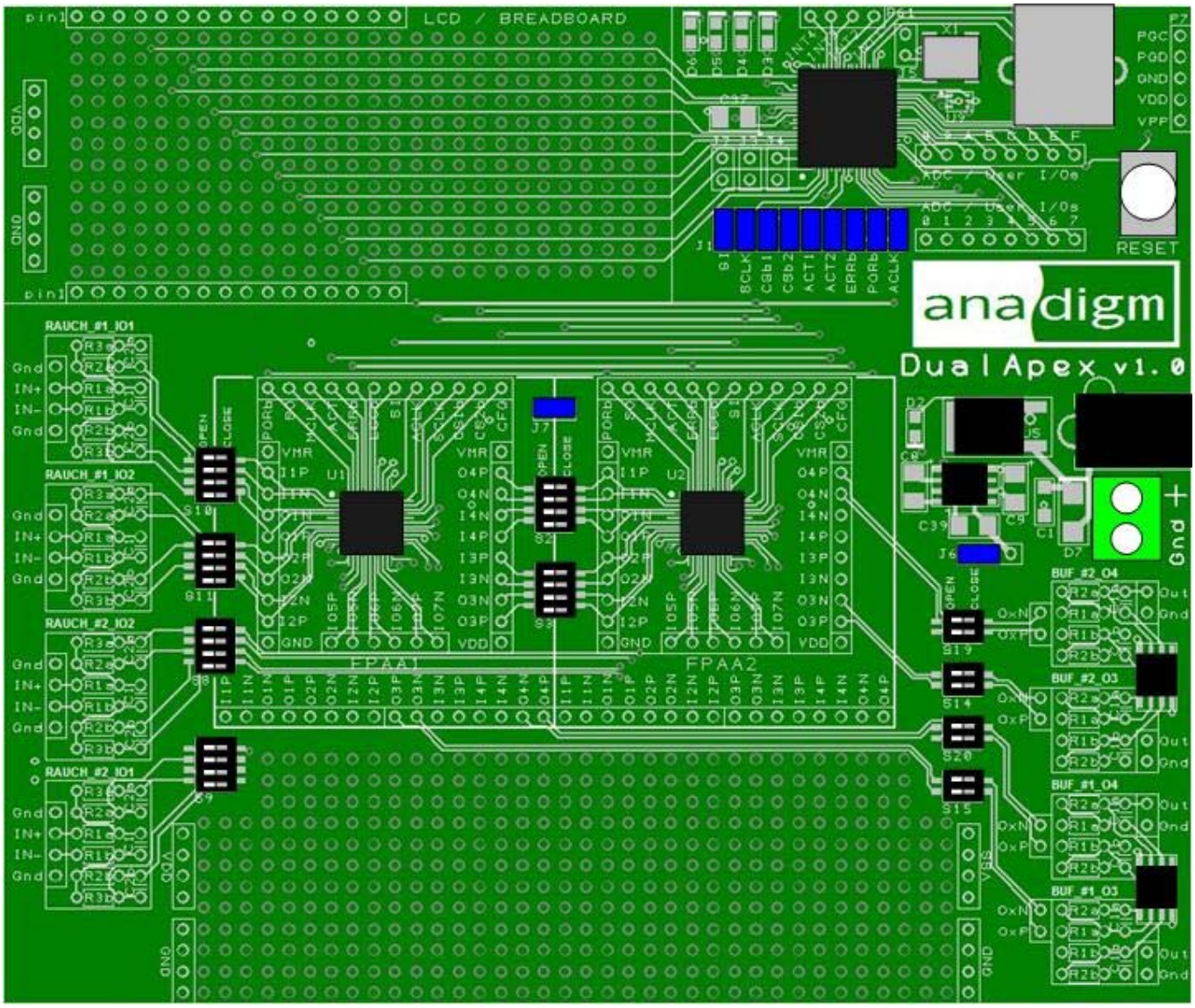


Figure 1: Anadigm DualApex Development Board

Disclaimer

Anadigm reserves the right to make any changes without further notice to any products herein. Anadigm makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Anadigm assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typical's" must be validated for each customer application by customer's technical experts. Anadigm does not in this document convey any license under its patent rights nor the rights of others. Anadigm software and associated products cannot be used except strictly in accordance with an Anadigm software license. The terms of the appropriate Anadigm software license shall prevail over the above terms to the extent of any inconsistency.

1.0 Overview

The Anadigm DualApex development board is an easy-to-use platform designed to help you get started with implementing and testing your analog designs on the AnadigmApex FPAA silicon devices. Furthermore, with its 32bit PIC32 microcontroller and 2 FPAA devices, it provides an extremely powerful platform on which to develop embedded systems.

This manual provides an overview on how to effectively use this board to implement your analog design. But first, here are some salient features of the Anadigm DualApex development board:

Features with this new development board

- **Board footprint – 5.6 x 4.7 inches**
- **Versatile supply requirement – single supply +4 to +7V**
- **On-board regulated +3.3V supply**
- **On-board generated -3.3V supply**
- **Standard USB serial interface for downloading AnadigmDesigner®2 circuit files**
- **32bit PIC32 with 80MHz clock, 512KB program memory, 32KB data memory, 16 x 10bit ADCs**
- **Simple 5 pin header for reprogramming of PIC32**
- **Option for 16MHz or 40MHz analog master clock**
- **16 x 2 LCD display capable**
- **Breadboard areas**
- **Access points to all relevant pins of each FPAA**
- **Input Rauch LP filters, easily configured with passive components**
- **Output LP filter buffers, easily configured with passive components**
- **DIP switches for easy connectivity between FPAAs, input Rauch filters, and output buffers**
- **Ability to electrically isolate digital & analog sections**
- **Ability to configure 1 or 2 FPAA chain**
- **Ability to store configurations in FLASH**
- **Ability to self configure with test circuit**
- **Reset button**

2.0 Layout

Figures 2 shows the layout of the board allowing easy location of all the components, power connections and jumpers.

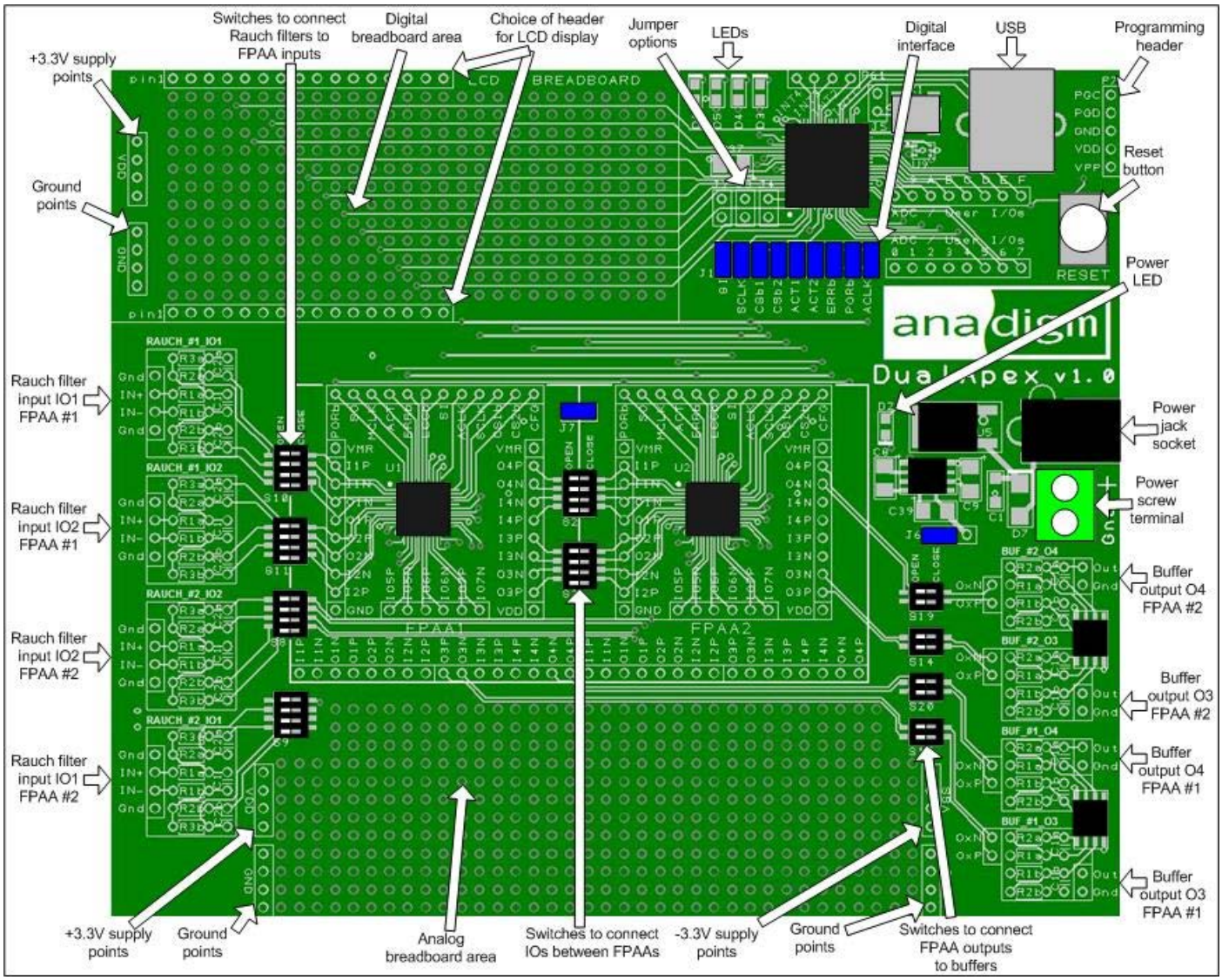


Figure 2: Top-level layout of the Anadigm DualApex board

3.0 Powering up the Anadigm DualApex Development Board

The options for powering up the board are as follows:

- Connect a transformer with centre voltage between +4V and +7V to the jack socket input, OR
- Connect wires from a single precision, regulated supply to the on-board 2-way terminal with the voltage set to between 4V and 7V.

NOTE1: the board is protected against connection to a supply with the wrong polarity

NOTE2: The power supply "jack socket" specification is 5.5mm outer conductor, 2.1mm inner connector, inner (or center) connector is Positive.

WARNING: the board should not be powered with more than 7V

There is a green LED to indicate that the board is successfully powered up. The board should take approximately 100mA when first powered up and before the AN231E04s (FPAAs) are configured. The current after the FPAAs are configured depends very much on the circuit programmed into the FPAAs and how many FPAAs are configured. If both of the FPAAs are configured with a circuit using all of the FPAA's resources, then the board will take approximately 250mA (if an LCD attached but no circuitry added to the breadboard areas).

NOTE3: if a supply of 7V is used to power the board then it is not recommended to let the current exceed 320mA for any length of time. If using a 6V supply then the board can take up to 440mA. If using a 5V supply then the board can take up to 700mA. If a large amount of extra circuitry is to be added to the breadboard areas then Anadigm recommends using a 5V.

4.0 Programming the board

Programming the board in this case means configuring the FPAAs. This can be done from a PC or laptop using a standard type A-B USB cable. The board uses serial port emulation so that from the computer end the link will appear as a COM port.

NOTE1: Windows 8.1 needs to have the "usbser.sys" file loaded to recognise the USB connection to the board.

To program the board (configure the FPAAs) connect the USB cable between the board and a computer and power up the board. Open AnadigmDesigner®2 (AD2) on the computer and go to the Settings menu, then select Preferences. Click on the Port tab. In the drop down menu under "Select Port" there should be a COM port corresponding to the Anadigm board. This is actually a virtual COM port because the on board PIC32 microcontroller is emulating a serial port.

NOTE2: the COM port will have no label indicating that it is the Anadigm board, so if the computer has other devices connected to COM ports (real or virtual) then there may be more than one COM port appearing in the drop down box. In this case it may be necessary to do some trial and error to determine which port corresponds to the Anadgim board.

Select the appropriate COM port and check the box marked USB. Click on Apply, then click on OK. To check that AD2 is now able to communicate with the board, go to the Target menu and select Display Board Information. Figure 3 shows what should be seen.

Figure 3: Board information displayed by AD2

Figure 3 shows the information displayed by AD2 about the board where both FPAAs are set up in a chain. If one FPAA only is set up then the board information will display only one FPAA. For more information on how to set up different numbers of FPAAs see later.

It should now be possible to download a circuit from AD2. To do this simply click on the download icon (to the left of the question mark, beneath Dynamic Config). Alternatively press Ctrl and 'W'. A yellow LED next to the microcontroller will flash as each FPAA is configured, and finally a green LED will turn on to indicate that all FPAAs have been configured correctly. A red LED will indicate that configuration failed.

NOTE3: it is important that the number of FPAAs in AD2 should correspond with the number of FPAAs enabled on the board. For example, if the board has been set up with a chain of 2 FPAAs, then it will be expecting 2 FPAA configurations to be downloaded from AD2.

5.0 Other Features

5.1 Reset Button

There is a reset button at the top of the board, just to the left of the Anadigm logo. This resets both the FPAAs and the PIC32.

5.2 Variable Chain Length

The board contains 2 FPAAs in a chain. This chain can be shortened to 1, but must be shortened by disabling FPAA #2 from the end of the chain (from the right hand side). To reduce the chain to 1 FPAA, remove jumper J7 and also remove the jumper marked ACT2 from J1. This will disable FPAA #2 (nearest the power supply).

NOTE: if a chain of 2 FPAAs is configured and then FPAA #2 is disconnected from the chain, the disconnected FPAA will still hold its configuration until either the board is reset or the single FPAA is configured again.

5.3 FLASH Storage

To place the board in FLASH storage mode, place a jumper on J4 and then reset the board (make sure there is no jumper on J3). After resetting the board, the FPAAs will at first be configured with blank circuits and the green LED will light to show successful configuration. The board will now be in FLASH storage mode which means that any configurations (primary) sent to the board will be stored in FLASH memory in the PIC32 microcontroller. Subsequent resetting of the board will result in these stored configurations being downloaded to the FPAAs automatically. These circuits will be stored indefinitely until they are over-written with other configurations or are cleared by taking the board out of FLASH storage mode.

To take the board out of FLASH storage mode, remove the jumper from J4 and reset the board. The board will no longer be in FLASH storage mode and any stored configurations in the PIC32 will be overwritten with blank configurations.

NOTE1: if only one configuration is stored by configuring a single FPAA (#1) in FLASH storage mode, and the chain of FPAAs is then increased to 2, then on reset of the board FPAA #1 whose configuration was stored will be configured from FLASH with its stored configuration, but FPAA #2 will be configured with a blank circuit.

NOTE2: if 2 configurations are stored by configuring both FPAAs in FLASH storage mode, and the chain of FPAAs is then decreased to 1, then on reset of the board FPAA #1 will be configured from FLASH with its stored configuration, but the extra configuration will not be downloaded.

NOTE3: FLASH storage mode should only be used for storing primary configurations.

5.4 Test Circuit

The board is capable of self configuring the FPAAs with a test circuit for quick checking of the board. To do this, place a jumper on J3 and reset the board. Both of the FPAAs will now be configured with the test circuit (green LED indicates configuration successful). This test circuit outputs a sine wave on all 7 FPAA outputs (14 differential outputs). The frequency of this sine wave is 1kHz on FPAA #1 and 2kHz on FPAA #2. If only FPAA#1 is enabled in the chain then only FPAA #1 will be configured.

NOTE: if a jumper is placed on both J3 and J4, a different circuit will be loaded into the FPAAs. This is a special test circuit used by Anadigm during production. The user is advised not to place jumpers on both J3 and J4.

5.5 Analog Access Points

Each of the FPAAs is surrounded by vias which give access to all of its analog and digital pins.

5.6 VMR

Included in the access points described above is VMR which is brought out to 2 vias, one on each side of the FPAA. This allows the user to connect VMR pins between the FPAAs (with short wires) in the case where only one FPAA in the chain is driving VMR.

5.7 Digital Section

The digital section of the board contains the PIC32 microcontroller, status LEDs, 16MHz oscillator module and USB interface. This section can be electrically isolated from the analog part of the board by removing all of the jumpers from J1, however, if it is required that the FPAAs continue to receive an analog clock (ACLK) then the jumper marked ACLK must be left on. If the user needs to disable the PIC32 microcontroller, one way is to place a jump wire between the pins marked VPP and GND on the programming header P7 (next to the reset button). Again all jumpers except ACLK should be removed from J1. In this case the PIC32 will be disabled but the FPAAs will continue to receive ACLK from the 16MHz oscillator module.

5.8 Analog Clock

The analog clock for the FPAAs (ACLK) is supplied from a 16MHz oscillator module. Alternatively, by placing a jumper on J2 the analog clock can be driven by a 40MHz clock from the PIC32 microcontroller.

NOTE1: the new clock frequency only takes effect after resetting the board.

NOTE2: if using the board in 40MHz analog clock mode, the 16MHz oscillator module should not be disabled by placing a jumper on J5. The reason for this is that the PIC32 (which generates the 40MHz clock) is clocked by this oscillator module so the 40MHz clock is derived from it. So stopping the 16MHz clock will stop the 40MHz clock.

5.9 PIC32 Programming Header

There is a standard 5 pin header P7 to the left of the reset button for re-programming the PIC32. The pins are labelled on the board: PGC, PGD, GND, VDD, VPP. The PIC32 has 512KB program memory. The default software that comes with the board, called the Anadigm Boot Kernel (ABK), uses just 10% of this memory. This software is available for download from the Anadigm website. The user is free to develop his own embedded software.

5.10 PIC32 ADC Channels & User I/Os

The PIC32 microcontroller on the board has 16 pins brought out for the user to use. These pins can be used as standard I/Os (RB0-15) or as 10bit ADC channels (AN0-15). These I/Os and ADC channels are not used by the ABK but are available for the user developing his own embedded software.

NOTE1: AN15 is also used for the LCD so is not available as an ADC channel or I/O if an LCD display is used.

NOTE2: AN6,7 are shared with the programming header. This does not prevent use of these channels, but leaving a programmer connected to the board may cause issues when using these channels.

5.11 LCD Display

A standard 16x2 LCD display can be added to the board. This can be fitted onto one of two 16 pin headers, one immediately below the digital breadboard area and one immediately above the digital breadboard area. The header pins are not fitted as standard so need to be added. The header required is a standard 16 pin 2.54mm (0.1”) pitch SIL header pin. Pin 1 is to the left and is marked.

5.12 Breadboard Areas

There are 2 breadboard areas where the user can add their own circuitry. There is a digital breadboard area at the top left of the board which should be used for digital circuits which may be more noisy. This area is provided with its own power and ground. There is also a larger analog breadboard area running along the bottom of the board. This is provided with positive supply, negative supply and ground.

5.13 Negative Supply

There is a negative supply (-3.3V) which is generated on the board from the positive supply (+3.3V). This negative supply is used for the output buffer stages (bottom left and bottom right), but can also be used by the user for circuits built onto the breadboard. The negative supply is enabled by moving the jumper J6 to the left. It is disabled by moving J6 to the right, in which case the negative supply is grounded.

5.14 DIP Switches

The user can make his own connections on the board with wires, but there are a set of DIP switches that allow for easy connection of certain paths between neighbouring FPAAs and between FPAAs and input/output buffers.

These switches are open by default, which means they are all pushed to the left. The user can close switches by pushing them to the right. A summary of the DIP switches is shown in table 1.

Function	Type	Labels
Connect Rauch filters to FPAA i/ps	4way	S8,9,10,11
Connect between FPAAs	4way	S2,3
Connect FPAA o/ps to buffers	2way	S14,15,19,20

Table 1: DIP Switches

NOTE: The DIP switches are small and of the flush type, so a sharp tool such as a thin screw driver is recommended for opening/closing the switches.

EXAMPLES:
To connect Rauch filter input to I1 of FPAA#1 - close all 4 switches on S10.
To connect O4 of FPAA#1 to I1 of FPAA#2 - close upper 2 switches of S2.
To connect O1 of FPAA#2 to I4 of FPAA#1 - close lower 2 switches of S2.
To connect O3 of FPAA#1 to output buffer - close both switches of S15.

NOTE1: Rauch filters (see section 5.14) use 4 of the FPAA pins, using both the differential inputs AND the differential outputs.

NOTE2: these DIP switches provide the ability to make certain connections on the board very easily and quickly, but the user can make any connections they want by soldering on wires.

5.15 Input Rauch Filters

There are 4 input buffer stages called Rauch filters. These are multi purpose buffers which do the following: they can convert a single-ended signal to differential into the FPAA, they can step up a ground referenced signal to Voltage Mid Rail (VMR = +1.5V) into the FPAA, they can amplify or attenuate an input signal to allow for perfect matching to the FPAA supply, and they provide a low pass filter function (two pole) which is very useful in minimising high frequency noise from being aliased into the FPAA.

The Rauch filter consists of 9 passive components (6 resistors and 3 capacitors) connected to the 4 pins of a type 1 I/O (IOCell1,2,3,4 of the FPAA). The 6 resistors are in the form of 3 identical pairs, the capacitors are in the form of 1 identical pair and 1 on its own. These components work in combination with an amplifier built into the I/O of the FPAA. This amplifier must be enabled for the Rauch filter to work. To enable the amplifier in AD2, double click on the appropriate IO cell (IOCell1-4), select the radio button marked Input, select the radio button marked Amplifier (Filter). Click on OK. Figure 4 shows the amplifier as it appears in AD2 (left), a schematic showing how the components are connected (middle) and the layout of one of the Rauch filters provided on the board.

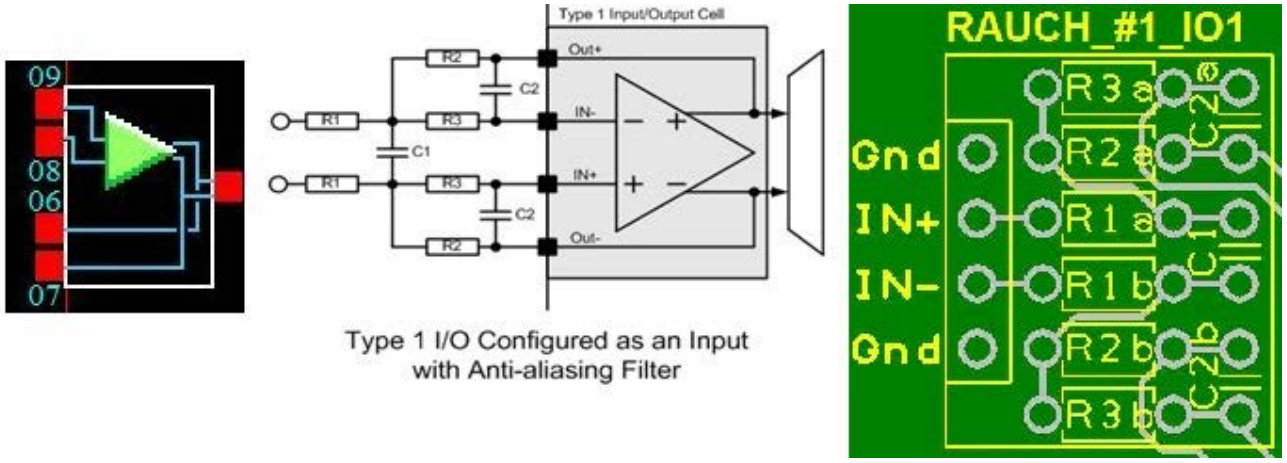


Figure 4: Rauch Filter

The equations for gain, corner frequency and Q of the Rauch low pass filter are as follows:

G = R2 / R1

Fo = [1 / (2.π.R2)] x SQRT[(R1+R2) / (2.C1.C2.R1)]

Q = SQRT[(C1.R1) / (2.C2.(R1+R2))]

To determine component values for required values of gain, corner frequency and Q, first select a suitable value for R1 (e.g. 10kΩ) and then use the following equations to calculate the other component values:

R2 = R1.G

R3 = R1.G / (G+1)

C1 = Q.(G+1) / (2.π.G.Fo.R1)

C2 = 1 / 4.π.G.Fo.Q.R1

As an example, suppose it is required to have the following parameters in the input stage:

Gain = 1.5
Fo = 20kHz
Q = 0.707

First a suitable value for R1 must be decided e.g. 10kΩ. Then using the equations for R2, R3, C1 and C2 we get:

R1 = 10kΩ
R2 = 15kΩ
R3 = 6kΩ (nearest preferred value is 5.6 kΩ)
C1 = 0.938nF (nearest preferred value is 1nF)
C2 = 0.375nF (nearest preferred value is 330pF)

The preferred component values can be fed back into the original equations to determine the realised parameters. There is an Excel spreadsheet provided by Anadigm that makes the whole process much simpler. This can be downloaded from the Anadigm website.

NOTE1: if the input signal to the Rauch filter is single-ended, then simply connect the left side of one of the R1 resistors (see schematic in figure 4) to ground.

NOTE2: for best noise results, the low pass corner frequency should be set just above the highest frequency of the input signal.

NOTE3: for best noise results, the gain should be set so that, for the maximum amplitude of the input signal, the output of the Rauch filter is just within 0 and 3V supply, corresponding to a +/-3V (6V pk pk) differential signal. For example, if the maximum input signal is 1V pk pk single-ended, then the gain should be set to 6 (or just under).

NOTE4: the Rauch filter will not work unless the input amplifier is enabled in AD2.

The board has footprints for 4 Rauch filters, 2 to each FPAA. The typical layout for these is shown on the right in figure 4. Note that there is only one C1, the other components (3 resistors and a capacitor) are in pairs marked ‘a’ and ‘b’, and these pairs should always be identical. The user is free to add through hole components to these Rauch footprints to make their own version (none are populated with through hole components but 2 are populated with surface mount components – see below).

For the user who wishes to use the board straight out of the box, 2 of the Rauch filters are already populated with surface mount components underneath the board. These are shown in table 2.

Rauch Filter	Component Values	Parameters
Rauch_#1_IO1	R1 = R2 = 22kΩ R3 = 10kΩ C1 = 22pF C2 = 10pF	G = 1.0 Q = 0.707 Fo = 490kHz
Rauch_#1_IO2	R1 = R2 = 470kΩ R3 = 220kΩ C1 = 22pF C2 = 10pF	G = 1.0 Q = 0.707 Fo = 23kHz

Table 2: Ready populated Rauch filters

These 2 Rauch filters are provided ready populated as examples. Rauch_#1_IO1 is connected to I/O1 of FPAA #1. It is set up with a very high corner frequency and unity gain. If using a single-ended signal, connect IN- to GND (see layout on the right of figure 4) and connect the signal to IN+. If a lower corner frequency is required, it is very easy to modify this filter without removing the surface mount components beneath, simply by adding through hole capacitors. For example, if a 2.2nF capacitor is added to C1 and 1nF capacitors added to C2a and C2b, the corner frequency can be reduced to 4.9kHz (increasing the capacitor values reduces corner frequency by the same factor).

Rauch_#1_IO2 is connected to I/O2 of FPAA #1. It is set up with a unity gain and corner frequency of just over 20kHz which is typical of an audio application. If using a single-ended signal, connect IN- to GND (see layout on the right of figure 4) and connect the signal to IN+. If a lower corner frequency is required, it is very easy to modify this filter without removing the surface mount components beneath, simply by adding through hole capacitors. For example, if a 2.2nF capacitor is added to C1 and 1nF capacitors added to C2a and C2b, the corner frequency can be reduced to 230Hz. Alternatively, if a higher corner frequency is required, through hole resistors can be added, again without removing the surface mount components beneath. For example, if 10kΩ resistors are added to R1a, R1b, R2a and R2b, and 4.7kΩ resistors are added to R3a and R3b, then the corner frequency can be increased to 1MHz (reducing the resistor values increased the corner frequency by the same factor).

The other 2 Rauch filters are not populated at all, so the user is free to design their own using the equations mentioned earlier or the Excel spreadsheet available from Anadigm's website. If the user does not wish to use Rauch filters, it is recommended to open the DIP switches connecting the Rauch to the FPAA and connect directly to the FPAA pins using wires soldered to the labelled vias that surround each FPAA.

5.16 Output Buffers

There are 4 output buffers, each pair is connected via DIP switches to each of the 2 FPAAs. The circuit for each of these buffers is shown in figure 5. Like the input Rauch filters, these buffers are multipurpose and perform the following functions: they convert the differential output from the FPAA to single-ended, they step down the FPAA output from VMR to ground, they can amplify or attenuate the signal out of the FPAA, and they provide a low pass filter function (one pole) which is useful for removing clock noise and smoothing the quantisation of the output.

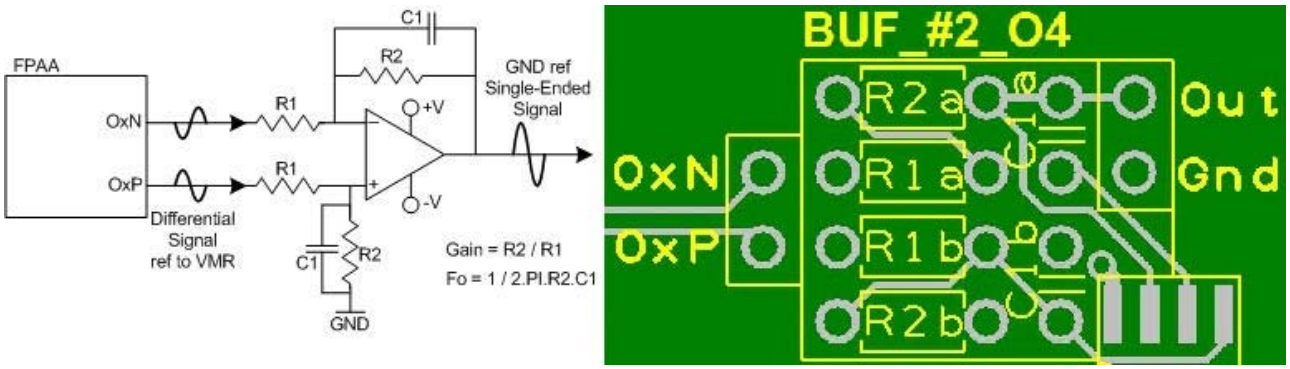


Figure 5: Output Buffer

The equations for gain and corner frequency of the output buffer are as follows:

G = R2 / R1

Fo = 1 / (2.π.R2.C1)

For the use who wishes to use the board straight out of the box, 2 of the output buffers are already populated with surface mount components underneath the board. These are shown in table 3.

Rauch Filter	Component Values	Parameters
Buf_#1_O4	R1 = 33kΩ R2 = 33kΩ C1 = 10pF	G = 1.0 Fo = 480kHz
Buf_#1_O3	R1 = 470kΩ R2 = 470kΩ C1 = 15pF	G = 1.0 F = 23kHz

Table 3: Ready populated output buffers

These 2 output buffers are provided ready populated as examples. Buf_#1_O4 is connected to O4 of FPAA #1. It is set up with a very high corner frequency and unity gain. If a lower corner frequency is required, it is very easy to modify this filter without removing the surface mount components beneath, simply by adding through hole capacitors. For example, if 1nF capacitors are added to C1a and C1b, the corner frequency can be reduced to 4.8kHz (increasing the capacitor values reduces corner frequency by the same factor).

Buf_#1_O3 is connected to O3 of FPAA #1. It is set up with a unity gain and corner frequency of just over 20kHz which is typical of an audio application. If a lower corner frequency is required, it is very easy to modify this filter without removing the surface mount components beneath, simply by adding through hole capacitors. For example, if 1.5nF capacitors are added to C1a and C1b, the corner frequency can be reduced to 230Hz. Alternatively, if a higher corner frequency is required, through hole resistors can be added, again without removing the surface mount components beneath. For example, if 10kΩ resistors are added to R1a, R1b, R2a and R2b, the corner frequency can be increased to 1MHz (reducing the resistor values increased the corner frequency by the same factor).

The other 2 out buffers are not populated at all, so the user is free to design their own using the equations mentioned earlier. If the user does not wish to use output buffers, it is recommended to open the DIP switches connecting the buffer to the FPAA and connect directly to the FPAA pins using wires soldered to the labelled vias that surround each FPAA.

6.0 Jumpers

Table 4 shows a complete list of the jumpers on the board and figure 6 shows their positions.

Jumper	Function	Default State	Default Condition
J1	Connects the digital section to the analog section	All 9 jumpers should be on	Fully connects power, ground and all FPAA digital signals to the digital section
J2	Selects between 16MHz and 40MHz for ACLK source	Jumper off	ACLK = 16MHz
J3	Enables download of test configuration to both FPAAs from FLASH (after reset or power cycling). Test circuit outputs sine wave to all FPAA outputs.	Jumper off	Test circuit download disabled
J4	Enables the storage of primary configurations in the PIC32's FLASH. If the jumper is on, the board will remember the last primary configuration after reset or power cycling. If a configuration has not yet been sent from AD2 then the board will configure itself with blank circuits in both FPAAs. If the jumper is removed and the board reset or power cycled, the FLASH will be cleared.	Jumper on	Not in primary configuration storage mode
J5	A jumper on J5 will disable the 16MHz oscillator module and tristate its output. This means that the ACLK pin of the FPAAs will not be clocked, also the PIC32 microcontroller will not be clocked, so the whole board will effectively be disabled.	Jumper off	16MHz oscillator enabled
J6	This jumper controls the generation of a negative supply (-3.3V) for the output buffer stages on the board. Jumper to the right disbales the negative supply (ties it to ground). Jumper to the left enables the supply.	Jumper to the right	Negative supply disabled
J7	This jumper connects the LCCb pin of FPAA #1 to the left to the CS1b pin of FPAA #2 to the right. This effectively chains the 2 FPAAs together. To shorten the chain to 1 FPAA, remove J9 and also the ACT2 jumper from J1.	Jumper on	Both FPAAs chained

Table 4: Summary of Development Board Jumpers

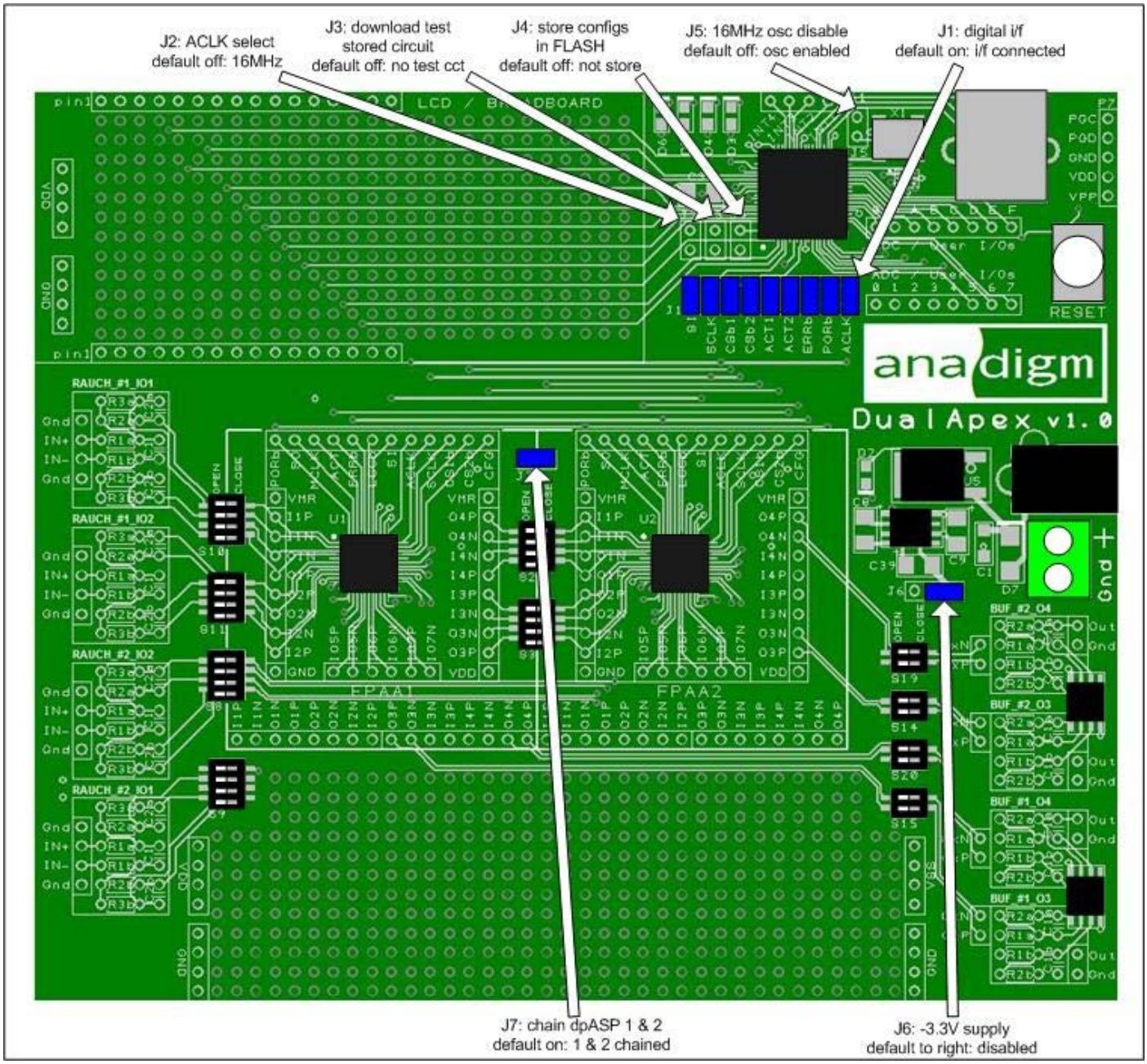
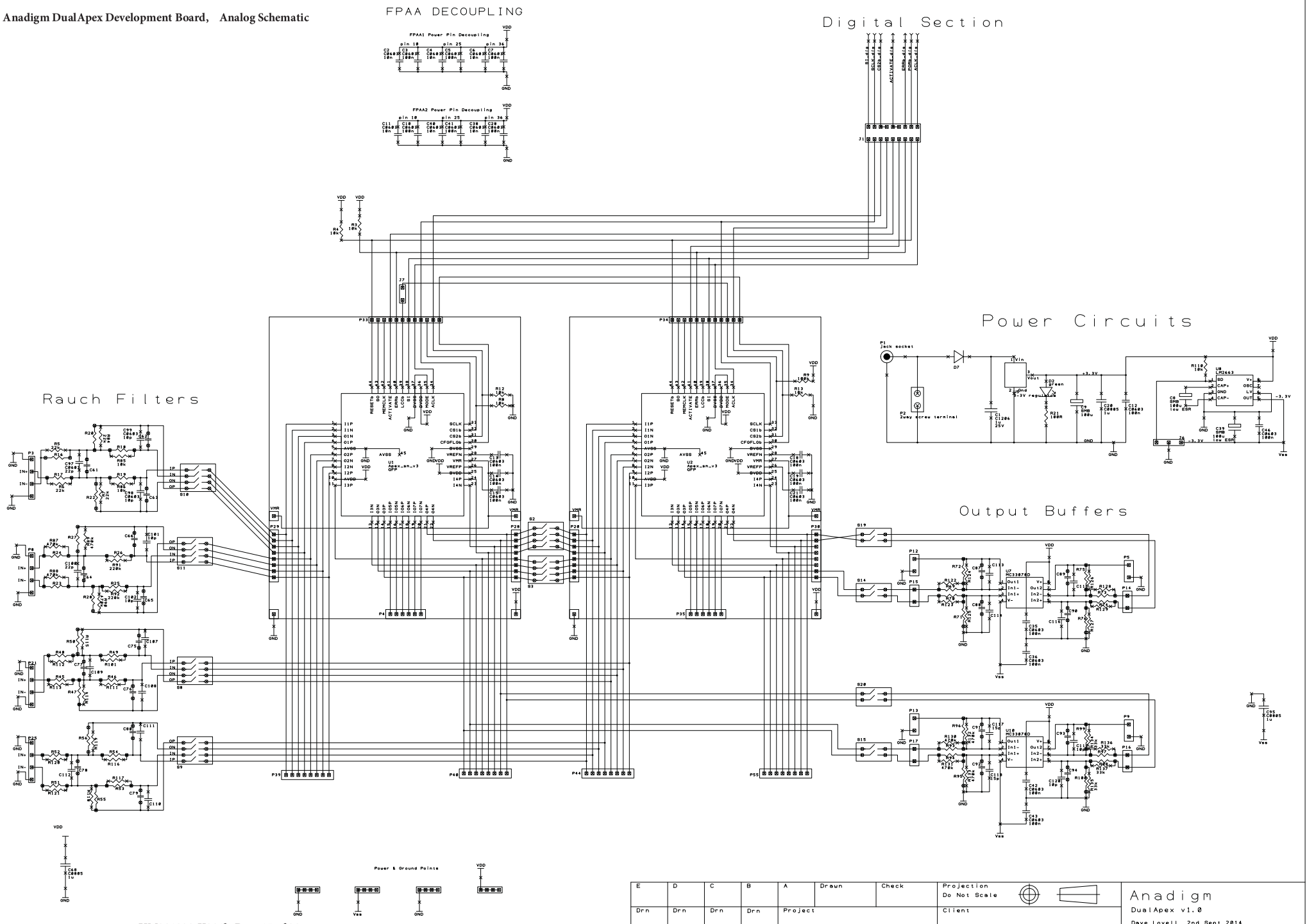


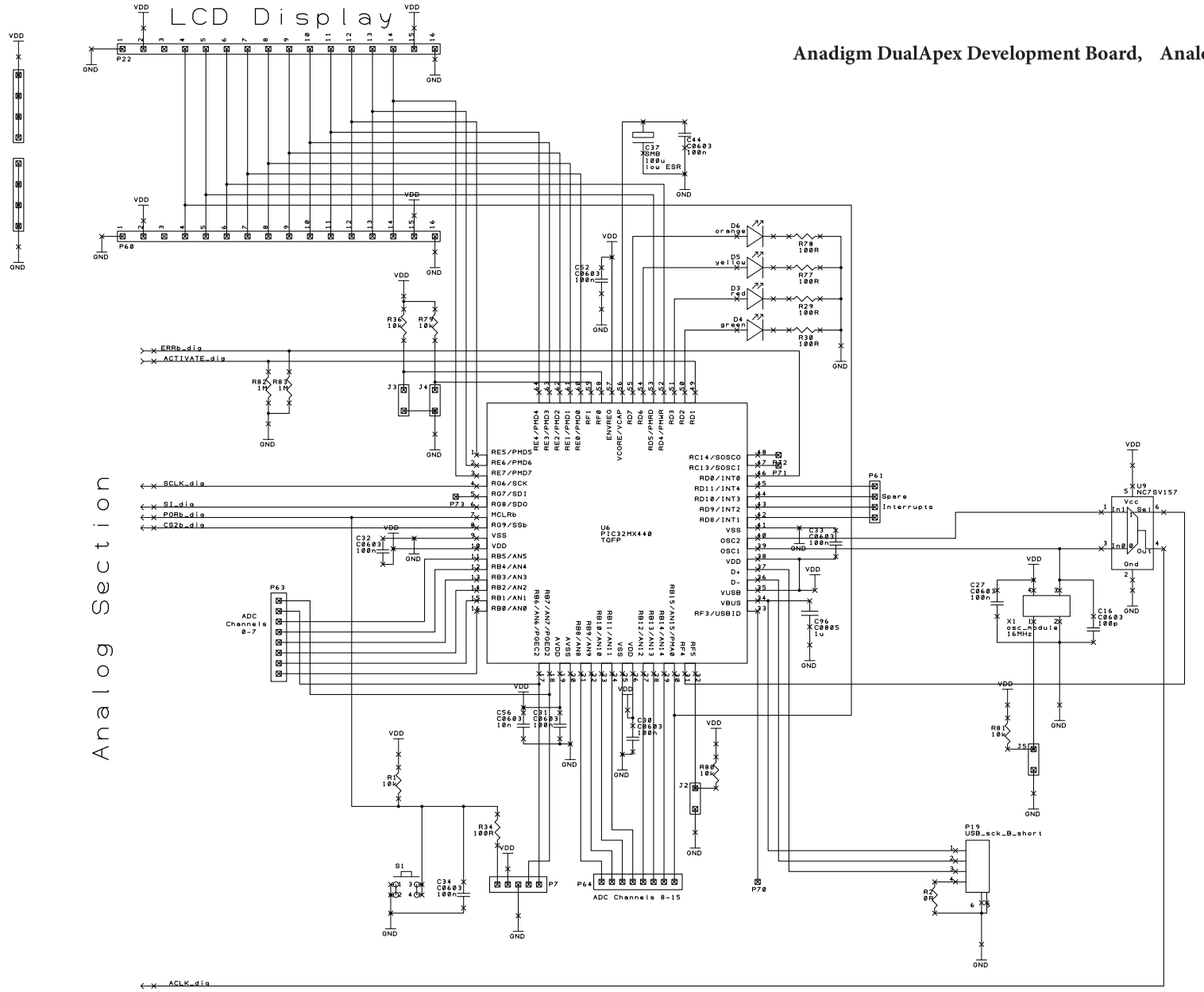
Figure 6: Positions of Jumpers

7.0 Absolute Maximum Ratings

Parameter	Symbol	Min	Typ	Max	Unit	Comment
DC Power Supply 3.5mm jack socket	V_{jack}	4	5	6	V	DC voltage only Centre pole is positive, outer sleeve is ground
DC Power Supply Screw terminal "+" post	V+	4	5	6	V	DC voltage only Voltage is relative to "Gnd" post
DC power supply regulator, max current, 7 volts input 6 volts input 5 volts input	Isupply	- -		320 440 700	mA mA mA	At room temperature The FPAAs require typically 40mA each (max 75mA). The additional available power may be used by the microcontroller and components added to the breadboard The regulator is capable of delivering 1A but the current is limited by the regulator power dissipation.
FPAA Input Voltage	F_{in}	-0.5	+3.6		V	Direct input to FPAA on analog IO access points or digital pins (J1 of FPAA access points)
FPAA Output Voltage	F_{out}	-0.5	+3.6		V	Direct output from FPAA on analog IO access points or digital pins (J1 of FPAA access points)
Operating Temperature	T_{op}	10		50	^oC	Ambient Operating Temperature
Storage Temperature	T_{stg}	-20		70	^oC	Ambient Storage Temperature



E	D	C	B	A	Drawn	Check	Projection Do Not Scale	Anadigm	
Drn	Drn	Drn	Drn	Project			Client	DualApex v1.0	
Chk	Chk	Chk	Chk	Title			Filename	Dave Lovell 2nd Sept 2014	
								Drawing No.	Sheet of



E	D	C	B	A	Drawn	Check	Projection Do Not Scale	Anadigm DualApex v1.0 Dave Lovell 2nd Sept 2014
Drn	Drn	Drn	Drn	Project			Client	
Chk	Chk	Chk	Chk	Title			Filename	Drawing No. / Sheet of

OTL2
A3

Notes:

Notes:

Notes:

Notes:

For More information Contact

anadigm®

http://www.**anadigm.com**
support@anadigm.com