DP5 Enhanced Digital Processor

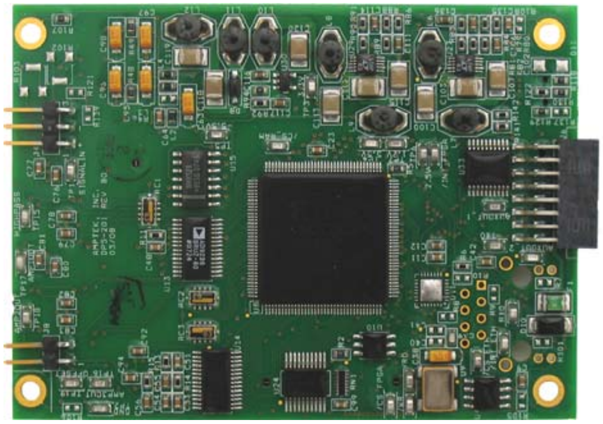
Product Migration Guide

Amptek Inc is introducing a family of enhanced X-ray spectroscopy products, based on a new generation digital pulse processor, the DP5. The DP5 will provide better performance, new signal processing features, improved data processing algorithms, faster data transfers, and a simpler interface. Amptek has also developed an enhanced power supply board for use with Amptek detectors and preamplifiers, the PC5. Products now available using the DP5 and PC5 include:

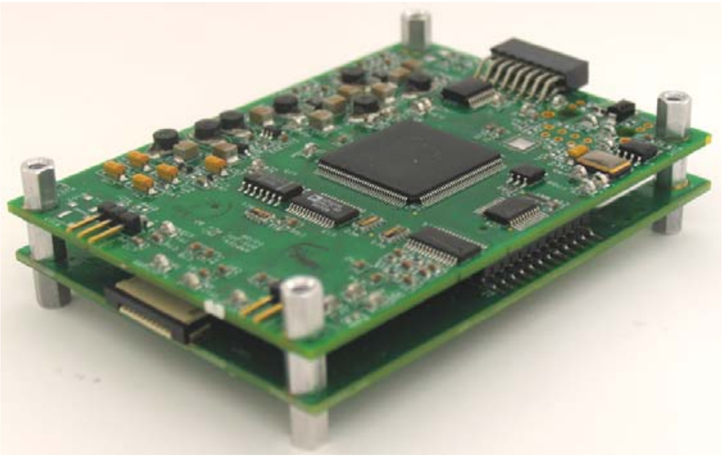
1. The X-123SDD, a significantly enhanced X-123 which uses the new silicon drift detector (SDD), the DP5 digital processor, and a new PC5 power supply module.
2. The DP5/PC5 combination, used primarily by OEMs with Amptek's detectors and preamplifiers.
3. The DP5 as a standalone unit.

The DP5 implements all of the features and performance of the DP4, with many enhancements. Communication protocols and software are generally compatible with the DP4. Specifically, accessing new features of the DP5 requires software changes but software written for the DP4 will work with the DP5 with minor changes, providing the capabilities of the DP4.

This document describes how users of the existing products can migrate to the new family of products. The focus of this document is on the changes to the products and their interfaces and presume familiarity with the prior products. This document should be used together with the DP4 User Manual.

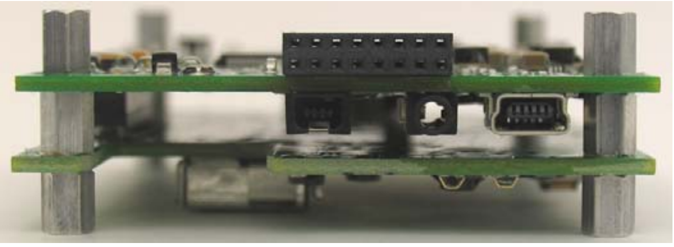


DP5



DP5/PC5 stack

X-123SDD



DP5/PC5 stack back connector view

1. FREQUENTLY ASKED QUESTIONS

1.1 WHAT ENHANCEMENTS WILL BE FOUND IN THE DP5 VS. THE DP4?

Performance enhancements

- Faster peaking time available, allowing higher input count rates. Shortest peaking time is 0.2 μsec (equivalent to 80 nsec shaping time).
- Additional peaking times are available, with over 350 settings covering 0.2 μsec to 102 μsec.
- Reduced dead time allowing higher throughput. Dead time is typically 10% shorter.
- Dead time due to USB & RS232 transfers is significantly reduced.
- Improved dead time measurement for more accurate corrections at high rate.
- Fast channel peaking time as short as 0.1 μsec (40 nsec shaping time). This improves pile-up rejection, measurement of incoming count rate, and risetime discrimination.
- Larger gain range, with all ranges overlapping. (16 analog gains, from 1x to 100x.)
- Lower noise analog prefilter. This improves resolution with the lowest noise detectors.
- Input offset, input polarity, and preamp pole zero under software control.
- Software control of the offset channel in the output spectrum, permitting zero offset.

Interface enhancements

- Ethernet (10-base T) (will be available summer 09)
- USB upgraded to 2.0 full speed (12 Mbps).
- RS232 upgraded to 115 kbaud (57.6 kbaud selectable).
- No daughter boards (PC4-2, etc) required for DP5 operation.
  - Power supplies are now on the DP5 board, so it only requires +5V input.
  - DP5 can be powered from USB (but not when connected to PC5 for Amptek detectors).
  - Communication connectors on the DP5 board.
- Power supply board for Amptek's XR100 detectors (new PC5) is under software control, to set temperature and high voltage bias.
- Firmware, both in FPGA and microcontroller, can be upgraded via standard serial interfaces by the user.
- New "List Mode" will be available in the next release. Instead of a spectrum, each event generates an amplitude and a time tag, which are stored and can be downloaded.
- New "Setup" spectrum can be acquired, showing the noise Gaussian, events rejected by pile-up rejection and risetime discrimination, by gating logic, etc. Useful for optimizing parameters.
- A "board interconnect" has been added to simply interface to daughter boards with other capabilities, or to systems containing multiple digital processors.
- A 512kB low-power SRAM was added to the DP5, adding buffering flexibility. For example, sequential spectra can be buffered for later transmission to the PC.

1.2 IS THE DP5 BACKWARD COMPATIBLE WITH THE DP4?

As close as feasible but some changes are inevitable.

Software

- A new release of Amptek's ADMCA software incorporates the DP5 interface over USB. With ADMCA one can access all of the DP5's features in the same manner as the DP4 or PX4.
- The DP5 uses FLASH memory to store its power-up configuration and some parameters which are unsupported by the legacy DP4 & PX4 protocols (for example, switching to 80MHz operation to access the fastest peaking times, selecting 115k/57.6k RS232 operation, etc.). These options can be selected via the standard serial interfaces.
- For users who have written or are writing their own software, there are a few options.
  - To use the new functions and features of the DP5, new data packets and communications libraries are required. This implies some software changes, for either USB or RS232.
  - With RS232, the DP5 recognizes the same data packets used with the DP4 and PX4. There are some parameters which must be specified with the DP5 which were not used in the DP4, e.g. input offset. These can be set once, during initial product configuration, and are then stored in the DP5's EEPROM. A stand-alone application is provided by Amptek to set the new features and store them. With this legacy DP4 software can be used unchanged.
  - With USB, the DP5 recognizes the same packets for commanding and for readout of spectra and status packets. Readout of configuration and of oscilloscope packets requires minor software changes. (The DP5 microcontroller supports fewer USB 'endpoints' or 'pipes' than the DP4 microcontroller, so a few required remapping.) An updated Windows API interface (DPP API) is available that is also backwards compatible with the DP4 and PX4.

Physical

- DP5 Standalone
  - The DP5 has the same physical size and mounting holes as the DP4.
  - The preamp input connector and DAC output connector are different, but in the same location.
  - Additional interface connectors have been added, eliminating the need for daughter boards and reducing the number of supply voltages required. The backplane (JP6) and auxiliary (JP9) connectors have been changed and moved. Standard USB, serial, Ethernet, and power supply connectors are now on the DP5.
- X-123SDD and DP5/PC5
  - The X-123SDD USB and serial connectors have been moved slightly. A different power connector is now used. An Ethernet connector has been added, although it is not yet supported in software.
  - With the PC5, the 8 pin power & serial header used previously is still available. The same USB and serial connectors are still available, moved from the PC4-2 and PC4-3 to the DP5.
- The power dissipation will increase to 0.9 W for a 20 MHz clock. To use faster peaking times, an 80 MHz clock is required, with dissipation of 1 W typical.

1.3 HOW WERE THESE ENHANCEMENTS ACHIEVED?

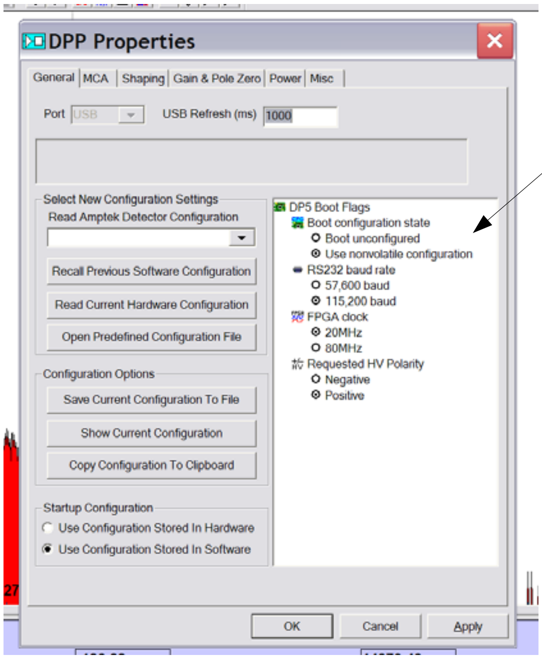
- The DP4's ADC and digital processor operated at a 20 MHz clock rate. The DP5 can be clocked at 20 or 80 MHz. The higher clock rates permit the faster peaking times in both the main and fast channels but the unit dissipates more power in this mode. For systems not requiring the fastest peaking times or operating at the higher rates, the 20 MHz clock can be used to reduce power.
- A number of improvements were made to the digital signal processing algorithms and to the analog prefilter. Amptek has conducted significant in-house research and development on digital signal processors since the release of initial DP4 and has tailored the design to meet the unique needs of many customers. These developments were folded into the new DP5.

- The constantly shrinking physical sizes of many components permitted us to move the power supplies and connectors from daughter boards, used with the DP4, onto the DP5. This significantly simplifies system integration.

CONTENTS

2. CHANGES TO ADMCA

Many users rely on Amptek’s ADMCA software for interfacing to the DP4 and PX4. With ADMCA, migration to the DP5 is quite straightforward. There are some additional configuration options in the “Acquisition Setup” button (DPP Properties). The screen captures below show which portions of the “acquisition setup” menus have changed. Note: the latest version of ADMCA, version 1.0.8.17, supports the DP5. For more information on changes to the software interface and an explanation of what is happening “under the hood” in the DP5, refer to section 5 of this document.


DPP Properties
General
MCA
Shaping
Gain & Pole Zero
Power
Misc
Recall Previous Software Configuration
Read Current Hardware Configuration
Open Predefined Configuration File
Save Current Configuration To File
Show Current Configuration
Copy Configuration To Clipboard
OK
Cancel
Apply

Port USB

USB Refresh (ms) 1000

Select New Configuration Settings

Read Amptek Detector Configuration

Configuration Options

Startup Configuration

○ Use Configuration Stored In Hardware

● Use Configuration Stored In Software

DP5 Boot Flags
- Boot configuration state
  - ○ Boot unconfigured
  - ● Use nonvolatile configuration
- RS232 baud rate
  - ○ 57,600 baud
  - ● 115,200 baud
- FPGA clock
  - ● 20MHz
  - ○ 80MHz
- Requested HV Polarity
  - ○ Negative
  - ● Positive

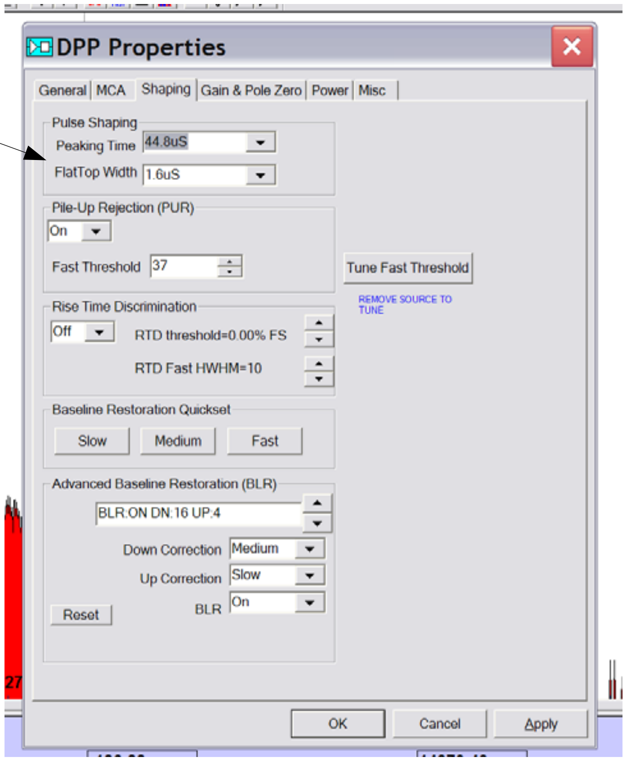
This section new for DP5

The legacy DP4 and PX4 used 20 MHz. DP5 uses 20 MHz for same performance of 80 MHz for faster peaking times. Select button and hit "apply".

All other options in this menu only take effect when the DP5 boots, i.e. it must be power cycled.

For HV polarity, the DP5 checks the polarity setting and only applies HV if the hardware setting matches that selected here. Set to negative only for SDDs

With DP5, shorter peaking times and flat tops available if clock set to 80 MHz



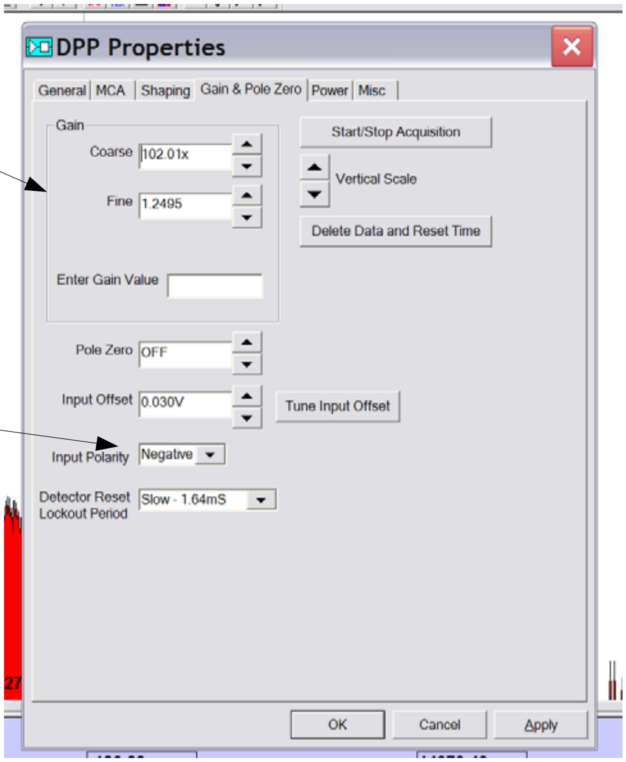
Pulse Shaping
Peaking Time 44.8uS
FlatTop Width 1.6uS

Pile-Up Rejection (PUR)
On
Fast Threshold 37

Rise Time Discrimination
Off
RTD threshold=0.00% FS
RTD Fast HWHM=10

Advanced Baseline Restoration (BLR)
BLR:ON DN:16 UP:4
Down Correction Medium
Up Correction Slow
BLR On

With the DP5, a different set of coarse gains is available.

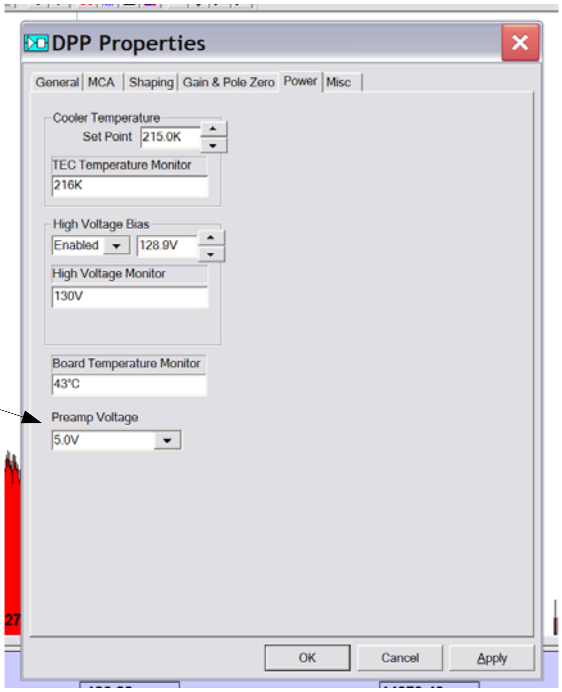


Gain
Coarse 102.01x
Fine 1.2495
Enter Gain Value

Pole Zero OFF
Input Offset 0.030V
Input Polarity Negative
Detector Reset Lockout Period Slow - 1.64mS

Positive polarity for Si-PIN and CdTe
Negative polarity for SDD

When you change the ADC clock, hit apply and then “OK”. You will then need to reset shaping time, BLR settings, and detector reset lockout period. Note that the updated values only appear if the software is acquiring data; if acquisition has stopped, “stale” values are shown.



Cooler Temperature
Set Point 215.0K

TEC Temperature Monitor 216K

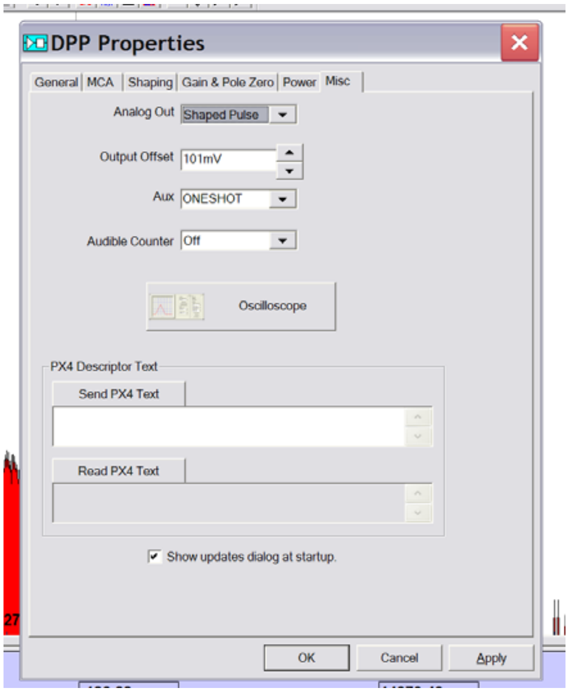
High Voltage Bias Enabled 128.9V

High Voltage Monitor 130V

Board Temperature Monitor 43°C

Preamp Voltage 5.0V

Preamp voltage cannot be set in software in the DP5



Analog Out Shaped Pulse

Output Offset 101mV

Aux ONESHOT

Audible Counter Off

PX4 Descriptor Text

☑ Show updates dialog at startup.

Note that the X-123SDD supports software control of power supplies, unlike the previous X123.

3. CHANGES TO SIGNAL PROCESSING ALGORITHMS

This section documents changes made to the signal processing algorithms used in the legacy instruments, the DP4 and PX4.

3.1 DP5 MAJOR FUNCTION BLOCKS

Figure 1 shows how the DP5 is used in the complete signal processing chain of a nuclear instrumentation system and its main function blocks. This is unchanged from the DP4. Like the DP4 and PX4, the input to the DP5 is the preamplifier output. The DP5 digitizes the preamplifier output, applies real-time digital processing to the signal, detects the peak amplitude (digitally), and bins this value in its histogram Ming memory, generating an energy spectrum. The spectrum is then transmitted over the DP5's serial interface to the user's computer. The DP5 has the same main function blocks, implementing the same basic functionality, although some of the details have been changed to improve performance.

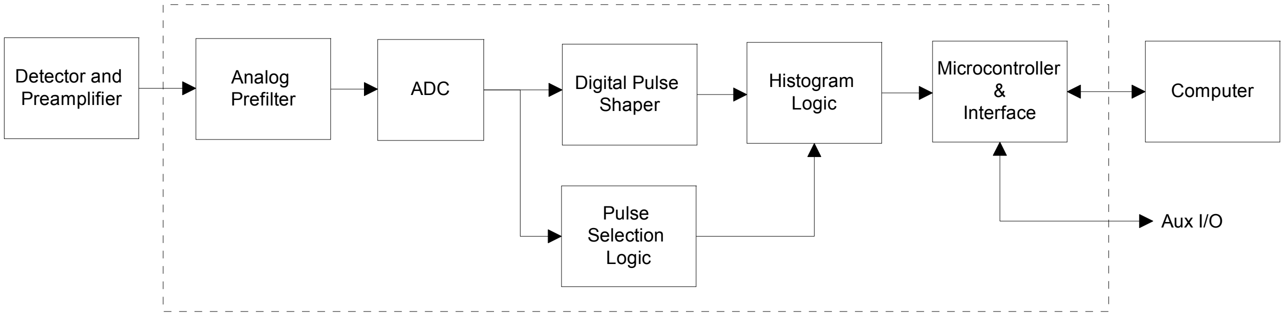


Figure 1. Block diagram of the DP5 in a complete system.

Analog Prefilter: The input to the DP5 is the output of a charge sensitive preamplifier. The analog prefilter circuit prepares this signal for accurate digitization. The main functions of this circuit are (1) applying appropriate gain, offset, and inversion (if necessary) to utilize the dynamic range of the ADC, and (2) carrying out some filtering and pulse shaping functions to optimize the digitization.

Changes to the DP5 analog prefilter include (1) adding additional coarse gain settings to provide a larger dynamic range and to provide overlapping gain ranges; (2) replacing the input offset potentiometer with a DAC for software control; (3) making the input polarity under software control; and (4) lower noise circuitry.

ADC: The ADC digitizes the output of the analog prefilter at a 20 MHz rate. This stream of digitized values is sent, in real time, into the digital pulse shaper.

The ADC can now be clocked at either 20 MHz or at 80 MHz. The higher frequency permits faster shaping times for better performance at high count rates. Like the DP4/PX4, a 12 bit ADC is used.

Digital Pulse Shaper: The ADC output is processed continuously using a pipeline architecture to generate a real time shaped pulse. This carries out pulse shaping as in any other shaping amplifier. The shaped pulse is a purely digital entity. Its output can be routed to a DAC, for diagnostic purposes, but this is not necessary. The peak value of the digital shaped pulse is determined by a peak detect circuit in the pulse shaper. The peak value for each pulse, a single digital quantity, is the primary output of the pulse shaper.

The basic slow pulse shaping is unchanged from the DP4. It is still a trapezoid, but with faster shaping times available. Some of the additional logic has been changed, including (1) improved fast channel shaping for better separation of pulses; (2) minor changes to baseline restoration logic for improved stability; (3) minor changes to peak detect logic for reduced dead time.

Pulse Selection Logic: The pulse selection logic rejects pulses for which an accurate measurement cannot be made. It includes pile-up rejection and risetime discrimination.

Pile-up rejection in the DP5 has been changed and results in better performance, i.e. fewer piled-up pulses in the output spectrum. But there are some subtle changes to the piled-up spectrum. The dead time is now measured directly rather than being estimated from the incoming count rate (ICR). Risetime discrimination is unchanged (note: it is not yet functional in the current firmware release). The DP5 will permit the user to measure the spectrum of these piled-up events, which is very useful in optimizing.

Histogram Memory: The histogram memory operates as in a traditional MCA. When a pulse occurs with a particular peak value, a counter in a corresponding memory location is incremented. The result is a histogram, an array containing, in each cell, the number of events with the corresponding peak value. This is the energy spectrum and is the primary output of the DP5. Auxiliary outputs include eight different single channel analyzers, and both a DAC output and a digital output showing pulse shapes from several points in the signal processing chain.

The basic operation is unchanged. The logic has been streamlined considerably, providing for much faster readout of the spectra, reducing dead time between acquisitions. The DP5 will include a spectrum offset parameter which can be set so that channel zero represents zero energy. The DP5 will be capable of storing and displaying additional spectra, not used during “primary” data acquisition but useful for setup and analysis, such as the spectrum of events rejected by PUR or RTD logic. The DP5 will also include a “List” mode, which stores both the amplitude and arrival time of each event.

Interface: The DP5 includes hardware and software to interface between these various functions and the user’s computer. A primary function of the interface is to transmit the spectrum to the user. The interface also controls data acquisition, by starting and stopping the processing and by clearing the histogram memory. It also controls certain aspects of the analog and digital shaping, for example setting the analog gain or the pulse shaping time.

The interface hardware and software are as compatible as feasible but have changed somewhat. These changes are documented in sections 4 and 5 of this document.

3.2 Clock Rate Changes

Several time-related settings of the digital processing are implemented in terms of the clock cycles. When the DP5 is set to 20 MHz, these operate as in the DP4. When the ADC clock is set to 80 MHz, there are some changes. First and foremost, the peaking time is defined in terms of clock cycles. If the software uses the same configuration byte for the peaking time, then the actual peaking time is a factor of four faster. The ADMCA software shows the true peaking time. Under the most common operating conditions, the electronic noise increases as short shaping times, so increasing the clock rate often increases noise. The optimum slow threshold setting usually increases with this, so the user must set this properly.

At 20 MHz, the fast channel has a 400 nsec peaking time, as in the DP4. At 80 MHz, the fast channel has a 100 nsec peaking time, equivalent to a semi-Gaussian shaper with 40 nsec time constant. This provides excellent pile-up rejection but the noise in the fast channel generally increases (due to the wider noise bandwidth) and the optimum fast channel threshold will increase as well.

The detector reset lockout interval scales with clock rate, as do the BLR settings. After the ADC clock is changed, if the user wants to keep the same interval, then the reset lockout and BLR settings should be changed appropriately.

3.3 Dead Time and Pile-Up Rejection

Most traditional analog nuclear instrumentation systems have several sources of dead time, including a time due to the duration of the shaped pulse (which is important for counting) and another due “conversion”, the time the peak height is sampled and digitized. Most traditional MCAs have a “live time clock” which is turned off during “conversion”. So this clock permits one to correct count rates for those lost during conversion. Since the “conversion” time is usually much longer than the pulse shaping time, this is the dominant correction.

Digital processors do not “lose” pulses due to conversion. Since they sample continuously, at the ADC clock speed, there is no conversion time and so no losses. However, the shaped pulses do have a finite duration and therefore some pulses are lost, i.e. not counted, due to overlap of the shaped pulses. This is the “dead time” for which one corrects in a digital processor. This “dead time per pulse” in the DP5 is different from that in the DP4 and the total dead time is measured differently. This difference has no impact on users operating at low count rates but is quite important for those operating at high count rates. In the DP4 (and PX4), there was a fixed dead time per pulse, arising from the peak detect logic, equal to

$$\left(\frac{19}{16}\right)\tau_P + \tau_F \quad \text{[DP4]}$$

where τ_P is the peaking time and τ_F is the flat top duration. To obtain the true incoming count rate, one used the measurement of the incoming count rate (ICR) in the fast channel and corrected for the known dead time of the fast channel.

In the DP5, an improved peak detect algorithm is used. One consequence is that the time required to determine if a pulse occurred (and thus the dead time) depends on the amplitude of the pulse relative to noise. For pulses much larger than noise, the dead time is essentially the peaking time. Dead time is reduced and so fewer counts are lost. However, the dead time increases for pulses near the slow threshold. The DP5's pile-up reject logic uses this same "per pulse dead time". For the vast majority of pulses, well over the noise threshold, it provides a higher throughput while still rejecting pile-up. For pulses very near the noise threshold, however, the fraction of pulses which are accepted will be reduced. To aid in understanding the pile-up behavior, the DP5 will permit the user to acquire and display the spectrum of pulses rejected by PUR and the dead time will be directly measured (these are not yet implemented in ver. 5.00).

The DP5, like the DP4 and PX4, disables data acquisition during the time a spectrum is read out over the serial interface. The "acquisition time" represents the actual time during which data are acquired and will be shorted than the elapsed "real" time. This is discussed in the DP4 documentation. Its operation is unchanged in the DP5, although the dead time is reduced due to faster data transfer logic.

4. HARDWARE MIGRATION

This section will discuss hardware changes, such as form factor, connectors, and power dissipation. Software interface changes are discussed in the next section.

4.1 X-123SDD

The X-123SDD has much better performance than the X-123 with a Si-PIN, but the interface is virtually unchanged.

- A silicon drift detector (SDD) is used as the detector rather than a Si-PIN. This has better resolution and operates at higher count rates. Information on the SDD can be found at Amptek's website at http://www.amptek.com. From an interface perspective, the optimum configuration for the SDD is different than the Si-PIN but the response and its use is otherwise unchanged.
- As discussed in more detail in the software interface section, the bias voltage and temperature set point in the X-123 were set at the factory by resistors or trim pots. In the X-123SDD these are under software control, providing additional flexibility in configuring the unit.
- The physical dimensions, mass, and mounting points of the X-123 package are unchanged.
- The USB and serial connectors are the same, with locations moved slightly, as shown below.
- A different power connector is used, still supplying +5 VDC. The new connector provides a positive mechanical connection.
- An Ethernet connector will be available in the next product release (summer '09).
- An "auxiliary" connector has been added. This brings out several digital I/O signals, including a timing output, gate input, SCA outputs, and other diagnostics.
- Power dissipation has increased from about 1 watt to approximately 1.5 watts.

4.1.1 Block Diagram

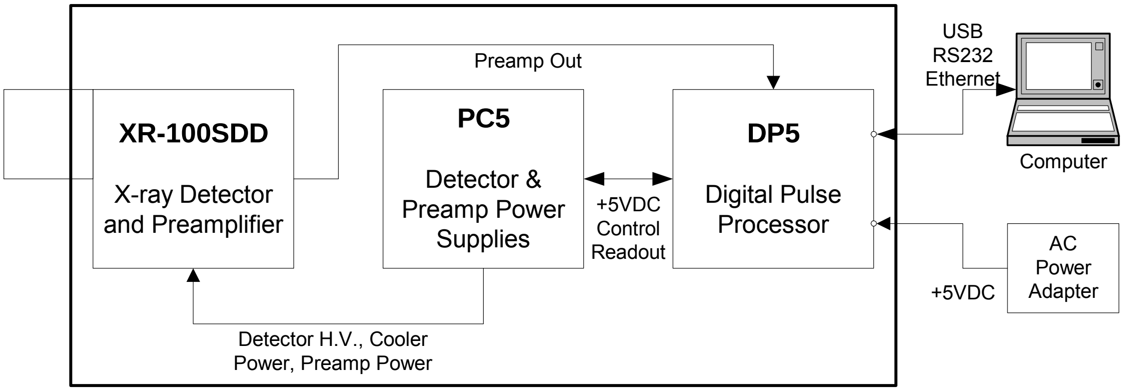


Figure 2. X-123SDD Block Diagram.

4.1.2 Connectors

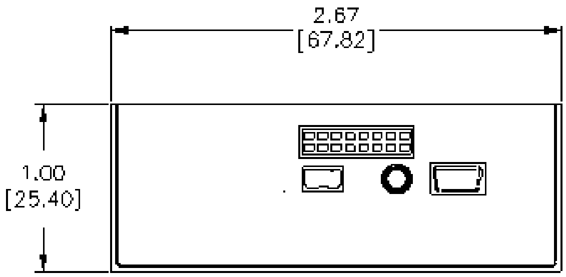


Figure 3. Mechanical drawing showing connector locations on the X-123SDD. All other mechanical dimensions are unchanged from the standard X-123.

Power

Hirose MQ172-3PA(55).

Auxiliary

2x8 16-pin 2 mm spacing (Samtec part number ASP-135096-01)

Ethernet

Standard Ethernet connector (available summer ’09)

USB

Standard USB mini jack

RS-232

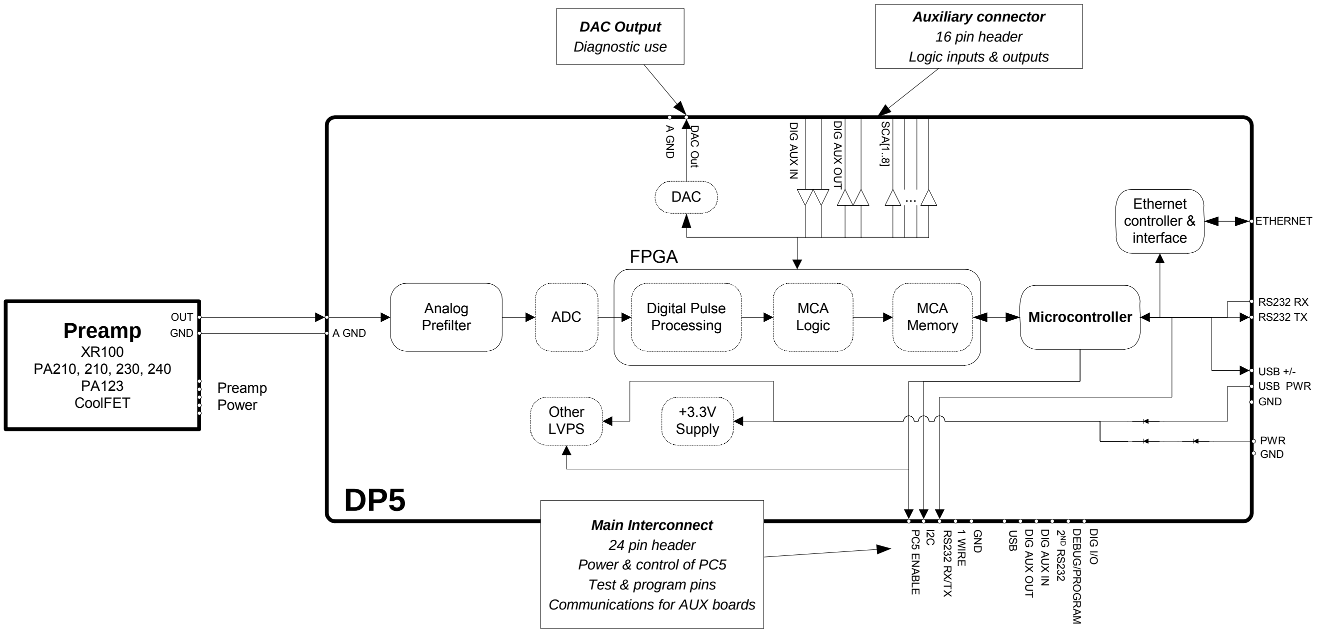
Standard 2.5 mm stereo audio jack.

4.2 DP5 ALONE

- The overall envelope and locations of the mounting points on the DP5 are unchanged, but the board length has been reduced slightly.
- Power
  - The DP5 includes low voltage power supplies. It is powered from a loosely regulated +5 VDC source and generates all of the necessary low voltages. This is different from the DP4, which requires several regulated low voltages.
  - Power dissipation has increased from 600 mW (typical) to 1 W (typical)
- Communications

  - The serial (RS232) interface now operates at 115k, but 57.6k can still be selected via a ‘boot flag’. (See section 5.2.1.)
  - It supports USB 2.0 (full speed, with the same data transfer rates as on the DP4)
  - It has support for Ethernet (summer ‘09)
- Connectors
  - The signal input connector (which connects to a preamp) and the output DAC connector are in the same locations but have changed. The input connector is now a 1x3 right angle header Molex part number 22-28-8032. The DAC output connector is a 1x2 right angle header Molex part number 22-28-8022.
  - The standard configuration for the input circuit is no longer differential and has a +/-4.5V range.
  - Communications interface and power connectors are now located on the DP5 board. Previously, a header on the DP4 took these lines, via a ribbon cable, to a daughter board. The daughter board is no longer required.
  - The “main interconnect” between the DP5 and any additional boards, such as the PC5, is completely different. It is a new connector, with new signals, and a new pinout.
  - The “auxiliary connector,” which brings auxiliary digital inputs/outputs, has been moved.
- Other
  - A new microcontroller is now used, the Silicon Labs 8051F340. The embedded software has been completely rewritten, now in C instead of assembler.

4.2.1 Block Diagram



4.2.2 DP5 Board Layout

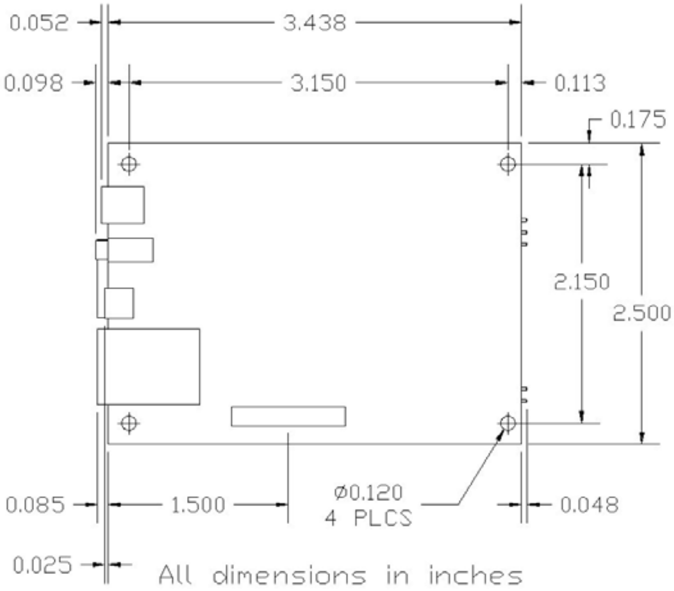


Figure 4. DP5 bottom side, showing key interconnects.

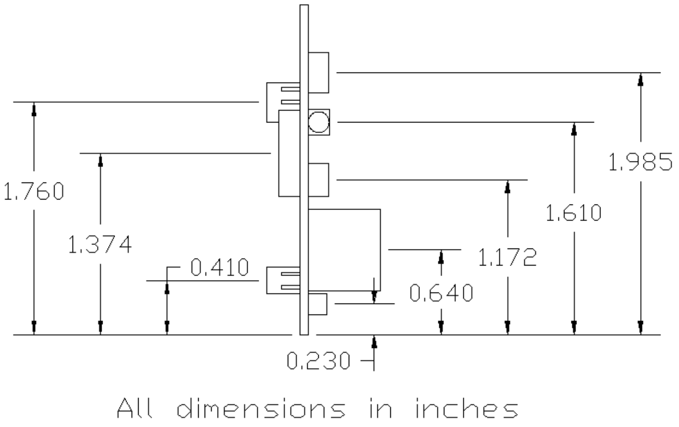


Figure 5. DP5 back/side view.

4.2.3 J5 Main Interconnect

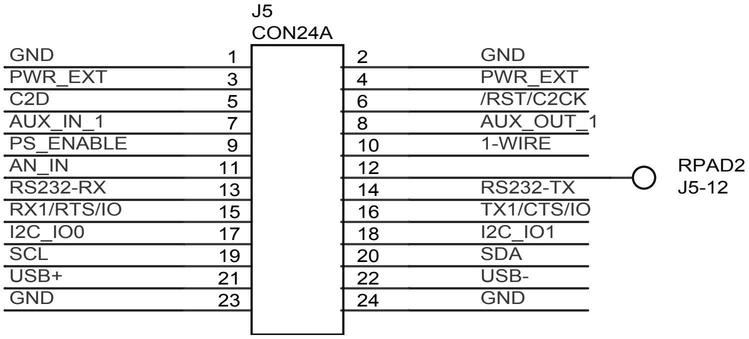
This connector replaces the JP6 ribbon cable interconnect on the DP4. The primary purpose of J5 is to connect the DP5 to the PC5, Amptek's power supply board for Amptek preamplifiers and detectors. It provides input power in either direction, from DP5 to PC5 or vice-versa and permits the DP5 to control PC5 supplies. Its secondary purpose is to permit addition of planned auxiliary boards such as Bluetooth, WiFi, or X-ray tube controllers. These can connect to the DP5 or to the DP5/PC5 stack. The connector is a 2X12 Socket Strip Samtec part number SMM-112-02-S-D-LC.

Pin #	Name	Use	Pin #	Name	Use
1	GND		13	RS232-RX	PC5, AUX, Test
2	GND		14	RS232-TX	PC5, AUX, Test
3	PWR EXT	PC5, AUX, Test	15	RX1/IO	AUX only
4	PWR EXT	PC5, AUX, Test	16	TX1/IO	AUX only
5	C2D	Test	17	IO0	AUX only
6	/RST/C2CK	Test	18	IO1	AUX only
7	AUX IN 1	AUX only	19	I^2C SCL	PC5, AUX, Test
8	AUX OUT 1	AUX only	20	I^2C SDA	PC5, AUX, Test
9	PS ENABLE	PC5 only	21	USB+	AUX only
10	1-WIRE	PC5 only	22	USB-	AUX only
11	AN_IN	AUX only	23	GND	
12	SPARE		24	GND	

- PWR EXT: This is the power connection between the DP5 and the PC5, the input to the regulators on both boards. It can serve as an input to the DP5 (if the external supply is connected to the PC5) or as an output from the DP5 (if the supply is connected to the DP5). It is nominally +5V (4-6V)
- C2D and /RST/C2CK: C2D and C2CK are used when initially programming the μC. Additionally, pulling pin 6 low will hold the entire DP5 in reset. Floating it or pulling it high allows normal operation. These pins are usually only used during manufacturing test.
- AUX IN 1 and AUX OUT 1: These are auxiliary digital logic input and output lines from the FPGA. They can be software configured to any one of several functions (ICR, RESET, GATE, etc). They are driven from (into) Schottky buffers (3.3 V logic) with 49.9Ω series on AUX OUT (100 kΩ to GND on AUX IN). These are not used in the PC5, only with AUX boards. These signals also appear on J6.
- PS ENABLE: Open-drain output used by the DP5 to enable the PC5 power supplies. This is only used with the PC5. It is pulled low to turn OFF the PC5 supplies, and floated to turn ON the PC5 supplies. (The PC5 has a 10k pullup to 3.3V on this signal.)
- 1-WIRE: Dallas 1-wire interface, used to communicate with 1-wire PROM in the detector hybrids. This is only used with the PC5.
- RS232-RX, RS232-TX: These are RS232 signals for interfacing the DP4 to a host system. The signal names are relative to the DP5, i.e. the DP5 receives on the RX pin and transmits on the TX pin. This may be used with the PC5 and with other units.
- RX1/IO and TX1/IO: These two pins can be assigned (via software) to one of two functions. This requires custom embedded software in the DP5 μC. They are not used in the PC5.
  - RX1/TX1: This is a second serial interface, which can be used by the μC to communicate with auxiliary devices.
  - IO: These become general purpose IO pins from the μC to interface with external hardware.

- IO0, IO1: These two pins connect to a I2C port expander and then to the µC. They can be used to communicate with or control external hardware. They can be controlled via serial commands. They are not used in the PC5.
- AN_IN: This is an uncommitted analog input. The DP5 microcontroller ADC periodically converts this signal, and the result appears in the Status Packet. The input range is 0 - 2.4V. This signal has a high input impedance.
- SDA, SCL: These are the I^{2}C bus signals. The DP5 µC is the bus master, though it is possible to communicate with another master, or other slave peripherals. They are used in the PC5 and may be used with Aux.
- USB+, USB-: This is the USB (Universal Serial Bus) communication bus. The DP5 microcontroller contains a USB communication core. They are not used in the PC5.

Pinout



Physical

- Mates with PCB mount connector (Samtec P/N) TMM-112-01-S-D-SM or with cable (Samtec P/N) TCMD-12-…
- This is a 24 pin connector, with 2 mm spacing.

4.2.4 J6 Auxiliary Connector

The primary purpose of this connector is to bring out logic signals which are not required for the primary use of the DP5: acquiring spectra and transmitting then over the serial interface. These are generally “low level” logic signal associated with each pulse processed by the DP5. They are primarily used for synchronizing the DP5 data acquisition to external hardware and for direct counter/timer outputs from the DP5. The signals are described below. The connector is a 2x8 right angle Samtec part number ASP-135096-01.

Pin #	Name	Pin #	Name
1	SCA1	9	AUX_IN_1
2	SCA2	10	AUX_OUT_1
3	SCA3	11	AUX_IN_2
4	SCA4	12	AUX_OUT_2
5	SCA5	13	IO2
6	SCA6	14	IO3
7	SCA7	15	GND
8	SCA8	16	GND

Physical

Mates with connector (Samtec P/N) TCSD-08-S(D)-xx.xx-01-F-N.

Single Channel Analyzers (SCAs)

- Each of the eight SCAs has an independently assignable LLD and a ULD. If the shaped pulse peaks with the range of an SCA, between its LLD and ULD, then a logic signal is output.
- These outputs produce pulses 100ns wide. Logic low is <0.1V @100 µA. Logic high is >3.1V @100 µA.

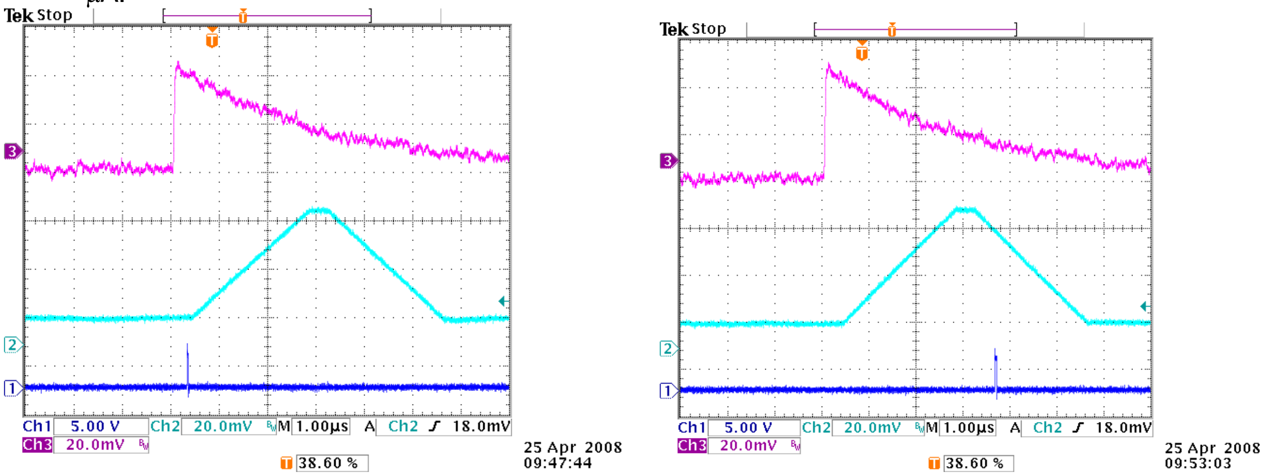


Figure 6. Oscilloscope trace showing the timing of several outputs. In both plots, the pink trace is the input to the ADC. The light blue trace is the shaped output, with a 2.4 µsec peaking time and a 0.4 µsec flat top. In the left-hand plot, the dark blue trace shows the ICR signal on an AUX_OUT. In the right-hand plot, the dark blue trace shows an SCA signal.

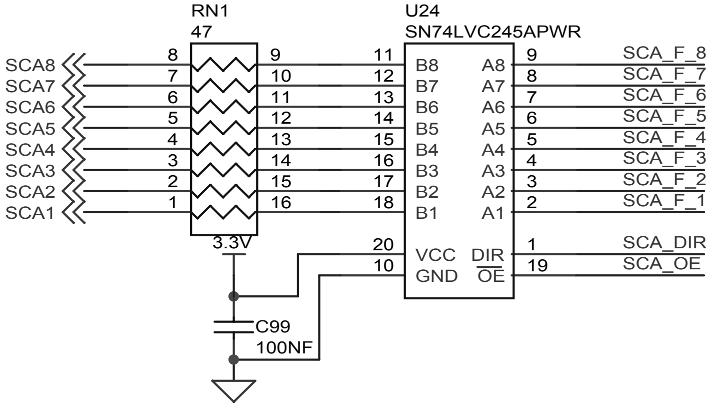
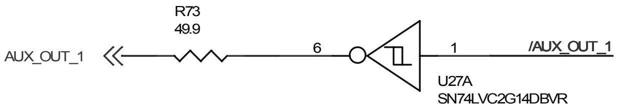
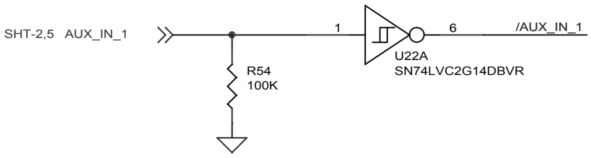


Figure 7. Schematic showing SCA interface hardware.

AUX_OUT_1 and _2

- Each of these two lines can be configured, via software, to output any one of several logic signals in the FPGA. These logic signals are associated with pulses processed by the FPGA.
- The pulse timing and duration depends on which output is commanded.
- AUX_OUT_1 is connected to the Interconnect (J5), to the auxiliary connector (J6), and can be jumpered to the stereo jack (J10). AUX_OUT_2 is only connected to the auxiliary connector (J6).
- Logic low is <0.1V. Logic high is >3.1V.





AUX_IN_1 and _2

- Each of these two lines can be configured, via software, to input any one of several logic signals in the FPGA. These logic signals are associated with pulses processed by the FPGA.
- AUX_IN_1 is connected to the Interconnect (J5), to the auxiliary connector (J6), and can be jumpered to the stereo jack (J10). AUX_IN_2 is only connected to the auxiliary connector (J6).

4.2.5 Configuration Notes

The DP5 has several configuration options, which the user should select or can change.

Input Range

In the default configuration, the input range of the DP5 is +/-4.5V. This is suitable for most charge amps with resistive or transistor feedback or reset preamps operating from +/- 5V. For reset preamps operating from a larger range, including Amptek's XR100 detectors, a zero ohm resistor should be installed in R50.

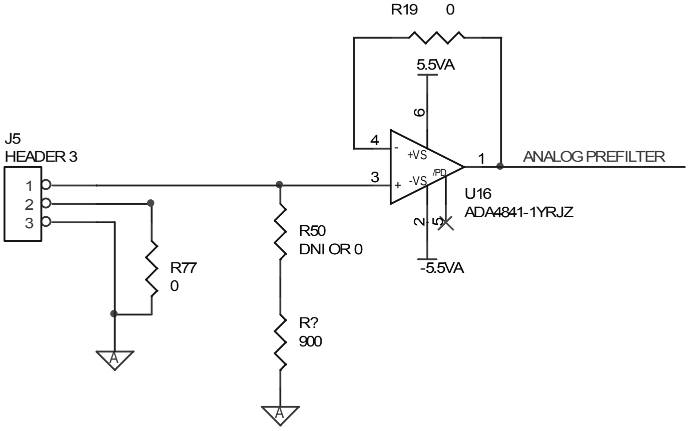


Figure 8. Schematic of DP5 analog input.

Differential Input

The DP5's standard input configuration, shown in Figure 8, is not differential. This is different from the DP4 for which the differential input was standard. The DP5 can be configured in a differential mode. This increases immunity to conducted interference, such as ground loops, but also increases the electronic noise contribution of the DP5's analog circuitry.

Resistive Feedback Preamps

The DP5's standard input configuration is for a reset preamplifier, with no pole zero resistor. The pole zero circuit changes are essentially the same as for the DP4, as discussed in its user manual, but the reference designators are different. The differentiating capacitor is C69, 6800 pF. The pole zero pot is R102, with nothing installed for the standard configuration.

1. Install in R102 a pot such that (R102)(C69) can be adjusted to give the preamp's time constant.
2. Remove R26.
3. Install a pot in R103. This pot will be in series with 411 ohms. The value of this pot should be such that the parallel combination of R102 and (R103 + 411 ohms) equals 460 ohms.

4.3 DP5 WITH PC5

The DP5 and PC5 can be used with Amptek's family of detectors and preamplifiers. It is appropriate for OEM use in a larger system.

The key points for this system are essentially the same as discussed above with the X123 and with the DP5. As in prior systems, Amptek has several different OEM preamplifiers which are electrically equivalent to the XR100 family but have different form factors for mounting in OEM sensor assemblies.

- The PC4-3's primary interface was via an 8 pin connector which included power in, RS232 signals, and power control. The PC5 retains the same 8 pin connector, with the same location and pinout.
- Mounting holes and connector locations are unchanged. The boards are 62 mil shorter.
- The primary difference is in control, discussed below in the software migration section. In the PC4-3, high voltage bias and temperature set point were controlled via resistors or pots, usually set at the factory. They are now under software control and can be readout via ADCs. The control of these is like the control previously available in the PX4. For users with legacy software written for the DP4, these set points can be sent once in the PC5 and then stored in the DP5 EEPROM. After that, legacy DP4 code will function normally. Alternately, a user can modify code to support the DP5 and have full software control over these parameters.
- The PC5 can be delivered with either a positive or negative HV supply, and preamp supplies of either +/-5V, or +/-8.5V. These are set by jumpers (zero ohm resistors soldered on the board) so are not under software control.

Block Diagram

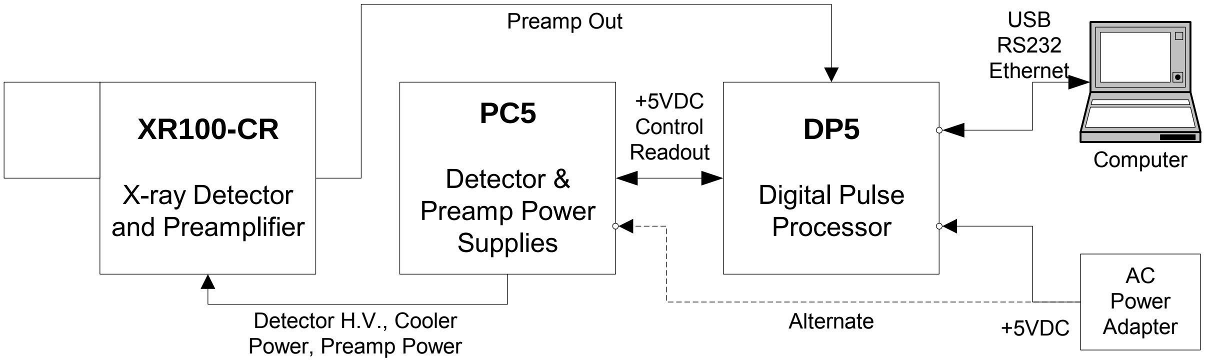


Figure 9. Block diagram of DP5/PC5 system.

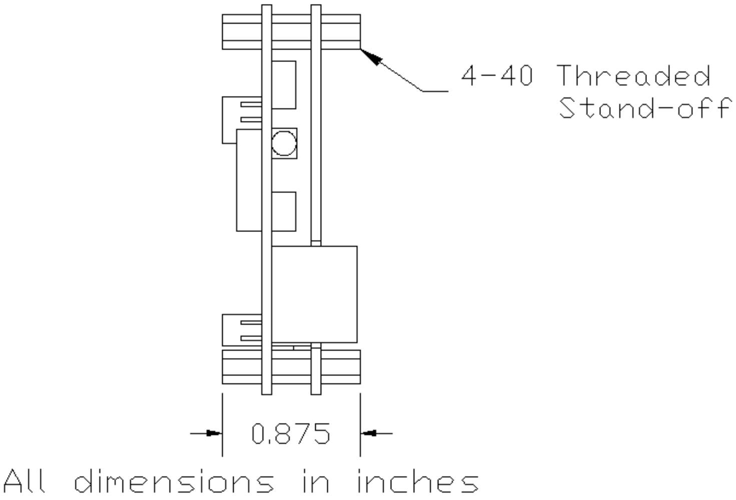


Figure 10. DP5/PC5 stack.

5. SOFTWARE MIGRATION

5.1 OVERVIEW

The DP5's software interface is very similar to that of the DP4 and PX4 and supports legacy software written for the older products. This document is intended to describe only the changes to the interface and assumes the user is familiar with the older software, documented elsewhere.

For lab users, Amptek recommends the use of the ADMCA software package. One can now select the “DP5” from the pulldown menu and gain access to all of the DP5’s features without going into the low level details described later in this document. For OEMs using their own software, it will be necessary to understand how the DP5 is configured and the changes which have been made to the data packets. OEMs and custom user ahouls use the DPP API when writing custom code in Windows. The advanced programming and low-level discussion later in this document is intended for user that want to write their own drivers, or are on a non-Windows platform.

This document applies to release FW 5.03 of the firmware and embedded software, which is the first production release. Additional features will be implemented in future releases. Users will be able to download the new releases from Amptek’s website and program their DP5s in the field. Section 5.8 tells how to upload new releases.

5.1.1 Configuring the DP5

The DP5’s configuration parameters are a superset of those found in the PX4, which is a superset of those found in the DP4. The DP4 has the minimum parameters (peaking time, MCA channels, etc). The PX4 includes these plus others such as detector HV bias, TEC temperature, and input polarity. The DP5 includes all of these plus additional parameters such as ADC clock rate. To provide backward compatibility with legacy software the DP5 handles these three classes of parameters quite differently.

The DP4 and PX4 use the same configuration data packets; the DP4 simply ignores the additional parameters which control options not available in the DP4 hardware. For backward compatibility, the DP5 recognizes the configuration packets of either the DP4 or the PX4, operating in a “DP4 emulation mode” or a “PX4 emulation mode. Since it needs all the parameters to operate properly, it reads the additional parameters from nonvolatile memory. These must be set via a new data packet, after which legacy software may be used. Some critical things to note are:

1. The new configuration options not available with either the DP4 or the PX4 are termed “boot flags,” and are set via a new data packet and stored in EEPROM in the DP5.
    a. One important “boot” parameter sets whether the DP5 is in “DP4 emulation mode” or “PX4 emulation mode” In “PX4 emulation mode,” the user can command the HV bias, temperature for the cooler, etc. In “DP4 emulation mode,” the DP5 reads these from nonvolatile memory. [ADMCA requires the ‘PX4 emulation mode,’ and will switch the DP5 to that mode upon connection, if necessary.]
    b. Other boot parameters set the ADC clock frequency, the RS232 baud rate, etc.
    c. These boot parameters are set via USB, using either a “Boot Manager” provided by Amptek or by customer written software. ADMCA can set the boot flags as well.
2. Some of the configuration parameters have a different meaning in the DP5 and the meaning may depend on the boot options. For example, the coarse gain settings of the DP5 are different from either DP4 or PX4. In addition, as a boot option one can set the ADC clock to either 20 MHz or 80 MHz, and this will change the peaking time settings (the same configuration byte will lead to a peaking time which differs by a factor of four). The details are in section 5.2 of this document.

5.1.2 Data From the DP5

The DP4 and PX4 provide four data packet types:

- The spectral data packets are unchanged from the DP4/PX4.
- The status packet has several additional parameters, placed into bytes not used with the DP4 or PX4. In addition, some parameters have been redefined Note: The status packet indicates if the device is a DP4, PX4, DP5 in “DP4 emulation mode”, or DP5 in “PX4 emulation mode”.
- The configuration readback packet works the same as the PX4/DP4. It will return the PX4/DP4 configuration packet most recently received by the DP5. Configuration readback over USB now operates on a different USB pipe, as discussed elsewhere.
- The oscilloscope data shares USB Pipe 3 with the configuration readback, as described in Section 5.3.

- The DP5 can store up to 512 bytes of miscellaneous data (text or binary) in nonvolatile memory, which is also read back via USB Pipe 3. The data is sent using 'USB Vendor Requests' as discussed later.

5.1.3 DP5 Buffering

The DP5 uses an improved technique to greatly reduce MCA deadtime associated with the readout of spectral data. The MCA memory in which the spectrum is acquired is internal to the FPGA. When the spectral data is requested, the MCA is disabled, and a high-speed hardware sequencer copies the spectral data to an external SRAM, simultaneously clearing the MCA memory if required. The MCA is then re-enabled and can continue to acquire data while the buffered spectrum is sent out.

The desired compatibility with the existing DP4 and PX4 protocols makes it difficult to use these new features in the ideal manner. For example, the sequencer that copies the spectral data can also simultaneously zero it, but only if it is configured to do so at the start of the transfer. With the existing USB protocol, it isn't known whether to zero the data until the transfer is complete. The net result is that for repetitive measurements, unnecessary deadtime is introduced. An extension to the protocol will be added to fix this in the next release of the firmware. The details of how this works are a bit different for USB vs. RS232, and are discussed below.

USB

Because of a limitation in the USB implementation in the DP5 microcontroller relative to that of the DP4 microcontroller, the DP5 USB buffering can't behave exactly the same way as the DP4. The DP4/PX4 use the initiation of a USB IN request (i.e. DP5 to PC host) as a trigger to buffer fresh data, and to start sending it immediately. The DP5 microcontroller lacks the ability to detect the initiation of an IN request (it only detects the completion), and thus can't work the same way as the DP4/PX4.

The solution is the use of USB retries as a mechanism for the DP5 to detect the start of an IN transfer. **When the application requests data from one of the IN pipes (IN1=status; IN2=spectrum; IN3=configuration readback/oscilloscope data/misc data), the DP5 will immediately return a packet of length = 0.** (The USB specification refers to this as a zero-length packet, or ZLP, and uses a ZLP to terminate a transfer.) The DP5 detects this transaction, and immediately fills the USB buffer with fresh data. **The application should detect that the first attempt returned 0 bytes, and should immediately retry the request. The 2nd attempt will return the correct data.** [Users of the DPPAPI won't need to worry about this – an updated version of the DLL will handle it transparently.**]**

Additionally, the relationship between the spectral and status data is slightly different. With the DP4/PX4, status packets were always 'live' – they would take a snapshot of the current state of the acquisition. In the DP5, a status packet is created at the same instant that the spectral data is buffered, so that the spectral & status data are synchronized. The first status read after the spectrum is read will return a status packet that is perfectly synchronized with the spectral data which was already read. For any subsequent status reads, the data will be "live", i.e. a new snapshot of status data is taken the instant the request is received by the DP5. Status packets will remain "live" until the next spectrum read, at which point a new synchronized status packet is generated, to be used for the first status read after the spectrum is read. [This isn't much different than the DP4/PX4. In the DP4/PX4, the spectrum and status data become synchronized (and static) after the acquisition ends (a preset expires, etc.) but there may be a bit of skew between them for intermediate reads.]

The following deadtimes due to USB communication were measured. The deadtime depends on the FPGA clock speed (20 or 80 MHz). There is an additional deadtime penalty if the spectrum is being cleared after it is read out – the time quoted below is the actual time the MCA is disabled to clear the spectrum. In reality, this additional deadtime includes the time between the spectrum being read out over USB, and the receipt of the USB command to clear the spectrum. [This time would depend on the host PC and amount of USB traffic, and is likely in the 5-10mS range.] This additional deadtime will be eliminated with the use of the protocol extension mentioned above.

# of MCA channels selected	FPGA clock = 20MHz	FPGA clock = 80MHz
256	240uS	124uS
512	431uS	200uS
1024	815uS	354uS
2048	1.58mS	661uS

4096	3.12mS	1.28mS
8192	6.19mS	2.50mS
Time for spectrum clear	415uS	105uS

RS232

The RS232 protocol used by the DP4/PX4 can be used unmodified with the DP5. [The boot flags must be set to the proper baud rate: 57.6k or 115.2k, as documented below. The DP4/PX4 has a fixed baud rate of 57.6k.] . With the use of protocol extensions for the DP5, RS232 deadtimes similar to the USB deadtimes discussed above can be expected.

The RS232 data request packet consists of a sync byte, requested packet number, and end of packet byte (this is the same as the DP4/PX4 data request):

Data request packet (3 bytes):

Byte 1	0xFD (sync)
Byte 2	requested packet number
Byte 3	0xFF (end of packet request)

Previously, any RS232 data request resulted in 256 bytes being returned. The DP5 adds several protocol extensions which allow a single request to result in the entire data set being returned: the spectrum, status and configuration packet.

Function	Request packet number
Request entire data set	0x65
Request entire data set plus Clear Spectrum	0x66

The amount of data generated by the 'Request Entire Data Set…' commands depends on how many MCA channels are currently selected. The data set consists of spectral data (768, 1536, 3072, 6144, 12288 or 24576 bytes), followed by the 64-byte status packet and ending with a 64-byte packet reflecting the current configuration.

The spectral data is 3 bytes/channel (LSB to MSB), going from lowest channel to highest channel. The 64 bytes of status data immediately follow the spectral data, and its format is discussed later. The final 64 bytes is the current configuration, and its format is the same as the configuration packet, also described later. The following table shows the structure of the data returned by the 'Request Entire Data Set…' commands:

# of Channels	Total Data Returned	Spectral Data	Status Data	Configuration Data
256	896 bytes	Bytes 0-767	Bytes 768-831	Bytes 832-895
512	1664 bytes	Bytes 0-1535	Bytes 1536-1599	Bytes 1600-1663
1024	3200 bytes	Bytes 0-3071	Bytes 3072-3135	Bytes 3136-3199
2048	6272 bytes	Bytes 0-6143	Bytes 6144-6207	Bytes 6208-6271
4096	12416 bytes	Bytes 0-12287	Bytes 12288-12351	Bytes 12352-12415
8192	24704 bytes	Bytes 0-24575	Bytes 24576-24639	Bytes 24640-24703

5.2 Changes to Configuration Packets

5.2.1 Boot Configuration ("Boot Flags")

This table shows the "boot flags", which are recognized only by the DP5. These values can only be sent via USB vendor requests, in the initial firmware release (FW 5.03). Most of these values are true boot options so only take effect on reset. The exceptions are noted as "live" in the table below. "Live" options will take effect when the next configuration packet is received by the DP5.

This table defines the individual bits of the 16-bit value field of the Vendor Request. All undefined bits should be regarded as "reserved" **and should be set to 0**. Note: the MSB bits (bits D15-D8) are volatile; they all default to 0 at power-up. The LSB bits (bits D7-D0) are non-volatile, and are restored to their commanded state at power-up.

Bit	Val	Description
D15-D10	0	Currently unused
D9	0	MCA source: Slow channel (normal operation):
	1	MCA source: Fast channel. Sets the source for the MCA to the fast channel. Note that the fast channel does not have BLR or a pole-zero.
D8	0	Peak detect: Max only (normal operation)
	1	Max and min: Allows the peak detect (and MCA) to capture local minima of the shaped pulse stream, in addition to local maxima. Primary use is for capturing the 'noise Gaussian' for determining the true spectrum zero.
D7	0	PX4 emulation mode (use PX4 parameters, reports 'PX4' in status)
	1	DP4 emulation mode (use nonvolatile HV & cooler parameters; report 'DP4' status)
D6	0	Boot in unconfigured state
	1	Use nonvolatile configuration, report 'power button config' in status. Equivalent to "holding the power button until the second beep" in PX4.
D5	0	57,600 baud for RS232. Used in legacy units.
	1	115,200 baud
D4	0	20MHz FPGA clock. Used in the legacy units. This parameter is "live".
	1	80MHz FPGA clock. Provides faster peaking times to operate at higher count rates.
D3	0	Negative HV PC5 required
	1	Positive HV PC5 required
D2	0	Positive pulse input to DP5, i.e. SDD [DP4 emulation mode only; use 'invert' bit in PX4 emulation mode]
	1	Negative pulse input to DP5, i.e. XR100 [" "]
D1	0	Use spectrum offset of 0. This parameter is "live".
	1	Use spectrum offset as programmed via boot option
D0	0	Normal fast channel peaking time (400nS @ 20MHz clock, 100nS @ 80MHz)
	1	4x slower fast channel peaking time (1.6uS @ 20MHz clock, 400nS @ 80MHz)

Notes

- ADMCA always uses "PX4 emulation mode". If ADMCA connects to a DP5 and finds it in "DP4 emulation mode", then it will send the command to change emulation mode and will then instruct the user to reboot the DP5 before proceeding.
- The "DP4 emulation mode" is intended only for use to support legacy applications and software. In this case, use the boot manager to set the mode and the options.

  High Voltage Polarity: The legacy power supplies only had a positive bias available. The new PC5 power supply can be configured for negative polarity, by setting jumpers on the board. The DP5 checks the jumper settings before enabling any supplies on the PC5. Setting this bit does not change the supply polarity; it tells the DP5 which polarity is required before applying power.

  NOTE: Applying the incorrect polarity may permanently damage a detector.

- Flags D4 (FPGA clock) and D1 (spectrum offset) are “live” - they take effect immediately without rebooting the DP5.

5.2.2 Changes to Standard Configuration Packet

This table shows the “standard configuration packet”, which is recognized by the DP4, the PX4, and the DP5. These values can be sent via RS232 or USB. The CLK_FAC equals 1 for 20 MHz ADC and 4 for 80 MHZ ADC.

Offset	Data bits	Allowed Value	Description	DP5 Changes
0	D7	0	0=Normal reset lockout, 1=fast reset lockout	
	D6-D3	t=0…15	Flat top width=200nS*(t+1)*2^d	Scales with ADC clock but otherwise same. See Table 2. (200nS/CLK_FAC)*(t+1)*2^d
	D2-D0	d=0…4	Decimation factor=2^d	
1	D7-D0	0…255	Slow channel threshold	
2	D7-D0	0…255	Fast channel threshold	Still set the same way but optimum setting will depend on fast channel peaking time, which depends on ADC clock. See section 3.2.
3	D7-D1	-64…+63	Output DAC offset (signed) (-500mV to +492mV)	
	D0	0	DAC disabled	
		1	DAC enabled	
4	D7-D6	0	normal operation	
	D5	0	MCA disabled	
		1	MCA enabled	
	D4-D2	0	MCA/MCS 4096 channel mode	
		1	MCA/MCS 2048 channel mode	
		2	MCA/MCS 1024 channel mode	
		3	MCA/MCS 512 channel mode	
		4	MCA/MCS 256 channel mode	
		5	MCA/MCS 8192 channel mode	
	D1-D0	0	DAC output=fast channel	
		1	DAC output=shaped pulse	
		2	DAC output=decimated input	
		3	DAC output= Test Pulser	
5	D7-D0	1…255	Pileup Reject interval: optimal setting is (p * 19 + (t + 1) * 4) / 2	Not used in new pile-up reject logic – PUR is enabled for any nonzero value. See section 3.3.
		0	Pileup Reject disabled	Disable is still used as before.
6	D7-D4	p=1…8	Peaking Time=800nS*p*2^d (See Config byte 0: bits D2-D0, and also Table 2.	Scales with ADC clock but otherwise same. See Table 2. (800nS/CLK_FAC)* p*2^d
	D3-D2	0	Detector reset lockout period=13.11mS (slow) or 819uS (fast) (see Config byte 0: bit D0)	Lockout period reduced by CLK_FAC
		1	Detector reset lockout period=6.55mS or 410uS	
		2	Detector reset lockout period=3.28mS or 205uS	
		3	Detector reset lockout period=1.64mS or 102uS	
	D1	0	No auto baseline reset during detector reset	Autobaseline replaced by BLR. This bit is reserved in FW 5.03 and should be set to 0.
		1	Auto baseline reset during detector reset	
	D0	0	Disable MCA during detector reset	

		1	Do not disable MCA during detector reset	
7	D7-D0	0…255	RTD slow threshold	This is the 8-bit RTD sensitivity setting, similar to the DP4/PX4 'RTD Time Threshold'
8	D7	1	Normal operation	
	D6	0	(Reserved)	
	D5	0-1	Analog gain control A (SeeTable 1.)	New gain table. See Table 1
	D4	0	RTD off	
		1	RTD on	
	D3-D0	0…15	RTD time threshold	This 4-bit setting is now the RTD slow threshold: 15=50% full-scale
9	D7	0	50% digital attenuation	
		1	Normal operation	
	D6	0	0=Baseline Restoration (BLR) off	
		1	BLR enabled	
	D5-D4	0	BLR Down correction=very slow	These essentially set the slew rate of the BLR correction. Changing the clock rate increases the slew for the same setting: a BLR down of "medium" at 20 MHz will produce the same slew as "slow" at 80 MHz.
		1	BLR Down correction=slow	
		2	BLR Down correction=medium	
		3	BLR Down correction=fast	
	D3-D2	0	BLR Up correction=very slow	
		1	BLR Up correction=slow	
		2	BLR Up correction=medium	
		3	BLR Up correction=fast	
	D1-D0	0	BLR correction threshold=very fast	The new algorithm fixes the threshold at "Very Fast". This parameter is no longer adjustable.
		1	BLR correction threshold=fast	
		2	BLR correction threshold=normal	
		3	BLR correction threshold=slow	
10	D7-D6	0	GATE off	The DP5 uses the 'AUX_IN1' signal for GATE, "Active high" means events are passed when the 'AUX_IN1' input is low, and events are gated off when it is high.
		1	N/A (Gate off)	
		2	GATE on, polarity active high	
		3	GATE on, polarity active low	
	D5-D4	0	Software MCA buffer select, buffer A	The DP5 does not support buffer B so these bits are "reserved".
		1	Software MCA buffer select, buffer B	
		2	Hardware MCA buffer select	
		3	N/A	
	D3	0	Digital scope trigger = rising edge	
		1	Digital scope trigger = falling edge	
	D2-D0	0	AUX_OUT=ICR	
		1	AUX_OUT=PILEUP	
		2	AUX_OUT=MCS_TIMEBASE	
		3	AUX_OUT=ONESHOT	
		4	AUX_OUT=DET_RES (active low)	
		5	AUX_OUT=MCA_EN	
		6	AUX_OUT=TRIGGER	
		7	AUX_OUT=SCA8	
11	D7-D0	0-255	Preset time LSB	
12	D7-D0	0-255	Preset time Byte2	
13	D7-D0	0-255	Preset time MSB [Time=0.1 sec * [MSB*(2^16)+Byte2*(2^8)+LSB]	

Note 1: These parameters do nothing in FW 5.03; they are ignored. Future releases will use them as in the DP4 and PX4.

Analog Gain Control A	Analog Gain Control B		
(Config byte 8, bit D5)	(Config byte 15, bits D3-D0)	PX4 Gain	DP5 Gain
1	0	4.13	3.78
1	1	4.95	5.26
1	2	5.94	6.56
1	3	7.17	6.56
0	0	8.22	8.39
0	1	9.84	10.10
0	2	11.80	11.31
0	3	14.30	14.56
1	4	17.50	17.77
1	5	20.90	22.42
1	6	25.10	22.42
1	7	30.30	30.83
0	4	34.70	38.18
0	5	41.60	38.18
0	6	49.90	47.47
0	7	60.30	66.26
0	8	76.70	66.26
0	9	91.90	102.01
0	10	110.00	102.01
0	11	133.00	102.01
1	12	163.00	102.01
1	13	195.00	102.01
1	14	234.00	102.01
1	15	283.00	102.01
0	12	324.00	102.01
0	13	388.00	102.01
0	14	466.00	1.00
0	15	563.00	2.22

Table 1. Gain settings in the PX4 and the DP5. Note that the DP5 has a slightly different gain range.

20 MHz Clock		**80 MHz Clock**					
Peaking Time (μS)	**Flat Top (μS)**	**Peaking Time (μS)**	**Flat Top (μS)**	**Decimation**	**'Rise' register**	**Flat Top register**	**Decimation register**
0.8	0.2...3.2	0.2	0.05…0.8	1	1	0…15	0
1.6	0.2...3.2	0.4	0.05…0.8	1	2	0…15	0
2.4	0.2...3.2	0.6	0.05…0.8	1	3	0…15	0
3.2	0.2...3.2	0.8	0.05…0.8	1	4	0…15	0
4.0	0.2...3.2	1.0	0.05…0.8	1	5	0…15	0
4.8	0.2...3.2	1.2	0.05…0.8	1	6	0…15	0
5.6	0.2...3.2	1.4	0.05…0.8	1	7	0…15	0
6.4	0.2...3.2	1.6	0.05…0.8	1	8	0…15	0
8.0	0.4...6.4	2.0	0.1…1.6	2	5	0…15	1
9.6	0.4...6.4	2.4	0.1…1.6	2	6	0…15	1
11.2	0.4...6.4	2.8	0.1…1.6	2	7	0…15	1
12.8	0.4...6.4	3.2	0.1…1.6	2	8	0…15	1
16.0	0.8...12.8	4.0	0.2...3.2	4	5	0…15	2
19.2	0.8...12.8	4.8	0.2...3.2	4	6	0…15	2
22.4	0.8...12.8	5.6	0.2...3.2	4	7	0…15	2
25.6	0.8...12.8	6.4	0.2...3.2	4	8	0…15	2
32.0	1.6...25.6	8.0	0.4...6.4	8	5	0…15	3
38.4	1.6...25.6	9.6	0.4...6.4	8	6	0…15	3
44.8	1.6...25.6	11.2	0.4...6.4	8	7	0…15	3
51.2	1.6...25.6	12.8	0.4...6.4	8	8	0…15	3
64.0	3.2...51.2	16.0	0.8...12.8	16	5	0…15	4
76.8	3.2...51.2	19.2	0.8...12.8	16	6	0…15	4
89.6	3.2...51.2	22.4	0.8...12.8	16	7	0…15	4
102.4	3.2...51.2	25.6	0.8...12.8	16	8	0…15	4

Table 2. Peaking time settings in the DP5 at 20 MHz and at 80 MHz.

5.2.3 Changes to PX4 Additional Parameters

This table shows the PX4 portion of the configuration packet, parameters not recognized by the DP4 but needed by the PX4 and the DP5. These values can be sent via RS232 or USB, as part of the configuration packet and when using the DP5 in "PX4 emulation mode". When using the DP4 in "DP4 emulation mode", these must be set over the USB interface, as boot options. This is discussed in section TBD.

14	D7-D6	0	Audible Count Rate Monitor (ACRM) off	Not recognized by DP5 – it has no audible count rate monitor.
		1	ACRM volume=medium	
		2	ACRM volume=low	
		3	ACRM volume=high	
	D5	0	Disable HV power supply	
		1	Enable HV power supply	
	D4	0	±8.5/5V supply=±5V	Not recognized by DP5 – these are set in hardware
		1	±8.5/5V supply=±8.5V	
	D3	0	Turn OFF PX4 power supplies	Turns OFF and ON the PC5 power supplies
		1	Turn ON PX4 power supplies	
	D2	0	Disable ±8.5/5V supply	
		1	Enable ±8.5/5V supply	
	D1	0	N/A	
	D0	0	Disable Thermoelectric Cooler (TEC) supply	
		1	Enable TEC supply	
15	D7	0	Front-end=non-inverting. Use if the input to the PX4 is a positive pulse	Note that the SDD produces positive pulses
		1	Front-end=Inverting. Use if the input to the PX4 is a negative pulse such as that produced by Amptek's XR100 detectors	Note that all Amptek detectors except the SDD produce negative pulses.
	D6-D4	0	N/A	
	D3-D0	0-15	Analog Gain Control B (See Table 1.)	New gain table. See Table 1.
16	D7-D0	0-15	HV MSB	
17	D7-D0	0-255	HV LSB HV (Volts) =(MSB*256 + LSB) * 0.732V]	
18	D7-D0	0-15	TEC Temperature setting MSB	
19	D7-D0	0-255	TEC Temperature setting LSB Temp (°C)=[(MSB*256 + LSB)*300/4096]-273	
20	D7-D0	0-15	Input Offset MSB	A setting of -2.048V instructs the DP5 to use the default; any other setting is interpreted the same as the PX4
21	D7-D0	0-255	Input Offset LSB Input Offset (mV)=(MSB*256 + LSB)-2048	
22	D7-D0	0	Disable Input Pole-Zero	See Note 1
		1-255	Enable input Pole-Zero	See Note 1
23	D7-D0	0-255	Fine Gain & Normalizer LSB[1]	
24	D7-D6	0	Digital scope trigger position=87% pretrigger	
		1	Digital scope trigger position=50% pretrigger	
		2	Digital scope trigger position=12% pretrigger	

[1] 'Fine Gain & Normalizer' setting controls both digital fine gain, and normalizes for different peaking times. The setting is 14 bits in size, and is:

Setting= INT[(FineGain x 8192)/PeakingTime], where

FineGain is in the range 0.75…1.25.

PeakingTime is an integer in the range 1…8 (see configuration byte 6 and)

Amptek recommends that fine gain be kept in the +/-25% range. The hardware will support a larger range but some loss of resolution will be observed.

		3	Digital scope trigger position=25% delayed trigger	
	D5-D0	0-63	Fine Gain & Normalizer MSB	
25	D7-D0	0-255	Preset Counts LSB[2]	
26	D7-D0	0-255	Preset Counts byte 2	
27	D7-D0	0-255	Preset Counts byte 3	
28	D7-D0	0-255	Preset Counts MSB [Preset counts=0x0000 if unused]	
29	D7	0	MCA mode (see byte 4 for # of channels)	
		1	MCS mode (see byte 4 for # of channels)[3]	
	D6	0	(Reserved)	
	D5	0	MCS disabled	
		1	MCS enabled	
	D4	0	(Reserved)	
	D3-D0	0	MCS Timebase=10mS/channel	
		1	MCS Timebase=20mS/channel	
		2	MCS Timebase=50mS/channel	
		3	MCS Timebase=100mS/channel	
		4	MCS Timebase=200mS/channel	
		5	MCS Timebase=500mS/channel	
		6	MCS Timebase=1 sec/channel	
		7	MCS Timebase=2 sec/channel	
		8	MCS Timebase=5 sec/channel	
		9	MCS Timebase=10 sec/channel	
		10	MCS Timebase=20 sec/channel	
		11	MCS Timebase=30 sec/channel	
		12	MCS Timebase=60 sec/channel	
		13	MCS Timebase=90 sec/channel	
		14	MCS Timebase=120 sec/channel	
		15	MCS Timebase=300 sec/channel	
30	D7	TBD		
	D6-D4	0	AUX_OUT2 = ICR	
		1	AUX_OUT2 = STATE(7)	
		2	AUX_OUT2 = TRIGGER	
		3	AUX_OUT2 = ONESHOT	
		4	AUX_OUT2 = RTD_ONESHOT	
		5	AUX_OUT2 = RTD_REJ	
		6	AUX_OUT2 = LIVE (Livetime counter gate)	
		7	AUX_OUT2 = VETO_	
	D3	0	General Purpose Counter: count falling edge	
		1	General Purpose Counter: count rising edge	
	D2-D0	0	General Purpose Counter = AUX_IN1	
		1	General Purpose Counter = AUX_IN2	
		2	General Purpose Counter = PILEUP	
		3	General Purpose Counter = RTD_BAD	
		4	General Purpose Counter = SCA8	

[2] The ‘Preset Count’ function counts events produced by SCA8. Therefore, the SCA8 thresholds must be set, and SCA8 must be enabled.

[3] The MCS mode also uses SCA8 as its source of counts. Therefore, the SCA8 thresholds must be set, and SCA8 must be enabled. .

		5	General Purpose Counter = TBD	
		6	General Purpose Counter = TBD	
		7	General Purpose Counter disabled	
↓				
63	D7	0	SCA8 disabled	
		1	SCA8 enabled	
	D6-D0	0-31	SCA8 Upper Threshold MSB	

5.2.4 Configuration Flow

Using ADMCA

ADMCA takes care of all the various configuration packets and options so the user need not worry about what is set and how.

For those who want to understand a little more what it does, ADMCA (which only supports the USB interface) will send all of the packets with their appropriate parameters. It will automatically put the DP5 into “PX4 emulation mode” to gain ready access to all hardware parameters. It will then send boot flags and configuration parameters as needed. When the DP5 is turned off, it will be in “PX4 emulation mode” (not DP4 emulation mode) and will have the high voltage and other settings last used with ADMCA.

DP4 Emulation Mode

To use the DP5 with legacy DP4 software, the user must first connect the DP5 to a computer using a USB connection. Then, use the Boot Manager to set the boot flags (including DP4 emulation mode) and the PX4 configuration parameters (section 5.2.3). These values will be stored in the DP5 and cannot be changed by legacy DP4 software (without modifying the software). The Standard Configuration parameters may be set using legacy DP4 software. **Note: some changes must be made to legacy DP4 software using USB for operation with the DP5. This is discussed in section 5.5 and summarized in section 5.7.**

PX4 Emulation Mode

To use the DP5 with legacy PX4 software, the user must first connect the DP5 to a computer using a USB connection. Then, use the Boot Manager to set the boot flags (including PX4 emulation mode). These values will be stored in the DP5 and cannot be changed by legacy PX4 software (without modifying the software). **Note: some changes must be made to legacy PX4 software using USB for operation with the DP5. This is discussed in section 5.5 and summarized in section 5.7.**

5.3 DATA PACKET CHANGES

5.3.1 Spectral Data Packets

No changes.

5.3.2 Status Packet Changes

Offset	Data bits	Allowed value	Description	DP5 Changes
0	D7-D0	0-255	Fast count LSB (Buffer A or B, depending on packet request)	Buffer A only
1	D7-D0	0-255	Fast count byte 2 (Buffer A or B, depending on packet request))	Buffer A only
2	D7-D0	0-255	Fast count byte 3 (Buffer A or B, depending on packet request))	Buffer A only
3	D7-D0	0-255	Fast count MSB (Buffer A or B, depending on packet request))	Buffer A only
4	D7-D0	0-255	Slow count LSB (Buffer A or B, depending on packet request))	Buffer A only
5	D7-D0	0-255	Slow count byte 2 (Buffer A or B, depending on packet request))	Buffer A only
6	D7-D0	0-255	Slow count byte 3 (Buffer A or B, depending on packet request))	Buffer A only

7	D7-D0	0-255	Slow count MSB (Buffer A or B, depending on packet request))	Buffer A only
8	D7-D4	3-15	FPGA version, major	
	D3-D0	0-15	FPGA version, minor	
9	D7-D0	0-99	Acc. Time (0-99, 1mS/count)	
10	D7-D0	0-255	Acc. Time LSB, 100mS/count	
11	D7-D0	0-255	Acc. Time byte 2	
12	D7-D0	0-255	Acc. Time MSB	
13	D7-D4	3-15	Firmware version, major	DP4/PX4 = 4; DP5 = 5
	D3-D0	0-15	Firmware version, minor	
14	D7-D0	0-255	Serial Number LSB	
15	D7-D0	0-255	Serial Number byte 2	
16	D7-D0	0-255	Serial Number byte 3	
17	D7-D0	0-255	Serial Number MSB	
18	D7-D4	0	(unused)	
	D3-D0	0-15	HV MSB	
19	D7-D0	0-255	HV LSB (0.5V/count)	
20	D7-D4	0	(unused)	
	D3-D0	0-15	Detector temperature MSB	
21	D7-D0	0-255	Detector temperature LSB (0.1 degree Kelvin/count)	
22	D7-D0	-128…+127	Board temp (1 °C/count, signed)	
23	D7	0	DP4 detected	“DP4 emulation mode” selected
		1	PX4 detected	“PX4 emulation mode” selected
	D6	0	Auto Fast Threshold not locked	
		1	Auto Fast Threshold locked	
	D5	0	MCA disabled	
		1	MCA enabled	
	D4	0	Preset count not reached	
		1	Preset Count reached, MCA disabled	
	D3	0	PX4 power supplies are OFF	
		1	PX4 power supplies are ON	
	D2	0	Oscilloscope data not ready	
		1	Oscilloscope data ready	
	D1	0	Unit is unconfigured	
		1	Unit is configured	
	D0	0	No power button configuration occurred	Not used by DP5
		1	Power button configuration occurred	
24	D7-D0	0-255	General Purpose Counter LSB	
25	D7-D0	0-255	G. P. Counter byte 2	
26	D7-D0	0-255	G. P. Counter byte 3	
27	D7-D0	0-255	G. P. Counter MSB	
28	D7	0	Auto Input Offset locked	
		1	Auto Input Offset searching	
	D6	0	MCS not finished	
		1	MCS finished	
	D5-D2	N/A		
	D1	0	No MCA RAM test was run	The RAM test is skipped on power-up in FW 5.03 to reduce the DP5 boot time.
		1	MCA RAM test was run	
	D0	0	No MCA RAM error detected or test wasn’t run Error! Bookmark not defined.	

		1	MCA RAM error was detectedError! Bookmark not defined.	
29	D7-D0	0-255	DCAL DAC setting, LSB	If a calibrated PC5 is detected, DCAL is read from the PC5's nonvolatile memory. Otherwise, DCAL is read from the DP5 nonvolatile memory.
30	D7-D4	0	N/A	
	D3-D0	0-15	DCAL DAC setting, MSB	
31	D7-D0	0-255	TECCL DAC setting, LSB	PZ Correction; set at the factory
32	D7-D4	0	N/A	uC temperature sensor offset calibration; signed byte
	D3-D0	0-15	TECCL DAC setting, MSB	
33	D7-D0			AN_IN MSB
34	D7-D0			AN_IN LSB (2.383mV/count)
35	D7-D0			VREF_IN MSB
36	D7-D0			VREF_IN LSB (2.383mV/count)
37				
38				
39	D7-D0	0-255		PC5 S/N LSB (if present)
40	D7-D0	0-255		PC5 S/N (if present)
41	D7-D0	0-255		PC5 S/N (if present)
42	D7-D0	0-255		PC5 S/N MSB (if present)
43	D7	0		PC5 not detected at power-up
		1		PC5 detected at power-up
	D6	0		PC5 HV polarity = negative
		1		PC5 HV polarity = positive
	D5	0		PC5 preamp supply = +/- 5V
		1		PC5 preamp supply = +/- 8.5V
	D4-D0			TBD
44	D7-D0	0-255		Livetime LSB (1mS/count)
45	D7-D0	0-255		Livetime
46	D7-D0	0-255		Livetime
47	D7-D0	0-255		Livetime MSB
48	D7-D6	TBD		DP5 Power supply status
	D5	0		-5.5V supply out of limit
		1		-5.5V supply in limit
	D4	0		+5.5V supply out of limit
		1		+5.5V supply in limit
	D3	0		1.2V supply out of limit
		1		1.2V supply in limit
	D2	0		2.5V supply out of limit
		1		2.5V supply in limit
	D1	0		3.3V supply out of limit
		1		3.3V supply in limit
	D0	0		Power input out of limit
		1		Power input in limit
48-51			N/A (Currently unused)	N/A (Currently unused)
52				Boot flags LSB
53				Boot flags MSB
54				HV LSB
55				HV MSB
56				TEC LSB
57				TEC MSB
58				Input Offset LSB
59				Input Offset MSB

60				Housekeeping ADC gain calibration (0-127 = 100-112.4%; 128-255 = 87.5-99.9%)
61				Housekeeping ADC offset calibration (8 bit signed, 1 LSB per count)
62				Spectrum offset LSB (1/4 channel per count @ 8K channels; 1/8 channel per count @ 4K channels… 1/128 channel per count @ 256 channels)
63	D7-D0	0		Spectrum offset MSB

5.3.3 Configuration ReadBack Packet Changes

No changes, but since Pipe 3 is shared between configuration readback packets and oscilloscope packets, care should be taken to be sure they aren't mixed up. If the oscilloscope data isn't used in the application, then nothing further needs be done. If oscilloscope data is used, then the application should check the amount of data returned from a Pipe 3 read to confirm that it's 64 bytes for a configuration packet readback, or 512 bytes for a read of the oscilloscope data. The mechanism for reading oscilloscope data is described below.

5.3.4 Oscilloscope Packet Changes

For RS232, the oscilloscope packets are unchanged. For USB, the oscilloscope packets share USB Pipe 3 with the configuration readback packet. The oscilloscope is configured and armed the same as in the DP4/PX4, but the data readback requires a few changes.

a. The application detects that oscilloscope data is ready (via the Status packet, offset 0x17, bit D2). When it's ready to read the oscilloscope trace data…

b. Send USB 'Vendor Request' 0x98 with a value field of 0x0000. This switches Pipe 3 from Configuration readback Packets to Oscilloscope packets. If oscilloscope data is not ready when this Vendor Request is received, it will be ignored and Pipe 3 will continue to produce Configuration readback Packets. If oscilloscope data is ready when the Vendor Request is received, then…

c. A read of Pipe 3 will first produce a zero-length packet. A retry (2nd read) of the pipe will produce the 512-byte oscilloscope trace data. Pipe 3 will automatically switch back to producing Configuration readback Packets after the oscilloscope data has been read.

5.3.5 DP5 Text Block (similar to 'PX4 Name')

The DP5 has the ability to store up to 512 bytes of text or binary data in its nonvolatile memory. (The PX4 could store 64 bytes of text.) The method for writing and reading this block is significantly different (and simpler) than the PX4. As of FW 5.03, the DP5 Text Block can only be written via USB. RS232 support will be added.

The text or binary data is written using a series of 16 USB 'Vendor Requests', each of which contains 32 bytes of data. The Vendor Request 'index' field indicates which 32-byte block is to be sent; index ranges from 0..15. The 'Length' field must be 32, and the data field contains 32 bytes of text or binary data. An Index value of 0 sends bytes 0..31; an Index value of 1 sends bytes 32..63. An Index value of 15 sends bytes 480-511 **and also triggers the copying of the entire 512-byte block into EEPROM**. It isn't necessary to write all sixteen 32-byte blocks if they aren't all needed, but the final block (Index=15) must be written in order for the data to be saved.

Vendor Request	Value field	Index field	Length field	Bytes written
0x7F	0	0	32	0-31
0x7F	0	1	32	32-63
0x7F	0	2	32	64-95
0x7F	0	3	32	96-127
0x7F	0	4	32	128-159
0x7F	0	5	32	160-191

0x7F	0	6	32	192-223
0x7F	0	7	32	224-255
0x7F	0	8	32	256-287
0x7F	0	9	32	288-319
0x7F	0	10	32	320-351
0x7F	0	11	32	352-383
0x7F	0	12	32	384-415
0x7F	0	13	32	416-447
0x7F	0	14	32	448-479
0x7F	0	15	32	480-511 and triggers copy to EEPROM

The readback of the DP5 Text Block, like the oscilloscope readback, shares USB Pipe 3 with the configuration readback packet. To read back the 512-byte Text Block:

- Send USB 'Vendor Request' 0x98 with a value field of 0x0001. This switches Pipe 3 from Configuration readback Packets to the Text Block.
- A read of Pipe 3 will first produce a zero-length packet. A retry (2nd read) of the pipe will produce the 512-byte Text Block. Pipe 3 will automatically switch back to producing Configuration readback Packets after the oscilloscope data has been read.

5.4 Changes to RS232 Interface

The RS232 interface now supports a baud rate of 115,200 baud, in addition to the DP4/PX4 baud rate of 57,600 baud. (This is selected by the boot flags – see section 5.2.1.) Also, several additional packet requests are supported which feature improved deadtime and ease of use. (See section 5.1.3.).

5.5 Changes to USB Interface

Due to the fact the DP5 microcontroller supports fewer USB endpoints than the DP4/PX4 microcontroller, some of the USB pipes have changed in the DP5. This is made easier by the fact that 'Buffer B' functionality has been removed in the DP5. [Buffers A & B didn't support double buffering in the traditional sense, and were not useful for most customers.]

Pipe 0 (configuration), pipe 1 (status packet) and pipe 2 (spectral packets) are unchanged. Pipe 3 is now used by the configuration readback function (which used to be on Pipe 6). Oscilloscope data also uses Pipe 3, as described above.

USB Pipe	DP4/PX4 Pipe Function	DP5 Pipe Function
Pipe 0	Configuration data OUT to PX4 (USB Device Endpoint OUT1)	Same
Pipe 1	Buffer A Status IN from PX4 (USB Device Endpoint IN1)	Same – retries required
Pipe 2	Buffer A Spectrum IN from PX4 (USB Device Endpoint IN2)	Same – retries required
Pipe 3	Buffer B Status IN from PX4 (USB Device Endpoint IN3)	Configuration readback, oscilloscope packets, and misc data – retries required
Pipe 4	Buffer B Spectrum IN from PX4 (USB Device Endpoint IN4)	N/A
Pipe 5	Configuration readback IN from PX4 (USB Device Endpoint IN5)	N/A
Pipe 6	Oscilloscope trace data IN from PX4 (USB Device Endpoint IN6)	N/A

Application software must be modified to perform retries on failed reads of IN1 (status), IN2 (spectrum) or IN3 (configuration readback/oscilloscope data.) See sections 5.1.3 and 5.7.

5.6 USB Vendor Requests

The DP5 continues to use USB 'Vendor Requests' for a number of discrete commands (enable MCA, clear spectrum, etc.) These operate through the default control pipe (endpoint 0) as per the USB specification. Table 3 summarizes which Vendor Requests are supported by the PX4, DP4 and DP5.

USB 'Vendor Request'	**16-bit 'Value' field**	**PX4 Function**	**DP4 function**	**DP5 function**
0x7F		Write 'PX4 Name'	N/A	Write Misc Data
0x80	Ignored	Clear spectrum buffer A	Same as PX4	
0x81	Ignored	Clear spectrum buffer B	Same as PX4	N/A
0x82	Ignored	Enable MCA/MCS	Same as PX4	
0x83	Ignored	Disable MCA/MCS	Same as PX4	
0x84	Ignored	Arm digital oscilloscope	Same as PX4	
0x85	Ignored	Autoset Input Offset (fine steps)	N/A	Same as PX4
0x86	Ignored	Autoset Input Offset (coarse steps followed by fine steps)	N/A	Same as PX4
0x87	Ignored	Autoset fast threshold[4]	Same as PX4	
0x88	Ignored	N/A	Clear PA0 (JP10.1)	Clear IO0 (J5.17)
0x89	Ignored	N/A	Set PA0 (JP10.1)	Set IO0 (J5.17)
0x8A	Ignored	N/A	Clear PA1 (JP10.2)	Clear IO1 (J5.18)
0x8B	Ignored	N/A	Set PA1 (JP10.2)	Set IO1 (J5.18)
0x8C	Ignored	N/A	Clear PA2 (JP10.3)	Clear IO2 (J6.13)
0x8D	Ignored	N/A	Set PA2 (JP10.3)	Set IO2 (J6.13)
0x8E	Ignored	N/A	Clear PA3 (JP10.4)	Clear IO3 (J6.14)
0x8F	Ignored	N/A	Set PA3 (JP10.4)	Set IO3 (J6.14)
The following shaded settings are saved in nonvolatile memory				
0x90	DAC setting	Set Diode Calibration DAC (DCAL)	N/A	Same as PX4; also written to PC5 EEPROM if present
0x91	Calibration settings	Set TEC Current Limit DAC (TECCL)	N/A	PZ Correction (LSB) uC Temp Cal (MSB)
0x92	Boot flags	N/A	N/A	Set boot flags
0x93	DAC setting	N/A	N/A	Set HV (for DP4 emulation mode; not used in PX4 emulation mode)
0x94	DAC setting	N/A	N/A	Set detector temp (for DP4 emulation mode; not used in PX4 emulation

[4] 'Autoset fast threshold' only works with no source present and low levels of background counts (less than approx. 20 cps.)

				mode)
0x95	DAC setting	N/A	N/A	Set input offset (for DP4 emulation mode; not used in PX4 emulation mode)
0x96	Calibration setting	N/A	N/A	HK ADC Gain Cal (LSB); HK ADC Offset Cal (MSB)
0x97	User setting	N/A	N/A	Set spectrum offset
0x98	0	N/A	N/A	Request digital scope data (Pipe 3)
0x98	1	N/A	N/A	Request Text Block/misc data (Pipe 3)
0x99	TBD	N/A	N/A	TBD
0x9A	TBD	N/A	N/A	TBD
0x9B	TBD	N/A	N/A	TBD
0x9C	TBD	N/A	N/A	TBD
0x9D	Reserved	N/A	N/A	Reserved
0x9E	TBD	N/A	N/A	TBD
0x9F	TBD	N/A	N/A	TBD

Table 3. USB Vendor Requests in the DP4, PX4, and DP5 (FW 5.03).

5.7 HINTS FOR SOFTWARE MIGRATION

This section discusses a number of issues that need to be considered when migrating an existing DP4/PX4 application for use with the DP5. Applications that use the DPP API library to handle USB communications should only need to use an undated version of the DPP API.

[USB] Do not attempt to open USB pipes 4-6, which no longer exist in the DP5.

[USB] Configuration readback has moved from Pipe 5 (DP4/PX4) to Pipe 3 (DP5.)

[USB] Oscilloscope trace data now shares Pipe 3 with the configuration readback function..

[USB] The 64-byte ‘PX4 Name’ function has been replaced by the 512-byte DP5 ‘Text Block’ (which can be 8-bit binary or text data). Like the oscilloscope data, it shares Pipe 3 with the configuration readback function. The mechanism for writing the Text Block is quite a bit different than the PX4 method although they both utilize USB Vendor Request 0x7F.

For application software that needs to work with the DP4/PX4/DP5:

a. [USB] First, open the device, then open only Pipe 1 (status). Read a status packet, and following the logic details below, open the DP4/PX4 pipes, or the DP5 pipes.

b. [USB/RS232] In the status packet, a FW version of 5.00 (0x50) or higher indicates a DP5, less than 5.00 indicates a DP4 or PX4.

b1. For DP5: Status offset 0x17, bit D7: 1=PX4 emulation mode, 0=DP4 emulation mode

b2. For DP4/PX4: Status offset 0x17, bit D7: 1=PX4, 0=DP4

[RS232] The DP5 supports 115.2kbaud, in addition to 57.6kbaud used in the DP4/PX4. A boot flag determines the DP5 baud rate – make sure it matches the baud rate of the application software.

[RS232] Several new packet requests are available to make the data exchange much easier – send one request, receive the entire data set. Plus, deadtime due to communications is much less using these new requests. See section 5.1.3.

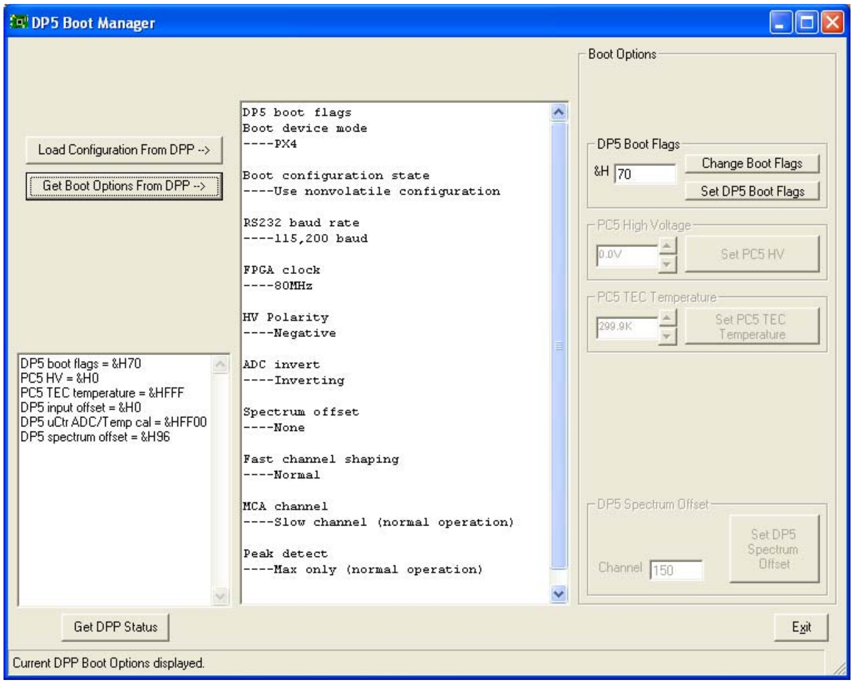
[USB] A mechanism for retrying USB IN transfers (DP5-to-host) must be added. The application software should detect when a read attempt on USB pipes 1, 2 or 3 returns zero bytes, or fails because of a timeout. At least one additional attempt to read the pipe must be tried. [Calls to the USBDRVD_PipeReadTimeout() function of the USB driver should check to see if either zero bytes were read, or if the return value=0, indicating a timeout or other failure. In either case, the call should be attempted again.]

[USB] On program exit, be sure to close only those USB pipes that were opened. [i.e. if the DP5 pipes were opened, be sure that only those pipes are closed, and not the additional DP4/PX4 pipes. Trying to close unopened pipes will cause problems.]

5.8 Using the Boot Manager

The DP5 Boot Manger application is used for changing the various non-volatile settings in the DP5. In addition, it can read the current configuration of the DP5 and display the various settings. To make use of the DP5 Boot Manager, it must first be installed from the Amptek CDROM. Run 'setup.exe' from \Additional DPP Software\DP5\DP5BootManager\SWSETUP on the CDROM to install the application. The DP5 Boot Manager only operates via the USB interface.

After installing the application, connect the USB cable and apply power to the DP5. Clicking the 'Get Boot Options From DPP' button will display the current Boot Options from the DP5. Below is an example of a DP5 with an Amptek SDD detector.



```
DP5 boot flags
Boot device mode
----PX4

Boot configuration state
----Use nonvolatile configuration

RS232 baud rate
----115,200 baud

FPGA clock
----80MHz

HV Polarity
----Negative

ADC invert
----Inverting

Spectrum offset
----None

Fast channel shaping
----Normal

MCA channel
----Slow channel (normal operation)

Peak detect
----Max only (normal operation)
```

```
DP5 boot flags = &H70
PC5 HV = &H0
PC5 TEC temperature = &HFFF
DP5 input offset = &H0
DP5 uCtr ADC/Temp cal = &HFF00
DP5 spectrum offset = &H96
```

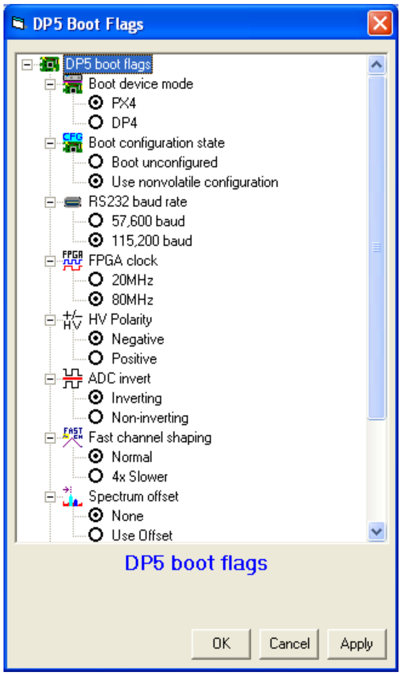
&H 70

0.0V

299.9K

Channel 150

Current DPP Boot Options displayed.



DP5 boot flags
- Boot device mode
  - ● PX4
  - ○ DP4
- Boot configuration state
  - ○ Boot unconfigured
  - ● Use nonvolatile configuration
- RS232 baud rate
  - ○ 57,600 baud
  - ● 115,200 baud
- FPGA clock
  - ○ 20MHz
  - ● 80MHz
- HV Polarity
  - ● Negative
  - ○ Positive
- ADC invert
  - ● Inverting
  - ○ Non-inverting
- Fast channel shaping
  - ● Normal
  - ○ 4x Slower
- Spectrum offset
  - ● None
  - ○ Use Offset

DP5 boot flags

5.8.1 Boot Flags

To change the individual Boot Flags, click the 'Change Boot Flags' button, which displays the window shown below. After changing the Boot Flags, click the 'OK' button. Then, on the main Boot Manager screen, **click 'Set DP5 Boot Flags' to send the new Boot Flags to the DP5**. The meaning of the individual Boot Flags is discussed below.

Boot Device Mode

- PX4: In the PX4 emulation mode, the DP5 uses the HV, TEC, Input Offset and ADC Invert settings from the configuration packet. ADMCA requires the PX4 Emulation Mode, and will switch to this mode if it detects that the DP5 is configured in DP4 Emulation Mode. [In the PX4 Emulation Mode, the HV, TEC and Input Offset Boot Options are ignored, since these are taken from the configuration packet.]
- DP4: Since the HV, TEC, Input Offset and ADC Invert settings don't exist in the DP4 configuration packet, these settings come from the HV, TEC and Input Offset Boot Options.

Note: This setting requires the DP5 to be power-cycled to take effect.

Boot Configuration State

- Boot unconfigured: In this mode, upon power-up, the DP5 will be in an unconfigured state. If present, the PC5 and detector will not be powered on until the first configuration packet is received. Also, the shaper, MCA, etc. are not configured. [After power-up, the status packet byte 23, bit D1 indicates whether the device is configured or unconfigured.]
- Use Nonvolatile Configuration: Upon power-up, the DP5 will use the last configuration it received before it was powered off. It will turn on all supplies, including the PC5 HV & TEC supplies (if present, and if these were previously configured to be powered on). [The HV & TEC supplies will be turned on approximately 2 seconds after power is applied.] The DP5 will be ready to take data, once the detector has cooled.

Note: This setting requires the DP5 to be power-cycled to take effect.

RS232 Baud Rate

- 57,600: This primarily exists for backward compatibility with DP4/PX4 RS232 applications.
- 115,200: Recommended, for higher-speed transfers.

Note: This setting requires the DP5 to be power-cycled to take effect.

FPGA Clock

- 20MHz: Operating the DP5 at 20MHz consumes less power than 80MHz and allows longer peaking times (up to 102us), but limits throughput, minimum peaking time (0.8us), etc.
- 80MHz: This is the recommended clock rate for SDD detectors; it allows faster peaking times (min 0.2us) and higher throughput than 20MHz, but requires a bit more power.

Note: This setting is ‘live’ and will take effect immediately.

HV Polarity

The polarity of the HV supply on the PC5 is not under software control; it is configured in hardware, via jumpers set at the factory. This software configuration setting tells the DP5 what polarity is required, and the DP5 will not allow the PC5 to turn on if its polarity doesn’t match the Boot Flag.

- Negative: Amptek’s SDD detectors require negative HV.
- Postive: All other Amptek detectors require positive HV.

Note: This setting requires the DP5 to be power-cycled to take effect.

ADC Invert

This setting is only used in DP4 Emulation Mode. In PX4 Emulation Mode, the polarity is commanded via the configuration packet.

- Inverting: Used for SDDs and other positive-pulse preamps. (ADC Inversion, combined with the DP5’s inverting front-end results in a non-inverting input.)
- Non-inverting: Used with negative-pulse preamps (all Amptek non-SDD preamps.)

Note: This setting requires the DP5 to be power-cycled to take effect.

Fast Channel Shaping

- Normal: The fast channel peaking time is 400nS (for a 20MHz clock) or 100nS (at 80MHz).
- 4x Slower: The fast channel peaking time is 1.6uS (for a 20MHz clock) or 400nS (at 80MHz).

Note: This setting requires the DP5 to be power-cycled to take effect.

Spectrum Offset

- None: No spectrum offset is applied – the spectrum zero is derived soley from BLR.
- Use Offset: The Boot Option ‘Spectrum Offset’ is added to BLR. This shifts the spectrum left or right along the horizontal axis. This permits channel “zero” to have “zero” energy.

Note: This setting is 'live' and will take effect immediately. To change the spectrum offset, set this option to 'Use Offset", click 'OK', then 'Set DP5 Boot Flags.' Next, read the boot flags again by selecting the 'Get Boot Options from DPP' button. This will enable the 'DP5 Spectrum Offset' box, where the offset channel can be entered and set.

Peak Detect

- Max only (normal operation): This is the correct setting for normal operation.
- Max and min: This allows the peak detect (and MCA) to capture local minima of the shaped pulse stream, in addition to local maxima. This is useful for capturing the 'noise Gaussian' for determining the true spectrum zero. (The slow threshold must be set very low to accomplish this.)

Note: This setting is 'live' and will take effect immediately. It is also volatile – it will reset to 'Max only' the next time the DP5 is powered on.

MCA Channel

- Slow channel (normal operation): This is the correct setting for normal operation.
- Fast channel: This sets the source for the MCA to the fast channel, rather than the slow channel. The fast channel does not have BLR or a pole-zero, but this setting does allow some insight into the operation of the fast channel.

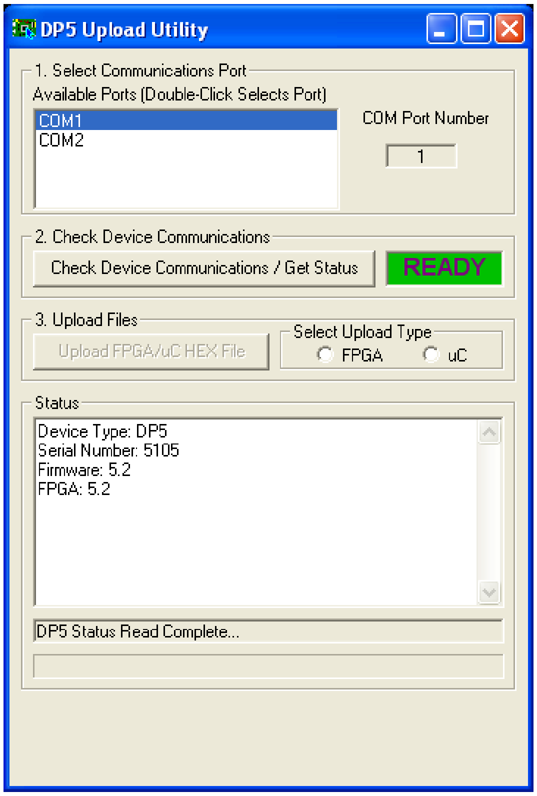
Note: This setting is 'live' and will take effect immediately. It is also volatile – it will reset to 'Slow channel' the next time the DP5 is powered on.

5.9 UPLOADING FIRMWARE AND SOFTWARE

One of the major upgrades of the DP5 (compared to the PX4 and DP4) is the ability to upload new FPGA and microcontroller firmware through the RS232 interface. An FPGA upload takes approximately 40 seconds and a microcontroller code upload takes around 3 seconds.

To upload updated firmware:

1. New firmware files will need to be acquired, either via email from Amptek, or via the Amptek.com website.
2. Install the DP5Loader application from the Amptek CD. The installer is located at \Additional DPP Software\DP5\DP5 Upload Utility\SWSETUP\SETUP.EXE on the CDROM.
3. Connect the 9-pin-to-stereo-plug RS232 cable that came with the DP5 between a PC serial port and the DP5, and apply power to the DP5.
4. Run DP5Loader.EXE (which you installed in Step #2).
5. Double-click the COM port that the DP5 is connected to.
6. Click 'Check Device Communications / Get Status'. This will show 'READY' if it successfully finds the DP5, or ERROR if not. For example, these screen shots show updating a DP5 from FW 5.02 & FP 5.02 to FW 5.03 & FP 5.03:

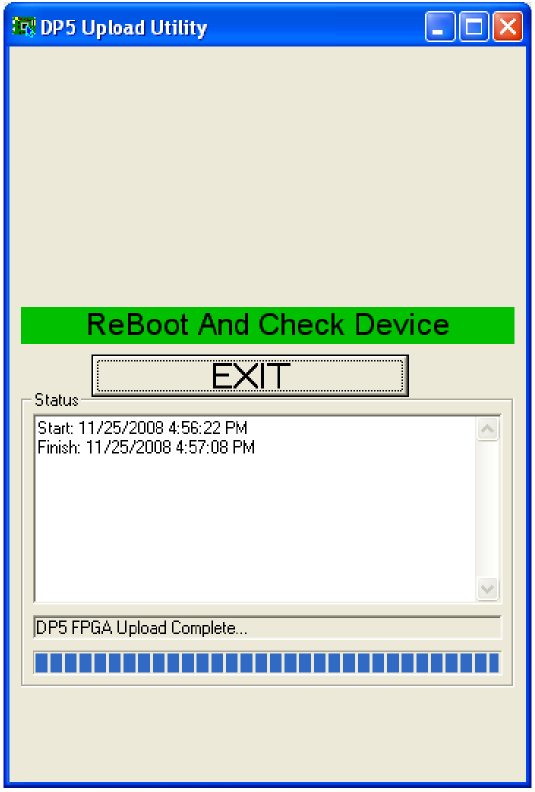

DP5 Upload Utility
1. Select Communications Port
Available Ports (Double-Click Selects Port)
COM Port Number
2. Check Device Communications
Check Device Communications / Get Status
READY
3. Upload Files
Upload FPGA/uC HEX File
Select Upload Type
○ FPGA
○ uC
Status

COM1
COM2

COM Port Number: 1

Device Type: DP5
Serial Number: 5105
Firmware: 5.2
FPGA: 5.2

DP5 Status Read Complete...



Start: 11/25/2008 4:56:22 PM
Finish: 11/25/2008 4:57:08 PM

DP5 FPGA Upload Complete...

7. Select upload type "**FPGA**".

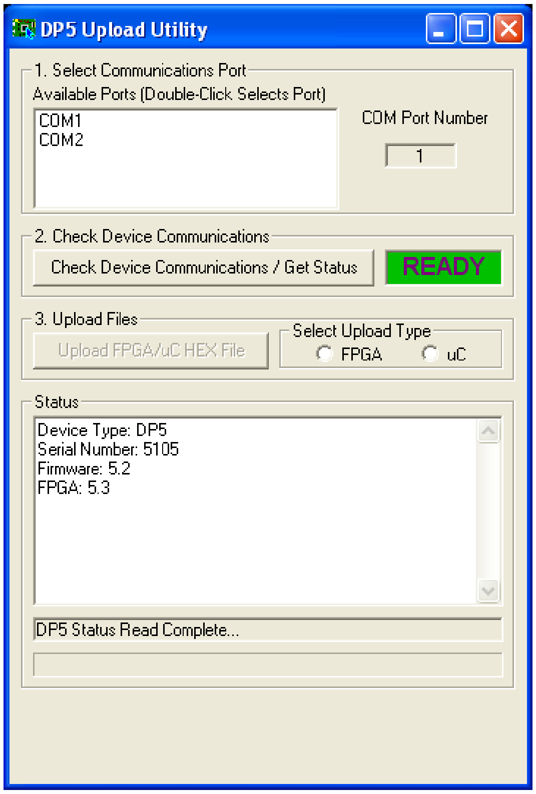
8. Click the 'Upload FPGA/uC HEX File' and select the FPGA programming file ('**_dp5_fpga_v502.mcs**' for example. All FPGA programming files have an .mcs extension.) It will take around 40 seconds to program at 115kbaud.

9. The program says 'Reboot and check device'. **Power cycle the DP5**.

10. Click 'Exit.'

11. Run DP5Loader.EXE again.

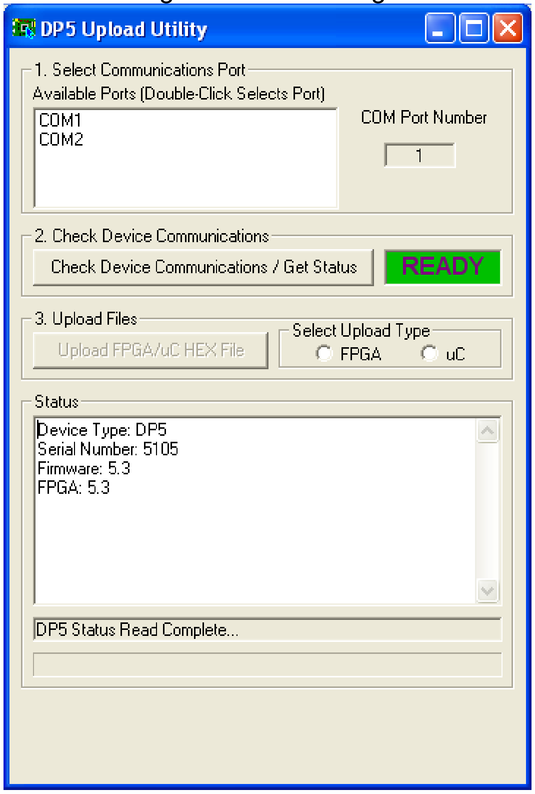
12. Click 'Check Device Communications / Get Status'. It should now show the version number of the FPGA you uploaded (5.3, for the example below), and the old Firmware version.


DP5 Upload Utility
1. Select Communications Port
Available Ports (Double-Click Selects Port)
COM1
COM2
COM Port Number
2. Check Device Communications
Check Device Communications / Get Status
READY
3. Upload Files
Upload FPGA/uC HEX File
Select Upload Type
○ FPGA
○ uC
Status

COM Port Number: 1

Device Type: DP5
Serial Number: 5105
Firmware: 5.2
FPGA: 5.3

DP5 Status Read Complete...


DP5 Upload Utility
1. Select Communications Port
Available Ports (Double-Click Selects Port)
COM1
COM2
COM Port Number
2. Check Device Communications
Check Device Communications / Get Status
READY
3. Upload Files
Upload FPGA/uC HEX File
Select Upload Type
○ FPGA
○ uC
Status

COM Port Number: 1

Device Type: DP5
Serial Number: 5105
Firmware: 5.3
FPGA: 5.3

DP5 Status Read Complete...

13. Select upload type "**uC**".

14. Click the 'Upload FPGA/uC HEX File' and select the microcontroller programming file ('**dp5_uC_v503.hex**' for example – uC programming files have the .hex extension.) It will take around 4 seconds to program at 115kbaud.

15. The program says 'Reboot and check device'. **Power cycle the DP5 again** and exit the program.

16. Run DP5Loader.EXE again and click 'Check Device Communications / Get Status' to confirm the Firmware & FPGA versions – in this example, Firmware 5.3 & FPGA 5.3. All done!

5.10 VERSIONS

DP5 FW 5.03 and FP 5.03 are the current versions as of the release of this document.

ADMCA 1.0.2.18 or higher should be used with the DP5.