_p = 20 \cdot \log(\sqrt{0.5 \cdot p_{\text{peak}}/p_0})$  with  $p_0 = 20\mu\text{Pa}$ .

**Network**

'U1' is the output voltage of the product of all Filter-elements or the Filter-network of this particular System. When present, U1 is the driving point voltage of the current network.

In the context of the 'Pressure'-inspection 'U1' is interpreted as the acoustic driving point pressure and displayed with the unit Pascal [Pa].

**Driver element**

This simulation displays the pressure across the acoustical poles, i.e. between the front and rearward side of the diaphragm.

## Inspect/ Volume Velocity

*Dialog: Inspect/ Volume Velocity*

Acoustical flow through the specified poles of an acoustic network element (peak value). If the element is a two pole the flow points from the first to the second node. If the element has more then two poles the simulated flow direction points always into the element (seen at the first pole of a port). The 'Level (Phase)' type is related to the constant  $1\text{m}^3/\text{s}$ .

**Network**

'I1' is the driving point flow fed into the network. In the context of the 'Vol. Velocity'-inspection 'I1' is interpreted as the acoustical driving point vol. velocity and displayed with the unit [m<sup>3</sup>/s].

**Radiator element**

When the diaphragm has a concave form, the radiation impedance is mapped via an in-built waveguide to model the standing wave pattern inside the cavity. This simulation offers you to inspect the flow and potential on the network side as well as on the radiation side.

The direction of the flow is into the element for the network side (net) and out of the element on the radiation side (rad), i.e. through the free field radiation impedance.

**Driver element**

The vol-velocity simulation calculates the flow into the acoustical side of this multi-pole, i.e. the phase is inverted to the flow of those elements which are connected to these poles.

## Inspect/ Network Impedance

*Dialog: Inspect/ Network impedance*

*Key: F9*



Node-impedance of a network. The 'Level (Phase)' type is related to the constant 1ohm.

**Network**

Lists the names of the System's (here no nodes or elements are displayed in the list).

Enter the node number in the input box 'Node' which impedance is to be analyzed. By default the driving point impedance 'Z1' is simulated (1).

When entering any other (valid) node number the driving point is short cut and the impedance a source would see when feeding the network at the specified node.

**Coupler**

Nodes connected with a Coupler element cannot be inspected.

# Tools

This menu offers special tools for loudspeaker design in form of special dialogs. The way how to operate these dialogs is left to the help documentation which can be accessed by just pressing the **F1** key. In this manual only the range and the principal working is described.

The tools are not part of the main program but out-sourced to a module called AkTools.exe which is launched from the AkAbak menu. AkTools can be called as an alone-standing program, too.

## Create Def\_MeasRadiator File...

*Dialog: Tools/ Create Def\_MeasRadiator File...*

In addition to simulation using mathematical models, there is a component with a transfer function in the form of a table. This special network element is the `MeasRadiator` element, with its associated definition `Def_MeasRadiator`. The two poles of the `MeasRadiator` element are the connection pole of any particular loudspeaker. The output from `MeasRadiator` is the sound radiation.

Before you can use a `MeasRadiator` element, you have to generate the frequency response table. To do this, use a spectrum analyzer to measure the sound pressure and the electrical driving-point impedance of the loudspeaker. It is best to use vector-measurement methods to obtain real and imaginary parts or amplitude and phase, respectively. As a result the `Def_MeasRadiator`-file will be like a photograph-picture of your transducer. There are very good results reported in using this element, at all in connection with cross-over design.

If your measurement device does not allow the vector-measurement, AkAbak uses the Hilbert transformation to generate the all-pass-free real and imaginary parts of the data. All-pass-free means that the frequency-response curve is freed of purely delaying elements. Usually there is no problem with the impedance curve. But great care is to take with the sound pressure curve, because the transmission of loudspeakers are not in all cases allpass-free. The main delay are found to be at high frequencies when standing waves of the diaphragm-cone are involved or with radiation via horns. Further difficulties arise from the Hilbert-transformation. The Hilbert-transformation fits best when the beginning and end of the response curve is equal in level and gradient. Otherwise a ripple and phase distortion is observed. Unfortunately there is no all-round window known to the author which applies here. Please, check your created `MeasRadiator` on correct phase and level response.

After the import of the SPL and impedance curves, the table is stored together with some parameter and commentaries in a file and from then on you can use it as a `MeasRadiator` element in the scripts as often as you like (see also chapters Def/ `Def_MeasRadiator` and chapter Net/ `Transducer/ MeasRadiator`).

It is possible to omit the sound pressure curve and just to generate the `MeasRadiator` file with an impedance curve.

## Procedure

### Measuring instrument

- Measure the sound pressure curve, if possible as level/phase or real/imaginary.
- Measure the distance from the source to the microphone.
- Measure the driving point voltage in Volt (rms or peak value).
- Measure the electrical driving-point impedance, if possible as vector curve.
- Export both measured curves in ASCII-format.

### AkAbak

- Open the menu dialog 'Tools/ Create Def\_MeasRadiator file...'
- Import the sound pressure curve.

- Import the electrical driving-point impedance.
- Enter the reference name in the 'Identification' input box.
- Enter the distance and the input rms or peak voltage during the sound pressure measurement.
- Enter the dimensions of the aperture.
- Enter a short comment if you want.
- Save the Def\_MeasRadiator file.
- If no other Def\_MeasRadiator files are to be generated: close the dialog using Alt + F4 or the Close system menu.

## Measured data

First, the measured data have to meet the import data format requirements as described together with the import procedure in chapter 'Import/ Import Script/ Import Script in Tools'.

### Frequency range

Before the measurement consider in which frequency range to determine the curves. The range should be the same for both measurements, and be equal or larger than the frequency range which is subsequently set in the simulation. If the frequency range is smaller then AkAbak fills out the frequency response. The sound pressure level will converge to zero at very low and very high frequencies. The impedance values stay at the first or last measured point.

The sound pressure is measured using a spectrum analyzer. Note the distance between the loudspeaker and the measuring microphone and also the voltage at the terminals(peak or rms).

### Def\_MeasRadiator Files - Directory

You should save the finished `Def_MeasRadiator` file in the `\AkAbak\MeasRad` directory, which is also set up by the installation program. This has the advantage that when you define the `Def_MeasRadiator` in the script, you do not need to specify the entire path for the file name, but only the file name itself. During the interpretation of the script, AkAbak searches this directory after the `Def_MeasRadiator` file if path name has been specified. You can also modify this default value, see chapter 'File/ Preferences/ Def\_MeasRadiator directory...'

## Sound pressure

### Distance between microphone and diaphragm

If you measure level/phase or real/imaginary then the distance is between the microphone and the 'phase-reference point' during the measurement which becomes also the mounting position of the MeasRadiator element in the script. In other words: the measured vector-SPL-data are assumed to be free of delay due to the distance between radiator and microphone (spectrum-analyzer for electro-acoustics usually automatically remove the phase-lag).

### Remove delay

When the spectrum analyzer does not remove the phase lag of the SPL vector measurement, the SPL data-table is multiplied by  $\exp(j2\pi f/c \cdot r)$  with  $r$ : distance, which causes a phase-rotation to compensate for the delay. In case this formula is too stiff or too simple, it is always possible to weight the measurement curves with your own individual formulae using the ordinate-formula of the import-part, or use the additional runtime-formulae of `Def_MeasRadiator`.

### Note on scalar SPL measurement and Hilbert-Transformation

In the case of scalar measurement and Hilbert-transformation things can become more difficult. The acoustic center of the loudspeaker being measured is the point 'without delay', i.e. without phase lag or phase lead. It is the reference point for positioning in the simulation.



In the mathematical models or with vector measurement, on the other hand, the reference point is not necessarily equal to the acoustic center. In these cases the program recalculates the position of the acoustic center for each frequency or this information is included in the vector measurement curve. Thus you do not need to care about it too much. For the scalar SPL-measurement, on the other hand, you have to estimate the distance to the acoustic center. In the case of conical diaphragms, for example, the center is at the level of the diaphragm suspension. At frequencies well above the directivity frequency, the acoustic center shifts towards the diaphragm interior, i.e. towards the dust cap. To specify the distance, either select the mean value or decide whether the distance should be more accurate for the lower or upper frequency range.

In the case of horn loudspeakers, the acoustic center is always at the level of the diaphragm and not, as in mathematical models, at the level of the horn mouth flange. Of course, when measured vectorially the exact position of the mounting point is at your disposal.

### Radiation environment

We need to examine more closely the radiation environment of the loudspeaker to be measured. The program standardizes the data to the radiation environment of an infinite acoustic baffle at 1 m distance with 1 V peak input voltage. If at the time of the measurement the loudspeaker is in a 'finite' acoustic baffle with respect to the wavelength, the data already include the sound diffraction of the enclosure. Therefore, you should, in this case, not specify the parameter of edge-diffraction `WEdge=` and `HEdge=` for the `MeasRadiator` additionally. The same applies for reflections.

### Cross section of aperture

We normally measure the SPL of the loudspeaker on-axis. The directivity is in this case evaluated by the size and shape of the radiation area, which you can specify in the dialog and also edit subsequently.

The cross section area affects also the absolute amount of the simulated diaphragm excursion, velocity and acceleration, because these mechanical parameters must be recalculated from the acoustic pressure. If the measured loudspeaker is a normal diaphragm loudspeaker, it is a simple matter to specify the diaphragm area. The situation is somewhat different in the case of a measured horn loudspeaker. In this case you specify an area fitting the radiation characteristic of the horn. Since this area is usually much larger than the diaphragm area of the horn driver, incorrect values are indicated for the analysis of the diaphragm excursion, the velocity and the acceleration.

### Listening angle

If you leave this input box empty the program will treat a `MeasRadiator` as a point source - no directivity is calculated. In this case the simulation of the diaphragm excursion, velocity and acceleration is normed to a diaphragm area of  $SD=1\text{m}^2$ .

If you want to analyze the loudspeaker and its directivity in greater detail, you have to set up several `Def_MeasRadiator`'s, each for a specific listening angle. In this case leave the input box of the cross section area blank. `AkAbak` will not calculate any directivity then.



#### Example

Here five `Def_Measradiator` files are created each for a different listening angle  $-60^\circ$ ,  $-30^\circ$ ,  $0^\circ$ ,  $+30^\circ$ ,  $+60^\circ$ . Switch in and out the active one by commenting out the other parameter. (It is planned to implement here some more convenient control.)

```
Def_MeasRadiator 'M1'
|Filename='Twee_H60'
|Filename='Twee_H30'
Filename='Twee_00'
|Filename='TweeH30'
|Filename='TweeH60'
```

## Electrical impedance

The electrical driving-point impedance is measured using a spectrum analyzer. Using the ordinate-formula-option built into the import part, you can easily convert the measured data so that the ordinate shows a resistance value (see the examples given in manual-chapter 'File/Import').

There are usually not so severe possible pitfalls with the Hilbert-transformation as with the SPL curve, which the program automatically applies to scalar measured curves. It is possible to measure for instance the SPL curve as a vector and the impedance curve as a scalar.

## Dyn. Driver Parameter...

*Dialog: Tools/ Dyn. Driver Parameter...*

This tool calculates the parameter of an electrodynamic driver from the measured curve of its electrical impedance by the 'least error squares method'. The object being measured may be a loudspeaker chassis or a loudspeaker in a sealed enclosure. The tool calculates the resonance frequency  $f_s$ , the mechanical and electrical quality  $Q_{ms}$  and  $Q_{es}$  and extended set of voice coil parameter from the measured electrical driving-point impedance of the driver. The measured curve may be vectorial or scalar.

A complete description of a driver motor also requires the conversion factor. The first of these is the conversion factor  $Bl$  and the second the diaphragm area. In practice, the measurement methods often lead to an unknown diaphragm mass  $M_{ms}$  which can be measured relative easily. The force factor  $Bl$  is then determined internally from this parameter. AkAbak provides three methods of determining  $M_{ms}$ . These are combined in a separate dialog: Tools/ $M_{ms}$ ,  $C_{ms}$ ,  $V_{as}$  parameter.

## Procedure

### Measuring instrument

- Measure the electrical driving-point impedance, if possible in vector format.
- Export the measured curve in ASCII - format.
- Carry out one of measurements to determine the diaphragm mass  $M_{ms}$
- Measure the dimensions of the diaphragm

### AkAbak

- Open the dialog 'Tools/ Dyn. Driver Parameter...'
- Import the electrical driving-point impedance.
- Estimate the starting values.
- Correct the value of the voice coil resistance **Re** if necessary.
- First optimize the voice coil inductivity  $L_e$  and the control factor  $ExpoLe$ ,  
Not fixed: **Le**, **ExpoLe**
- Start the optimization.
- When the two curves - especially the imaginary part - are sufficiently matched in the upper frequency range, stop the optimization with **Esc**.
- Now optimize the control factors of the real part of voice coil impedance,  
Not fixed: **fre**, **ExpoRe**
- Start the optimization.
- When the two curves - especially the real part - are sufficiently matched in the upper frequency range, stop the optimization with **Esc**.
- Now optimize the qualities  $Q_{ms}$  and  $Q_{es}$ ,  
Not fixed: **Qms/Qes**
- Start the optimization.

- When the two curves are sufficiently matched in the frequency range around the resonance frequency, stop the optimization with **Esc**.
- Repeat the process for the resonance **fs** if necessary, this is not required in most cases, though.
- Copy the data into the clipboard by pressing 'Copy and close' button.
- Insert data into the AkAbak script using the 'Edit/Paste' menu or **Ins**
- To the parameter, add the identifiers and name of the definitions. Def\_Driver, Def\_TwoCoilsDriver, Def\_Speaker, or Def\_BassUnit.

## Measured data

First, the measured data have to meet the import data format requirements as described together with the import procedure in chapter 'Import/ Import Script/ Import Script in Tools'.

With scalar measurement curves the voice coil control factors of the real part  $f_{re}$  and  $ExpoRe$  can not be determined.

The electrical driving-point impedance is measured using a spectrum analyzer. Using the ordinate-formula-option built into the import part, you can easily convert the measured data so that the ordinate shows a resistance value (see the examples given in chapter 'Import/ Examples').

### Frequency range

Before carrying out the measurement, you should consider in which frequency range to measure the impedance curve. The optimization process is very reliable and calculates the values even from curve parts. It is of course best if the amplitude response is present to an adequate degree and with sufficient resolution in the resonance range and in the effective range of the voice coil impedance.

### Voice coil resistance $R_e$

The voice coil resistance  $R_e$  is estimated from the measured value at the lowest frequency. Often, the measured curve is not yet low enough here, so that the value estimated for  $R_e$  is too high. If in doubt, measure  $R_e$  using an ohmmeter and enter the value in the input box by hand.

## Piezo Driver Parameter

*Dialog: Tools/ Piezo Driver Parameter*

This tool calculates the parameters of a piezo driver from the measured curve of its electrical impedance and from the sound pressure level in the transmission range of the driver.

The object being measured may be a loudspeaker or another piezo driver, for example acceleration pick-ups etc.

## Procedure

### Measuring instrument

- Measure the magnitude of the electrical driving-point impedance.
- Export the measured curve in ASCII format.
- Measure a reference sound-pressure level at a specific distance and at a specific input voltage.

### AkAbak

- Open the dialog menu 'Tools/Piezo Driver Parameters...'.  
– Import the electrical driving-point impedance.

- Set the diagram marker to the lower resonance frequency  $f_s$  and copy the resonance frequency into the 'fs' input box.
- Set the diagram marker to the upper resonance frequency  $f_{s2}$  and copy the upper resonance frequency into the 'fs2' input box.
- Estimate the starting parameters:  $C_p$ ,  $C_1$ ,  $Q_{ms}$ . If the program cannot find the starting values, vary the values of the resonance frequencies.
- First optimize the quartz capacity  $C_p$  at high frequencies, not fixed: **Cp**.
- When the two curves are sufficiently matched in the upper frequency range, stop the optimization with Esc.
- Now optimize the capacitance  $C_1$  at low frequencies, not fixed: **C1**
- When the two curves are sufficiently matched in the lower frequency range, stop the optimization with Esc.
- Now optimize the mechanical quality  $Q_{ms}$ , not fixed: **Qms**
- When the two curves are sufficiently matched, stop the optimization with Esc.
- Repeat the process for the resonances  $f_s$  if necessary, this is not required in most cases, though (the resonance  $f_{s2}$  is only needed for estimating).
- Enter the diaphragm dimensions.
- Compute the conversion constant  $F/U$  in the sub-dialog. Enter the measurement voltage, the distance and the average sound pressure level in the transmission range of the driver. Close the sub-dialog 'Piezoelectric-Conversion Factor  $F/U$ '. The value of  $F/U$  is automatically entered in the input box and the mass  $M_{ms}$  is calculated.
- Copy the data into the clipboard. 'Copy and close' button.
- Insert data into the script using the 'Edit/Paste' menu or **Ins**.
- To the parameters, add the name of the definition.

## Measured data

First, the measured data have to meet the import data format requirements as described together with the import procedure in chapter 'Import/ Import Script/ Import Script in Tools'.

The electrical driving-point impedance is measured using a spectrum analyzer. Using the ordinate-formula-option built into the import part, you can easily convert the measured data so that the ordinate shows a resistance value (see the examples given in chapter 'Import/ Examples'). For piezo drivers, the constant voltage method is usually a good choice, since the impedance is generally high enough to allow you to insert a shunt resistance.

### Frequency range

Before carrying out the measurement, you should consider in which frequency range to measure the impedance curve. The optimization process is very reliable and calculates the values itself from curve parts. It is of course best if the resonance range comes to lie approximately in the center of the measurement range. There should be approximately two octaves above and below the resonances.

### Driver constant $F/U$

You can measure the average sound pressure level for calculating the driver constants  $F/U$  by using a sound level meter or by reading the level from the curve.

A tricky way to overcome radiation problems in the upper frequency range of many piezo drivers is to measure the  $L_p$  directly at the main (lower) resonance frequency  $f_s$  and weight the  $L_p$  by the calculated  $Q_{ms}$ . With a high quality factor,  $f_s$  is approximately identical with the frequency at the  $L_p$ -peak and should be clearly visible. Then add to the measured  $L_p(f_s)$ :

$$L_{p_{ref}} = L_p(f_s) - 20 \cdot \log(Q_{ms})$$

Here it is assumed that the driver is mounted in a large baffle.

(Note, that this trick can also applied to dyn. drivers but instead using Qms the total Q should be inserted in the above formula).

## Mms/ Cms/ Vas Calculation

*Dialog: Tools/ Mms, Cms, Vas...*

In determining the parameters of an electrodynamic driver, not all factors can be determined solely from the impedance curve (see chapter Tools/ Dyn. Driver Parameter). At least one of the parameters must be determined from the output of the driver.

Instead of the conversion factor Bl of the driver we determine the mass Mms of the diaphragm assembly or Vas, the equivalent air volume to the compliance of the diaphragm suspension Cms. The parameter Vas, Cms and Mms can be transformed with the help of the diaphragm area and the resonance frequency:

$$f_s = \frac{1}{2\pi \cdot \sqrt{M_{ms} \cdot C_{ms}}} \text{ and } V_{as} = C_{ms} \cdot S_D^2 \cdot \rho \cdot c^2$$

with

SD: Area of diaphragm  
 fs: Resonance frequency  
 Cms: Compliance of the diaphragm suspension  
 Vas: Equivalent air volume to Cms  
 ρ: Density of air, c: Velocity of sound

Vas is purely a design factor. Vas describes a theoretical air volume whose compliance is proportional to the value of Cms. The parameter Vas originated in the design of bass speaker enclosures. There, together with the volume of the enclosure it forms a proportional factor which is simple to compute.

Please note, that the parameter Vas is highly dependent on other parameter and values of the surrounding medium. It is recommended to preferably use Mms or Cms in the specification. For instance if you enter Vas and modify the diaphragm area, AkAbak would calculate a fault spring force of the suspension.

The medium parameter c and ρ are fixed at this place and will not be modified when you change the system parameter (c=343.3m/s, ρ=1.187kg/m<sup>3</sup>, air @ 20°, see chapter "File/ Preferences/ Physical System Constants").

Using this tool, you can determine Mms and Vas, respectively, by three different measurement methods.

Application of an added mass

Installation in a sealed enclosure

Sound-level measurement

## Application of an added mass

This method is most accurate and most practical. It can be used both for large bass speakers and small tweeters.

### Tools

- 1) Oil-free 'Plasticine' modeling compound.
- 2) Precision assay balance, range up to 50g...100g with a resolution of 0.1g.
- 3) Measuring instrument for determining the resonance frequency of the driver.

### Measuring the resonance $f_s$ (without added mass)

First of all measure the resonance frequency  $f_s$  of the loudspeaker. You can read this from the magnitude of the impedance curve or with the aid of a sine-wave generator and an oscilloscope using the constant voltage or constant current measurement method.

### Attaching the added mass

Then stick a piece of the Plasticine modeling compound onto the loudspeaker diaphragm. The amount depends on the size or mass of the diaphragm. The mass of the piece should be about 1/10 to 1/5 of the diaphragm mass. Attach the modeling compound so that it can oscillate together with the diaphragm. It is best to attach the piece close to the voice coil fastening, so that the coupling is as rigid as possible.

### Measuring the resonance $f_s'$ (with added mass)

In the next stage, measure the resonance frequency again. This new resonance  $f_s'$  is now at a lower frequency, since the overall mass has increased.

### Weighing the added mass $m$

Finally, carefully remove the Plasticine piece and determine its weight  $m$ . The mass of the vibrating assembly of the loudspeaker is then

$$M_{ms} = \frac{m}{(f_s / f_s')^2 - 1}$$

where

$m$ :	mass of the attached piece in [kg]
$f_s$ :	resonance frequency without added mass in [Hz]
$f_s'$ :	resonance frequency with added mass in [Hz]

## Installation in a sealed enclosure

This method uses the increase in the resonance frequency of the driver when it is installed in a sealed enclosure.

$$V_{as} = V_b \cdot \left[ \left( \frac{f_c}{f_s \cdot m_b} \right)^2 - 1 \right]$$

where

$V_b$ :	enclosure volume in [m <sup>3</sup> ]
$f_c$ :	resonance frequency of the installed loudspeaker in [Hz]
$f_s$ :	resonance frequency of the free-radiating loudspeaker in [Hz]
$m_b$ :	air-load factor $m_b = 0.95$ .

### Prerequisite

The enclosure dimensions have to be much smaller than the wavelength at the resonance frequency  $f_c$  of the installed driver.

### Tools

- 1) Completely airtight and robust loudspeaker enclosure.
- 2) Measuring instrument for determining the resonance frequency of the driver.

### Measuring the resonance $f_s$ (free-radiating)

First measure the resonance frequency  $f_s$  of the free-radiating loudspeaker. You can read this from the magnitude of the impedance curve or determine it with the aid of a sine-wave generator and an oscilloscope using the constant voltage or constant current measuring method.

### Installation in the enclosure

Make sure you install the speaker in an airtight manner, otherwise a reflex enclosure results. Do not fill the volume with insulating material. The driver should radiate into the volume with the front side of the diaphragm.

### Measuring the resonance $f_c$ (in the installed state)

In the next step measure the resonance frequency again. The new resonance  $f_c$  is now at higher frequencies, since the overall spring stiffness has increased.

Installing the driver in the enclosure also increases the air load, and therefore the mass of the oscillating system. Unfortunately it is not so easy to determine this factor accurately. Before you enter  $f_c$ , increase the measured value by approx. 5%.

### Determining the enclosure volume $V_b$

The enclosure volume is determined by its geometrical dimensions. The volume is decreased or increased by a small amount after installation of the driver.

## Sound-level measurement (Reference $L_p$ )

Determining  $M_{ms}$  or  $V_{as}$  from the average sound-pressure level in the transmission band of the loudspeaker.

### Prerequisite

The presence of a frequency band in which the acoustic pressure level curve of the loudspeaker is linear. The response must not be disturbed by natural oscillations of the diaphragm, by refraction at edges, or by reflections. In practice it is not always easy to meet these prerequisites. To do this you have to install the driver in a sufficiently large baffle.

You also need the values of the driver parameters  $f_s$ ,  $dD$ ,  $R_e$  and  $Q_{es}$ . (see Dyn. Driver Parameter).

### Tools

- 1) A robust and sufficiently large baffle
- 2) A spectrum analyzer for measuring the sound level curve

### Installation of the driver in the baffle

Install the driver in a flat and robust baffle. The distance to the nearest edge is at least two wavelengths at the frequency at which you are determining the level.

### Measuring the sound level

To accomplish this, record the sound level curve along the axis using a spectrum analyzer. The acoustic pressure curve of a loudspeaker has numerous peaks and inflections. Either integrate it optically or use the smoothing function of the spectrum analyzer. Alternatively you can also measure the sound level using third or octave bands.

The transmission band of the driver in this case is the frequency range between the resonance frequency and directivity frequency ( $f_D = c/(\pi \cdot dD)$ ) of the diaphragm. You should therefore not approach the resonance frequency too closely when determining level, since the curve is altered by the resonances here.

Measure the distance  $r$  from the microphone to the acoustic center of the radiator. In the transmission mode, the acoustic center of a diaphragm loudspeaker is at the level of the bead of the suspension.

The measured rms or peak voltage  $U_o$  is the voltage of the constant voltage source.

### Trick:

It is in some cases easier to measure the  $L_p(f_s)$  exactly at the resonance frequency  $f_s$  and then subtract the 'level' of the total quality factor  $Q_{ts}$ :

$$L_{p_{ref}} = L_p(f_s) - 20 \cdot \log(Q_{ts})$$

## Impedance Compensation

This function determines the components  $R_L$ ,  $C_e$ ,  $R_e$ ,  $L_m$ ,  $R_m$  and  $C_m$  from the circuit in Fig. 1, which compensates the impedance curve of one or more electrodynamic drivers in a sealed enclosure.

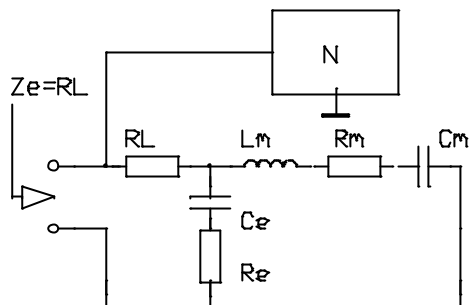


Fig. 1 Impedance compensation for network N

### Why?

Impedance compensation is sometimes required for the Synthesis of passive cross over networks. The synthesis method assumes a real, constant terminal resistance  $R_L$ . In cross over networks, this resistance  $R_L$  is replaced by the driving point impedance of one or more drivers with the following acoustic structure. Since the curve of this impedance in the frequency range of the poles of the cross over network is usually anything but constant, it has to be compensated. The aim is to generate the most constant possible impedance curve for the real part with a certain resistance level. The imaginary part should tend to zero.

### Sealed and reflex enclosures, Horns

In practice you will almost only require compensation of the resonance parts of the driver for loudspeakers in sealed enclosures. Bass reflex enclosures are usually controlled by active filters. This tool therefore only compensates impedance curves of structures that resemble a driver in a sealed enclosure. By 'resemble' we mean that the algorithms of this tool only require the curves of the real and imaginary part of an impedance to determine the compensation network. It is immaterial here which structures have generated these curves. For example, one of the 'impedance humps' of a reflex speaker or horn structure can also easily be compensated.

### Compensation network

It is parallel to the circuit to be compensated and therefore also has the same input nodes.

The resistance  $R_L$  is equal to the target resistance that the entire circuit ought to have at the input.

The elements  $C_e$  and  $R_e$  compensate the increase in impedance of the voice coil of the driver at high frequencies.  $R_e$  has nothing to do with the resistance of the voice coil.  $R_e$  helps in approaching the non-linear reactance of the voice coil. In many cases its value approaches zero ohms in the optimization process.

The inductivity  $L_m$ , the resistance  $R_m$  and the capacitance  $C_m$  compensate the selective rise of the impedance curve with driver resonance.

$L_m$ ,  $C_m$ ,  $R_m$  form the elements of a serial oscillating circuit, parallel to the serial network of  $C_e$  and  $R_e$ . This two branches are in series to a resistance called  $R_L$ .

### Optimization process

The optimization process calculates the square error between the impedance curve of the circuit to be compensated and that of the compensation network and attempts to reduce this gap.

The diagram shows the amplitude response of the impedance of the network to be compensated and the impedance variation of the real and imaginary parts of the network to be compensated.

During the optimization, the curve of the real part should be smoothed and the level correspond to the resistance  $R_L$ . The imaginary part should approach zero at all frequencies.



The optimization process is usually very reliable. Often it does no harm to speed up the process by enter new starting values.

By the way, optimization only takes place in the diagram-abcissa range currently set.

### Source of the impedance curve

The amplitude/phase or real/imaginary parts of the impedance part of the network to be compensated originated from a simulation of the network driving point impedance previously carried out (Inspect/ Network impedance) or of an imported curve of vector measurement.

Set the mode to 'amplitude+phase' or to 'real+imaginary' in the control dialog of this simulation and place the frequency range (abcissa) so that you can see the entire frequency band in which the compensation is to be carried out

When this simulation has been carried out and the diagram activated, you can read in the curves when opening this tool or using the 'Get impedance curves' button. The curves are drawn again in the dialog.

### Output of the compensation network

Connect the compensation network in parallel with the network to be compensated with the same input nodes. Ce and Re are combined in a capacitor element. Re is the serial loss resistance Rs of this element. Proceed in a similar manner for elements Lm, Rm and Cm. Cm is the main value, Rm and Lm are the loss values of the capacitor elements.

☒ Example:

```
| Impedance compensation
Resistor   Node=1=2   R=5.9ohm
Capacitor  Node=2=0   C=4.383uF   Rs=2.722ohm
Capacitor  Node=2=0   C=0.31mF   Rs=2.911ohm   Ls=11.083mH
Bassunit   'B1'      Def='Bu1'   Node=1=0
...
```

## Procedure

### AkAbak

- Prepare the script so that driver is connected at free entrance nodes.
- Simulate the network impedance in the 'Amplitude (Phase)' or 'Real and Imaginary' mode (Inspect/ Network Impedance..... menu).

## Tools/ Impedance Compensation

- Open the 'Impedance Compensation' dialog. This reads in the impedance curves of the active diagram.
- Determine the resonance frequency fs using the diagram marker and enter the resonance in the input box fs.
- If necessary, correct the target resistance RL, which the compensation network is to form in parallel with the circuit to be compensated.
- Choose the compensation range: 'At resonance' or 'At high frequency' or both.
- Estimate the starting values.
- Choose the optimization factors by fixing the others.
- Start the optimization algorithm.
- When the optimization stagnates, terminate using **Esc**.
- Repeat the optimization possibly using different optimization parameters.

- 'Close and copy' - close the dialog.
- Insert the network in the script.

# Appendix

## Installation

The installation program 'Install.exe' runs completely automatically. It creates a directory structure, copies the files onto the hard disk, decompresses them if necessary and installs the start icons in the Windows start menu. You can re-install or update at any time into the same directory structure. File created by your own are save, but all others will be over-written. The desktop files containing the current diagrams are erased. Save them temporarily into another folder.

- Windows™ 3.1, 95, NT operating system and an Intel® 80386 type processor (or higher or compatible).
- A numerical coprocessor (Intel® 80387) considerably speeds up calculations, but is not absolutely necessary. The program runs best on a computer with an Intel® 80486DX processor (or higher or compatible), in which the numerical coprocessor is already integrated.
- The video configuration should conform to the VGA (or higher) standard. The outputs are in color. Gray-scale-type monitors (LCD etc.) may not reproduce all the details satisfactorily. You can adjust the colors etc. in the 'File/ Preferences/ CRT Diagram Style' menu.
- The print out is on any A4 printer available in Windows™ 3.1 or 95.
- For the installation you need a 3½" high-density (HD) disk drive.

## Starting the installation program

Insert diskette 1 into the disk drive. There are two ways of starting the installation program:

### Using the Windows file manager

Open the file manager and change to the disk drive. Using the mouse, double click on `Install.exe` on the right-hand side of the list.

### Using the Windows desktop

In the 'File/ Execute...' menu enter the following: `A:\Install.exe`. You can also use the 'browse' button.

Start the installation program using the OK button.

## Select the target drive and directory

After you have started `Install.exe`, the dialog 'Installation of AkAbak' appears. Decide which drive you want to install AkAbak on. It does not take up more than 5 megabytes of space on your hard disk. In the input box, enter the drive and directory name in which you want to install the program. If the directory does not exist, one will be created. The format must conform to the DOS operating system: `Drive:\Directory`, for example: `C:\AkAbak`

## Starting the installation

Click on the 'Installation' button. The installation starts. At some stage you will be prompted to insert the second diskette. When you have exchanged the diskette, press on the OK button and the installation continues. Finally, two start icons are installed in the program manager. At first, the start items are located in their own program group. You can of course move or copy them into a different group and then delete the AkAbak program group. The first icon starts the AkAbak program and the second starts the Abakus equation calculator.

**Directory structure.**

After the installation, you have the following directory structure on your hard disk.

...	AkAbak	
	\Formula	files with equations of the Abakus program
	\Import	files for importing
	\MeasRad	Def_MeasRadiator files
	\Scripts	
	\Examples	example scripts
	\Project1	scripts of your first project
	\zProgram	program files

The `Scripts` sub-directory is located in the `Project1` directory. It is empty and you should regard it as a suggestion.. You can, of course, change the name if you want. With each new large project it is useful to install a new directory under `Scripts`. Use the Windows file manager for this. This file structure is not absolutely essential, but it has been found to be practical.

That concludes the installation.

**Terminating the installation**

The left-hand button 'terminate' immediately terminates the installation process.

If you accidentally click on this button using the mouse or press Esc, you just have to start the installation again from the beginning. Close the dialog by clicking on the 'terminate' button again or press Esc and insert diskette 1 in the drive again. You do not need to delete anything on the hard disk.

**Removing the AkAbak program**

All the files forming part of AkAbak are located in the installed directories. There are no hidden entries in the Windows directories or in the initialization file `win.ini`. Open the Windows file manager and activate the main AkAbak directory. Then press the Del key.

## Technical Program Information

The following information indicates the program's internal limits. There is no guarantee that these values will apply in all possible combinations. These limits are also not linked to the computational accuracy. The latter depends not only on the number of elements, but also on their properties.

### Elements

max. number of systems	10
max. number of radiator elements overall	500
max. number of driver definitions	500
max. number of nodes in a current network per system	56
max. number of network elements per system	500
max. number of filter element per system	60
max. degree of the polynomials of the element filter	30

### Character input

max. number of characters per script window	32000
max. number of characters in names	20
max. number of characters of an equation in the script	4000
max. number of characters in a feedback equation	1000

### Numerical input

numerical range of the parameters	$ 10^{-20} \dots 10^{20} $
numerical range of the filter coefficients	$ 10^{-100} \dots 10^{100} $

### Node numbers

Range of specification	0...32000
------------------------	-----------

### Internal calculation factors

numerical range	$ 10^{-4000} \dots 10^{4000} $
mantissa	19 digits
max. number of finite elements for horn radiation	400
FFT-points	1024
limiting value for $\log(0)$	-200dB

### Physical constants

velocity of sound (variable)	$c = 343.3 \text{ m/s } (20^\circ\text{C})$
density of air (variable)	$\rho = 1.1871 \text{ kg/m}^3 (20^\circ\text{C})$
dynamic viscosity of air	$\eta = 18.2 \mu\text{Pa s } (20^\circ\text{C})$
sound level at the threshold of audibility	$p_0 = 20 \mu\text{Pa}$

### Diagram

max. number of graphs per diagram	6
max. number of points per graph (resolution 100%)	1024

### Import

max. number of importable points	64,000
max. number of graph points	1024
minimum number of purely imported graph points	2
minimum number of graph points, tools and import	30

## Literature References

- Ber Beranek, L. L.: 'Acoustics'; American Institute of Physics, Inc., New York, USA, 1986 (335 East 45 Street, New York, New York 10017, USA)
- Bul1 Bullock, R.M.: 'Satisfying Loudspeaker Crossover Constrains with Conventional Networks - Old and New Design'; Jul/Aug, JAES<sup>9</sup>, 1983
- Col Colloms, Martin: 'High Performance Loudspeaker', John Willey & Sons, NY 1997
- Dav Davis, Don+Carolyn: 'Sound System Engineering'; H. W. Sams &Co, Macmillan, 1975
- Fli Fliege, N.: 'Lineare Schaltungen mit Operationsverstärkern', Berlin, Springer, 1979
- Ged1 Geddes, Earl R.: 'Acoustic Waveguide Theory Revisited'; Vol.41, No.6, Jun, JAES, 1993
- Hor Horn, P., Moschytz, G. S.: 'Handbuch zum Entwurf aktiver Filter', München, Oldenbourg, 1983
- Lin Linkwitz, S.H.: 'Active Crossover Networks for Noncoincident Drivers'; Jan/Feb, JAES, 1976
- Mor1 Morse, P., Ingard, K.: 'Theoretical Acoustics', Princeton, New Jersey, USA, Princeton University Press, 1986 (41 William Street, Princeton, New Jersey 08540, USA)
- Mor2 Morse, P.: 'Vibration and Sound', New York, McGraw-Hill, Inc., 1948
- Pan1 Panzer, J.: 'Konstruktion von Baßlautsprechergehäusen', 5.Auflg., München, Franzis, 1993
- Pan2 Panzer, J.: 'Frequenzweichen für Lautsprecher', 3.Auflg., München, Franzis, 1991
- Pan3 Kreutz, J.; Panzer, J.: 'Derivation of the Quasi-Butterworth 5 Alignments'; Vol.42, Nr. 5, JAES 1994
- Pan4 Panzer, J.; Campbell, R.: 'Multiple Driver Modeling with a Modern Lumped Element Simulation Program', AES Preprint 4441 (102nd Convention Munich)
- Pan5 Panzer, J.; Harris, N.: 'Distributed Mode Loudspeaker Simulation Model', AES Preprint 4739 (104th Convention Amsterdam)
- Sal Salmon, V.: 'A New Family of Horns', Journal of the Acoustical Society of America, Vol. 17, Jan. 1946
- Sma1 Small, R.H.: 'Closed Box Loudspeaker Systems'  
a) 'Part I: Analysis'; Vol. 20, Nr 10, JAES 1972  
b) 'Part II: Synthesis'; Vol. 20, Nr 1, JAES 1973
- Sma2 Small, R.H.: 'Vented Box Loudspeaker Systems'  
a) 'Part I: Small Signal Analysis'; Vol. 21, Nr 5, JAES 1973  
b) 'Part II: Large Signal Analysis'; Vol. 21, Nr 6, JAES 1973  
c) 'Part III: Synthesis'; Vol. 21, Nr 7, JAES 1973  
d) 'Part IV: Appendices'; Vol. 21, Nr 8, JAES 1973
- Thi Thiele, A.N.: 'Loudspeakers in Vented-Boxes'  
a) 'Part I'; Vol. 19, Nr. 5, JAES 1971  
b) 'Part II'; Vol. 19, Nr. 6, JAES 1971
- Wat Waterhouse, R.: 'Output of a Sound Source in a Reverberation Chamber and Other Reflecting Environments'; Journal of the Acoustical Society of America, Vol. 30, Nr.1, Jan. 1958
- Wei Weinberg, L.: 'Network Analysis and Synthesis'; New York, McGraw-Hill, Inc., 1962
- Zwi Zwicker E., Zollner, M.: 'Elektroakustik', Berlin: Springer 1984

---

<sup>9</sup> Journal of the Audio Engineering Society

# Software License Agreement

IMPORTANT INFORMATION: This Agreement is a legal agreement between Joerg Panzer, Mindelheim, Germany hereinafter referred to as the Author and the user of the AkAbak software supplied with this Agreement by installing, copying, or otherwise using the AkAbak software, the user agrees to be bound by the terms of this Agreement.

## 1. GRANT OF LICENCE

This Agreement licenses a software package for the simulation of electro-acoustic networks, particularly loudspeakers, by electronic data processing means, which is called AkAbak and is hereinafter designated as such. AkAbak is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties.

Subject to payment of the appropriate fee and on the following terms and conditions, The Author grants to the user, and the user accepts, a non-exclusive personal and non-transferable licence to use AkAbak on a single computer workstation. The user only acquires ownership of the physical data medium on which AkAbak is recorded and does not acquire rights to the programs themselves. There is no obligation on The Author to supply the source code. No right to print or copy AkAbak in whole or in part is granted except as expressly provided under this Agreement or given by law.

The Agreement is effective from the date of receipt by the user of AkAbak and the user's acceptance of the terms and conditions by loading it into any computer workstation.

## 2. RESTRICTIONS

AkAbak is protected by copyright laws and international treaty provisions. Therefore, the user must treat AkAbak like any other copyrighted material except that the user may physically copy AkAbak (i.e. save it on a data medium) solely for backup or archival purposes and transfer it from one computer workstation to another, under the proviso that it is only used on one single computer workstation at any one time. The user may not copy, in whole or in part, the printed materials accompanying AkAbak.

The Author retains any further rights ensuing from the copyrights, and the user renounces entitlement to any such rights. The user may not transfer AkAbak electronically via an interline network or a data transmission channel to other computer workstations or central processing units, or otherwise make it accessible to third parties. The user may not alter the program code, use AkAbak in creations and/or alter or translate written documentation.

The user shall not modify, adapt, merge, translate, reverse engineer, decompile, disassemble or create derivative works based on the whole or any part of AkAbak or its associated documentation, except and only to the extent that such activity is expressly permitted by applicable law.

## 3. TERMINATION

Without prejudice to any of The Author's other rights under this Agreement, if the user fails to comply with the terms and conditions of this Agreement the user shall automatically lose the right to use AkAbak and furthermore become liable to The Author for damages. In such event, the user must destroy all copies of the AkAbak and all of its component parts.

## 4. WARRANTIES, LIABILITY AND DEFECTS

The program has been carefully written but, nevertheless, errors cannot be entirely ruled out. The Author has no knowledge of either the hardware, operating systems and application programs of the user or of the particular benefits the user intends to gain from the use of AkAbak. The Author is therefore not liable for the usability of AkAbak by the user.

The Author does not guarantee any properties of AkAbak other than the details given in its descriptions of the program. The Author guarantees only that AkAbak can be used for normal purposes in an appropriate hardware and software environment as indicated in its descriptions of the program. If the program is operated in a hardware or software environment other than that described in its descriptions of the program, The Author accepts no liability for the correct functioning of the program.

The Author guarantees to the original user that, on delivery, the data medium on which AkAbak is recorded is free of material defects under normal operating conditions and subject to normal maintenance. If the data medium is found to be defective, the user may request a remedy within the guarantee period of 6 months from the time of delivery. To obtain this, the data medium, together with any back-up copy and a copy of the invoice, must be returned to The Author. Where The Author is under an obligation to remedy a defect, it will endeavour to eliminate the defect in the data medium returned, but cannot ensure this in all cases. If the defect cannot be eliminated, the guarantee will be fulfilled by the supply of a new data medium with AkAbak recorded on it. If such a defect is not eliminated by the delivery of a replacement within an appropriate time, the user can require either a reduction of the purchase price or the cancellation of this Agreement and the return of any money paid to The Author in respect of the defective item. Thereafter, the user shall have no further claim against The Author. The user is responsible for the installation of AkAbak. The Author is not liable for defects that can be attributed to improper installation, handling or operation of AkAbak. If, in examining items returned as defective, The Author ascertains that the fault is due to one of the aforementioned causes, it may request from the user compensation for its effort and expenses.

The above mentioned provisions are in lieu of all other warranties, express or implied. In no event will The Author be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use AkAbak.

## 5. OTHER PROVISIONS

Jörg Panzer owns the copyright and the registered trademark of AkAbak and all its components.

Alterations and additions to this Agreement are only valid if made in writing.

This Agreement shall be governed by and construed in accordance with English Law. The user irrevocably accepts the exclusive jurisdiction of the German Courts, but the Author reserves the sole right to bring proceedings in the user's jurisdiction.

If any of the above provisions or portions thereof are deemed invalid under any applicable law, they are to that extent only deemed to be omitted.