A flexible 2/4/8 channel remote level control

Jan Didden

Background

Back in 2005 or so, I decided to modify my DCX2496 digital cross-over/equalizer unit. I completely replaced the 6-channel output section, and in the process added a 6-channel level control which was controlled through a standard IR remote[1] . It turned out that the possibility to control the output levels, not just the overall level but also the relative levels between channels and the balance, from the listening seat was a great asset. Over the years many have build this mod and reactions have been very positive.

Then, last year, Jean-Claude Gaertner (whom I knew from his collaborative articles with Erno Borbely) contacted me with his plans for his Project21[2] . Would I be interested to work on the multichannel level control for P21? The requirement was for a 4-channel level control per stereo side for an up to 4-way active speaker system with DSP processing. The level controls would go between the DSP/DAC outputs and the inputs of the individual power amps for the speaker drivers. He also needed the possibility to shift balance between the 4 channels of the L and R side, as well as the option to offset the relative levels between the 4 ways on a side to compensate for power amp and driver sensitivity differences. The level circuits would be co-located or integrated with the DSP and power amps at each active speaker. Yes I definitely was interested!

Two basic questions needed to be answered: how to implement the actual level control, and how to implement the remote control.

In my DCX2496 mod I had used the 8-channel CS3318 from Crystal Semiconductor. A very high performance, very flexible chip that contains no less than 8 level controls, with a range from +22dB to -96dB in 0.25dB (!) steps with extremely low distortion and noise. This was a chip developed for mixing consoles and such, to be controlled by a simple microcontroller (I used a PIC in the DCX). I was and still am totally happy with the performance of this chip, but there is a practical issue: the CS3318 is housed in a quad-flat-pack with a pin pitch of 0.5mm…. No problem if you need 1000 boards from

1 www.linearaudio.nl/6-chan1.htm

2 Linear Audio Volume 0 (www.linearaudio.net).

a PCB assembly house, but a nightmare for the home assembler (well for many anyway). (I ended up providing the PCB's for the DCX mod with the CS3318 already soldered in place, but still had occasional problems). But the fact that you have 8 channels in a small package on a space-limited PCB made it a must for the DCX project.
With P21 we had more latitude. We looked at TI's offerings in the PGA-series. There's a 4 channel unit the PGA4311 in a SOIC package that can easily be soldered by home builders as the pitch is a leisurely 1.27mm. Electrical specs are on a par with the CS3318; the smallest step size is 0.5dB which is still more than fine enough. The only downside is the maximum input and output signal level of 2.5V RMS, due to the relatively low analog supply voltages of +/-5V. Occasionally you find source components that have output levels at or above this value, or you want more output than that. TI does have versions for the more common analog audio supply levels of +/-15V like the PGA2320, and these can accommodate input signals of up to 10V RMS, which is more than we would ever need. The PGA2320 is only a 2-channel chip however so we need two on each side, but that's a small price to pay. The '2320 also is an SMD chip with an easy 1.27mm pitch. So, that decision made, we turned our attention to the control system.

Early on we realized that a standard IR control system would not be appropriate. Standard IR controls send out command codes that can be interpreted as 'up' or 'down' or 'left' or 'mute'. We all know from experience that sometimes a command is missed because you didn't point it directly at the receiver or due to ambient light. With a multichannel control, with receivers in at least two different locations (L and R), it would be impossible to keep all channels and settings synchronized. What we needed was a system that would send out the actual level setting for each channel in, say, dB's, rather than just 'up' or 'down'. So in case of an 'up' command, what needs to be send to the receiver is a stream of increasing dB-values in say 0.5db steps. Even if a command would be missed, the next one would pull the channel in line again. We needed a system that could send commands and numbers the way we wanted. The initial idea was to cable up the control system between the control unit and the two speakers with something like cat5 cabling, but adding even more cabling to an already crowded room was not very appealing to us (or our spouses). Although I had never worked with RF links, Jean-Claude put enough trust in me to agree to that route. I selected an RF link system, using the license-free 488/966 MHz band, using the RFM23 transceiver modules from Hope RF in China; low-cost, small, self-contained modules that can be controlled from a microcontroller like a PIC (photo 1).

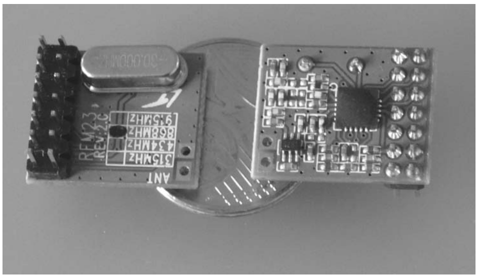

Photo 1: RFM23 module

The deceptively small and simple RFM23 is internally a very complex module that communicates

with whatever controls it through a standard serial link. There are roughly two types of information you need to send to or receive over that link. Obviously, you need to configure the module for the link you want to establish, which involves more than 150 (!) registers to configure anything from the serial link speed and data format, the message error checking information and the transmitter and receiver configuration. You'll read about RF terms that are totally unknown to an audio guy like frequency hopping and preamble detection. Mercifully, almost all default values are fine, and you need to set only a few for your application to establish a very robust and reliable link. In my case, I set the serial link at 56kBd speed, and the message header to 'project21' so that there would be no problem to identify messages as coming from a P21 unit.
The second type of info going across the serial link is the content of the message you want to send out, as well as the received message. The module will signal its controller when a valid message is received which you can then read out and decode. Both send and received messages are available in fifo registers. As far as the format of the messages is concerned, I drafted a simple structure that sends out data in 21 byte messages. For instance, there's a byte that says whether the message is a level set message or a mute/unmute message, one that specifies whether the level data is for left or right side, and then a bunch of level settings for each channel.

Jean-Claude needed a level control board that could be easily integrated with his choice of DSP/DAC modules from Ground Sound (see his P21 article elsewhere in this issue). Therefore, this P21 level board has a form factor, connectors and layout specifically for that application, and that is the unit I will describe here, but at the end I will also show a more universal board that can be used in any similar application. This article will also discuss the control unit design.
So, basically, what I present in this article is a flexible remote level system with up to 8 balanced channels, which can be configured by jumpers on the receivers as we will see later.

Level channels

Fig 1a and b show the 4-channel level control

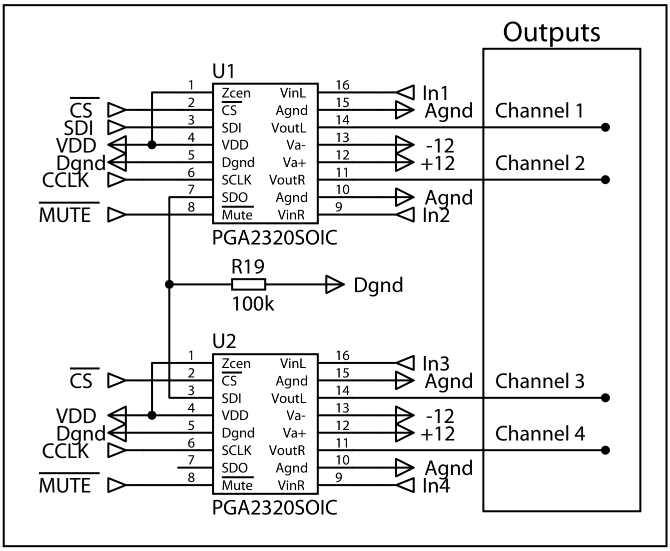

Figure 1a: Level control board, PGA section

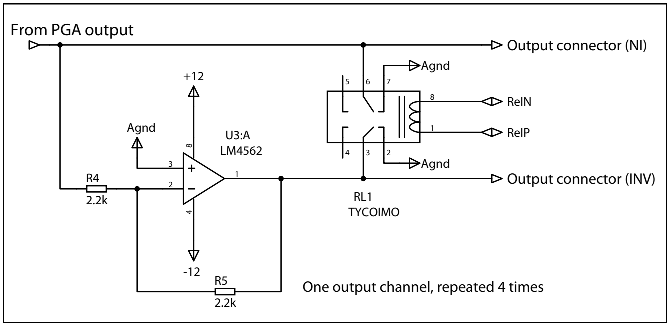

Figure 1b: One balanced output channel

circuit. The four single-ended input signals are input to the PGA, as shown. The four outputs of the PGA are converted into balanced signals using a pair of dual opamps. Next, the output signals are connected to a DPDT small-signal relay for muting purposes. The relay is configured such that the differential signal is shorted by the normally-closed contacts. Therefore, the output is hardware muted until the relays are activated. (In the P21 configuration the relays are activated by the Ground Sound DSP board power-up sequence; in the universal version, the relays are activated by the level control processor). This ensures absolutely quiet power-up, without the relay contacts being in the signal path. Photo 2 shows the level board for the P21 project; there are several additional connectors: a feed-through DSP USB connection, and a DB25 connector to connect the 4 balanced output channels to the four amplifiers that drive the individual drivers (see also the P21 article elsewhere in this issue).

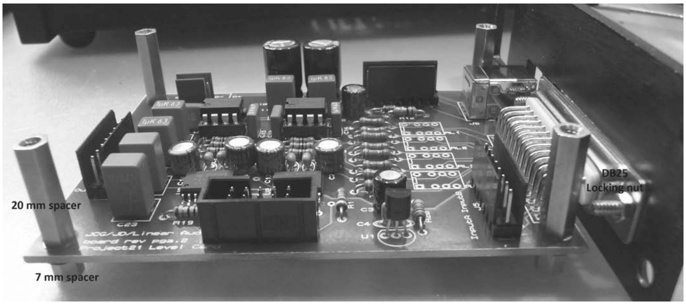

Photo 2: P21 Level board

Receiver

The PGA level settings are controlled by a PIC microcontroller (PIC16F690) which receives commands over the RF link as described above. The control circuit is shown in Fig 2. This circuit is more complex than the level board itself (is that the reason why managers get paid more than workers?) but can be dissected in a couple of functional parts. The processor interfaces directly with the RF module shown in photo 1; the module is plugged on the board through a 2*10 pin socket. The level card is connected to the processor through a flatcable plugged into J1. You see the familiar serial interface signals on the pins. There's also a 2*4 pin header for configuration purposes (DSW1); there's three config positions with pull-up resistors and a supply pin for possible unforeseen uses. Similarly, J4 offers connections to some signals and can act as an expansion port should that be necessary in the future. LED port J2 connects to an LED that lights whenever the unit recognizes a valid RF message, as well as another use described later.

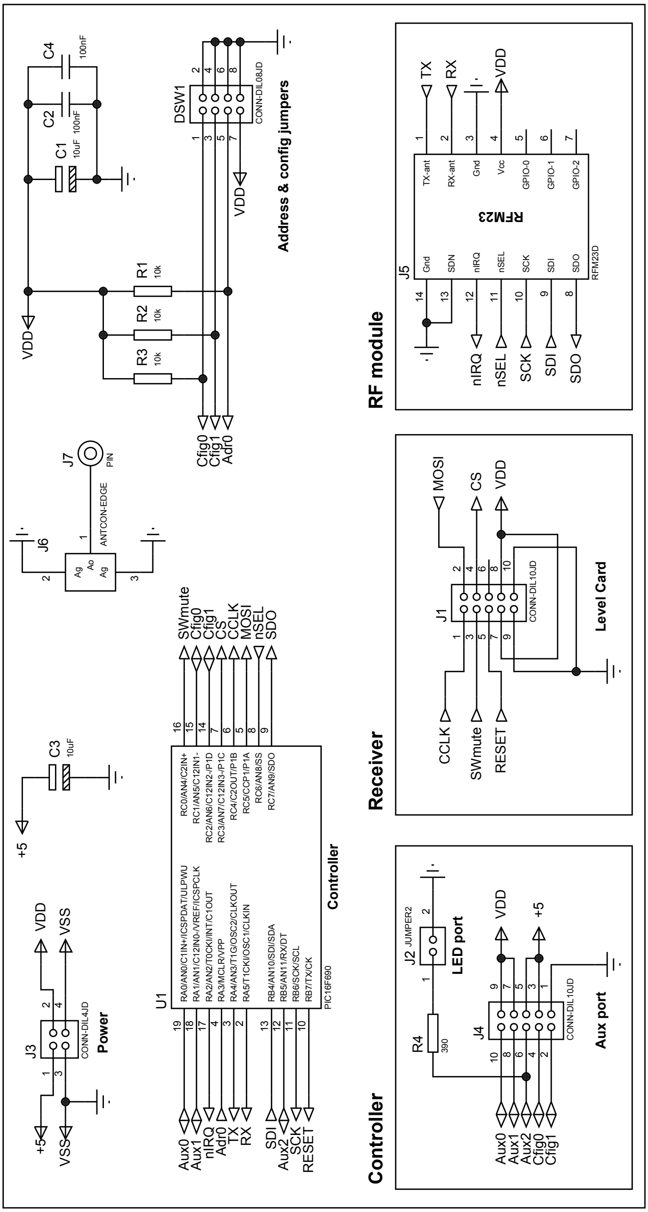

Figure 2: Receiver and level control circuit

The rest are the power supply and RF antenna connector and some supply bypass caps. The receiver board runs on the supply from the level board; the latter receives +/-15V from the Ground Sound board and has onboard regulators for the PGAs, the opamps and for the 3.3V for the receiver board.

Controller

Now we come to the top of the triangle, the control unit. Physically, the unit looks like a small preamp box (photo 3), and in a sense that is what it is, except that in the current configuration there is no source selection. The functions of the unit are summarized in Table 1.

Normally, used as a volume control, you can change the volume of the whole system, all 4 channels both for left and right by turning the rotary knob, shown in Table 1 as 'Vol up' and 'Vol dn'.
The display shows the volume setting in dB below (or above) a gain of one. Pushing the knob however brings up a small menu and then turning the knob steps you through the menu; you activate your selection with another push on the knob. Table 2 shows the available menu choices. To

Photo 3: Control unit

avoid any confusion what menu selection is active and whether you are changing the menu selection or the corresponding level value, the display will show a dot next to the currently active field. As an example, Table 3 shows the display when you are going through the menu selections in a), while b. shows the display when you are changing the balance setting.

After the volume setting, the most used function would be Balance: when the Balance function is selected, and you turn the knob, (you guessed it) you can change the balance between left and right. Again, the display shows the direction of balance change (the loudest side) and how much louder that side is in dB relative to the other side. To keep the volume setting constant, the balance change alternates in 0.5dB steps between increasing one side and decreasing the other side. (see table 3 again)

Then there are three more functions to individually change the levels of channels 1, 3 and 4. This needs some more explanation. The idea here is that you can change the gain in some of the channels of a multiway speaker system to compensate for a different gain in an amp or different sensitivity of a driver. I call this 'Level Offset' mode. Once you 'offset' the level in a channel, this gain offset

	Out1-L	Out2-L	Out3-L	Out4-L	Out1-R	Out2-R	Out1-R	Out4-R
Vol up	^	^	^	^	^	^	^	^
Vol dn	v	V	v	v	v	v	v	v
Bal L	^	^	^	^	v	v	v	v
Bal R	v	V	v	v	^	^	^	^
Out1 up*	^	-	-	-	^	-	-	-
Out1 dn*	v	-	-	-	v	-	-	-
StBy	mute	mute	mute	mute	mute	mute	mute	mute

Table 1. Control functions.
** Similar for Out 3 and Out 4; see text*

Menu option		
Bal	Changes relative level of the 4 L- and R-channels	Alternating in 0.5dB steps
Out1	Changes relative level of Out1 L and R wrt Out2 channel level	0.5dB steps
Out3	Changes relative level of Out3 L and R wrt Out2 L and R channel level	0.5dB steps
Out4	Changes relative level of Out4 L and R wrt Out2 L and R channel level	0.5dB steps
StBy	All channels muted	Display blanked

Table 2. Control menu choices

a: changing menu (currently Bal)	.BAL < 0.5DB VOL -12DB
b: changing balance (currently L at +0.5dB wrt R)	BAL.< 0.5DB VOL -12DB

Table 3: a - going through the menu selections; b - changing the balance setting

is maintained when you change volume or balance. This type of setting is not something you would do a lot of course. In offset mode, the difference in level of the selected channel to the 'overall' level is also indicated on the display in dB (+ or -). That begs the question: what is the overall level? Well, I have unilaterally decided that the level of channel 2 is the overall level. That's why there is no function to change channel 2 level; you change channel 2 level with the Volume function, together with all other channels. Why channel 2? I would assume that most people would use a 4 channel level unit as we used in P21: channel 1 for (sub) woofer, channel 2 for low-mid, channel 3 for mid-high, channel 4 for high. But you're right, I could have selected channel 4 as 'overall level' and the system would work just as well. Let's take an example. Suppose your new subwoofer, driven from channel 1, needs 4dB more signal to sound in balance with the rest of your system. You push the knob and turn it until you get to function OUT1, push again to select that function, the 'dot' in the display now is next to the level value. Then turn the knob clockwise until the display shows '+4dB'. Then push the knob again to return to Volume mode. That's it. You're all set. Photo 3 shows the unit set for mid-balance and a gain of -40dB, and the 'M' shows the outputs are muted.

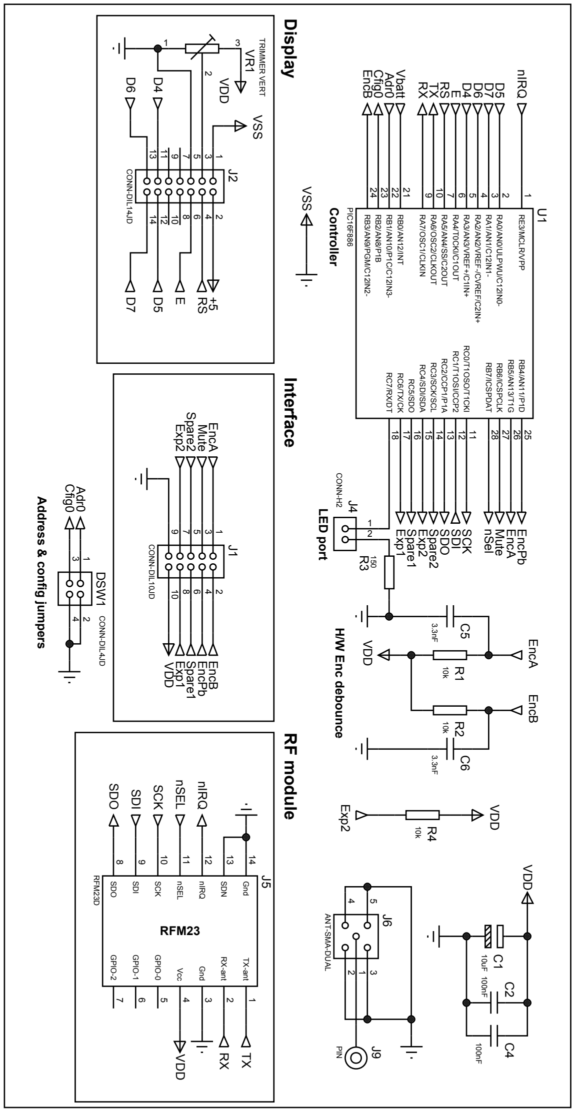

Figure 3: Control unit architecture

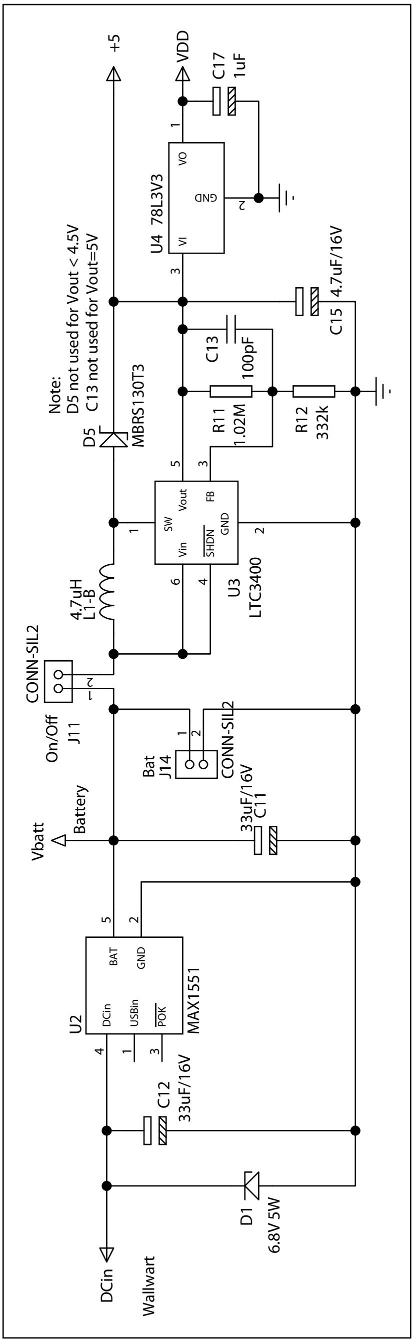

Figure 4: Control unit power supply

The software in the control unit will send out the updated level settings for all channels, anytime any setting is changed by turning the rotary control on the control box (except of course when you are in 'menu' mode). On the receiver side, each message is decoded and will be actioned depending on the configuration jumpers on the board. For instance, if a board is configured as a 'left' board, it will only process the 'left' level settings in a received message. The software running in the control unit and the receivers at the level boards has a few other 'housekeeping' functions. For instance, about 30 seconds after any change, all settings are stored in non-volatile memory. This is also shown by the LED on the receiver board, it flashes briefly when settings are saved. When the system is powered up, settings are restored from this memory.

The overall circuit for the control unit is shown in Figure 3. It looks a lot like the circuit for the receiver board; the main difference is in the PIC software! Instead of the level board port there is the display port. The interface port connects to a rotary encoder/pushbutton (connections EncA, EncB and EncPb) and a pushbutton to Mute the outputs of all channels at the receiver ends. The remainder of the PIC ports connect to the RF module.
To be able to use the control unit at your favorite listening place, it contains a battery that can run the unit for almost a day. It is recharged by plugging a walwart into the box at the back. The power supply subcircuit is shown in Figure 4. It's not rocket science: The max1551 is a switching battery charger, while the LTC3400 converts the varying battery voltage into a steady 5V to run the display. A standard TO-92 3-pin regulator steps that down to 3.3V for the controller and the RF module.

Software

I developed the software for both the receiver and the controller with a graphical programming environment called Flowcode 4, from Matrix Multimedia (www.matrixmultimedia.co.uk). This is a low-cost development environment for several microcontroller families which is perfect for this type of project. The learning curve is easy and you can focus on the logic of your program rather than on the syntax and grammar of the programming language. Actually, Flowcode produces an intermediate result in C-code that subsequently is run through a C-compiler but all this is mercifully hidden from you. Flowcode also includes a very nice logic-level simulator letting you 'run' your program step-by-step while watching what the program does, to catch most of the bugs before you commit to hardware. I used a vintage PicStart Plus programmer from Microchip (www.microchip.com) to program the controllers. The receiver uses a PIC16F690 while the controller, needing more memory space for a larger program, uses a PIC16F886.

A flexible multichannel level control

As noted, this system was specifically developed for Jean-Claude's Project21. However, there's no reason why this cannot be used as a very flexible, RF-remote controlled multichannel level/balance control. I made a version of the level board without the custom connectors and such for the Ground Sound DSP board; just the four channel level control, as shown in fig 6. The board and the stuffing guide is shown in fig 5. If you use two receiver boards each connected to a level board, you can configure the receiver for L or R channel and you have the basic stereo, 4 channel single-ended level control as used in P21. You could put the receiver and level board in your 4-way active speaker or

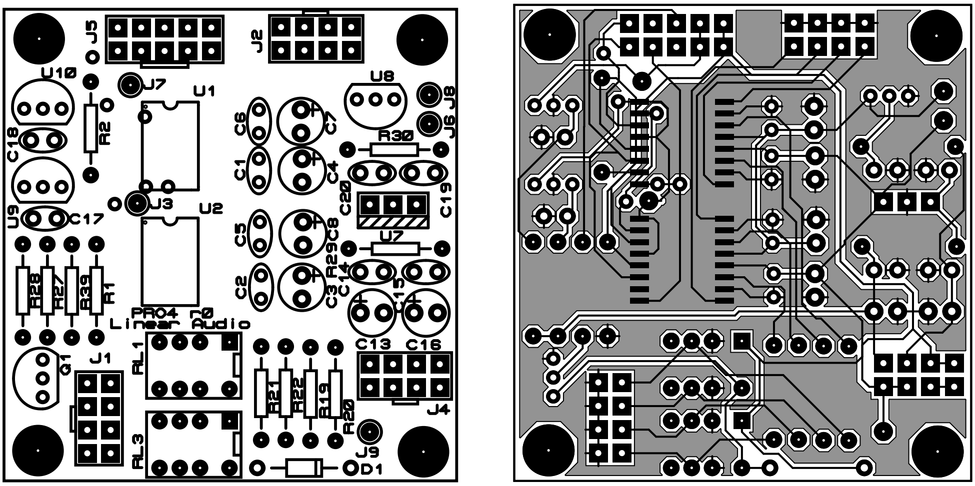

Figure 5: Universal level board

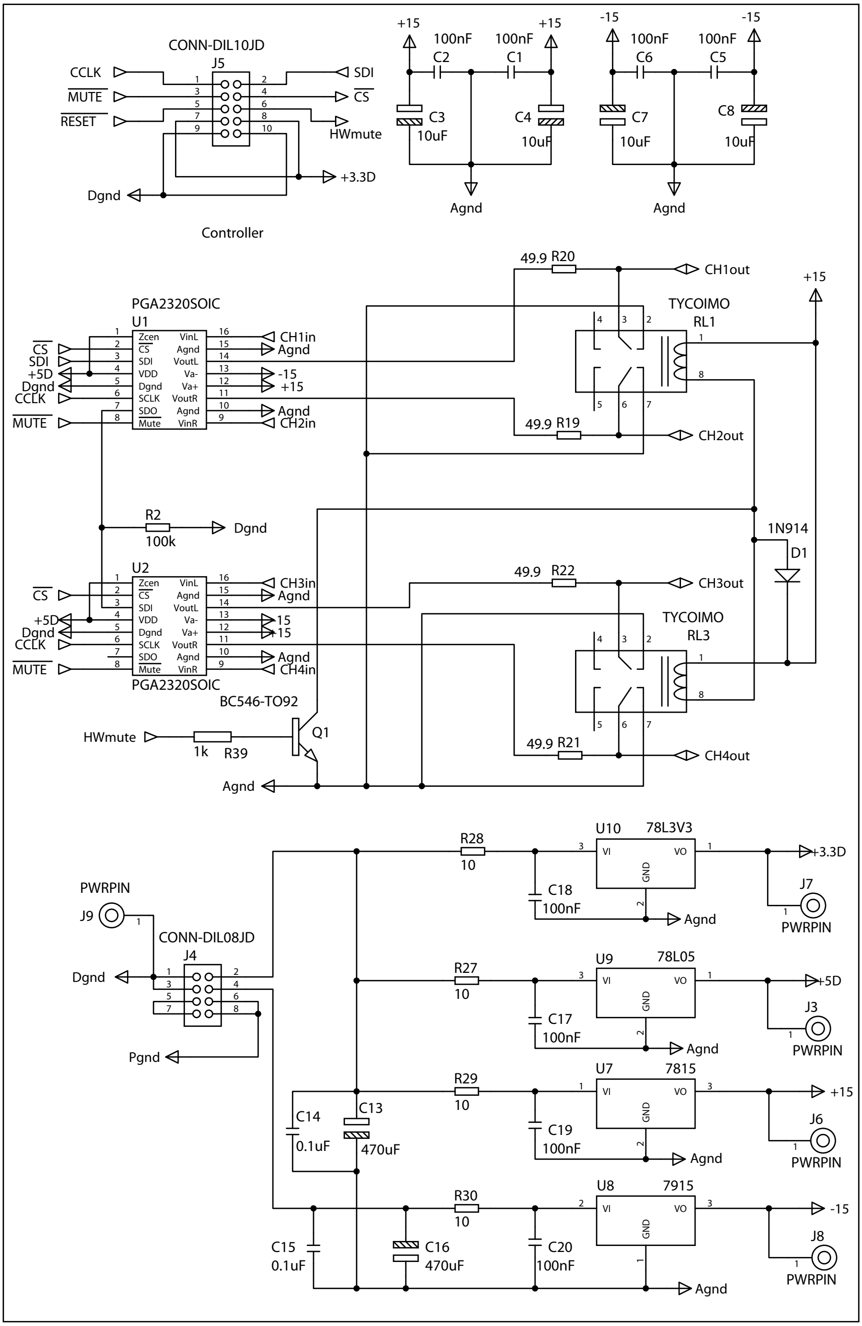

Figure 6: Universal level control circuitry

your 4-channel amplifier and have total level control as described above. You could also use two level boards per side, stacked above each other with their control flat-cable connections in parallel. (There are power pads that can be used to link one board to another above it so that only one board power supply section needs to be populated). That way you have a 4 channel fully balanced level control. You don't need to use 4 channels per side of course; you can use each level board for 2-channel or 3-channels active speakers as well. For example, for a 2-way active speaker you use Out1 for the woofer, Out2 for the mid/high range and leave out the 2nd PGA. In none of these uses is there any change to the controller software or hardware. In the mentioned 2-channel example, you'd use the menu for Out1 to change the relative level of the woofer to the mid/high and never change Out3 and Out4.

If you would like to try your hand at this, I provided the board layouts for this level board as well as the receiver- and controller boards to Pilgham Audio, along with some additional information. If there's sufficient interest they will have a batch manufactured; contact them for details. They will also make available sets of ready-programmed PICs to go with the boards.