Hardware Design Description

Plantation Productions Open Source/Open Hardware Data Acquisition System

**Table of Contents**

[1 Introduction 4](#_Toc6091279)

[1.1 Purpose 4](#_Toc6091280)

[1.2 Scope 4](#_Toc6091281)

[1.3 Contents of the Document 4](#_Toc6091282)

[1.4 Document Conventions 4](#_Toc6091283)

[2 Hardware Architecture 6](#_Toc6091284)

[2.1 Design Considerations 7](#_Toc6091285)

[2.1.1 Assumptions and Dependencies 7](#_Toc6091286)

[2.1.2 Related Software or Hardware 7](#_Toc6091287)

[2.1.3 End-User Characteristics 7](#_Toc6091288)

[2.1.4 System Interfaces 7](#_Toc6091289)

[2.1.5 Fail-Safe 8](#_Toc6091290)

[2.1.6 Hardware Basis 8](#_Toc6091291)

[2.1.7 User Interfaces 8](#_Toc6091292)

[2.1.8 Hardware Interfaces 8](#_Toc6091293)

[2.1.9 Software Interfaces 9](#_Toc6091294)

[2.1.10 General Constraints 9](#_Toc6091295)

[2.1.11 Goals and Guidelines 10](#_Toc6091296)

[2.1.12 Developmental Methods 11](#_Toc6091297)

[3 Hardware Design Description 12](#_Toc6091298)

[3.1 Design Stakeholders 12](#_Toc6091299)

[3.2 Design Concerns 12](#_Toc6091300)

[3.2.1 Dow Site Management 12](#_Toc6091301)

[3.2.2 DTRR Operators 12](#_Toc6091302)

[3.2.3 NRC 12](#_Toc6091303)

[3.2.4 DAQ Project Management 12](#_Toc6091304)

[3.2.5 DAQ Hardware Engineering/Development 12](#_Toc6091305)

[3.2.6 DAQ Hardware Engineering/Testing 13](#_Toc6091306)

[4 DAQ Detailed Design 14](#_Toc6091307)

[4.1 PPAIO-16/4 16-Channel Analog Input/4-Channel Analog Output Board 14](#_Toc6091308)

[4.1.1 Capacity, Interface, and Accuracy 14](#_Toc6091309)

[4.1.2 PPAIO-16/4 I2C Interface, Port Assignment, and Daisy-Chaining 15](#_Toc6091310)

[4.1.3 PPAIO-16/4 Power Supply 17](#_Toc6091311)

[4.1.4 PPAIO-16/4 Analog Output Amplifiers 18](#_Toc6091312)

[4.2 PPAC4 4-Channel Analog Conditioning Board 20](#_Toc6091313)

[4.2.1 Capacity and Interface 20](#_Toc6091314)

[4.2.2 PPAC4 Power Supply 21](#_Toc6091315)

[4.2.3 PPAC4 Isolation Amplifier Section 21](#_Toc6091316)

[4.2.4 PPAC4 Amplifier Section 22](#_Toc6091317)

[4.3 PPAC420 8-Channel 4-20 mA Analog Input Conditioning Board 24](#_Toc6091318)

[4.3.1 Capacity and Interface 24](#_Toc6091319)

[4.3.2 PPAC420 Power Supply 24](#_Toc6091320)

[4.4 PPRLYIO-12 12-Channel I/O Relay Board 25](#_Toc6091321)

[4.4.1 Capacity and Interface 25](#_Toc6091322)

[4.4.2 PPRLYIO-12 Power Supply 27](#_Toc6091323)

[4.4.3 PPRLYIO-12 Relay and Data LED Activation 28](#_Toc6091324)

Plantation Productions Open Source/Open Hardware Data Acquisition System

**HARDWARE DESIGN DESCRIPTION**

# Introduction

This document is the Hardware Design Description (HDD) for the hardware contained in the Digital Data Acquisition System (DAQ) created by Plantation Productions, Inc.

The DAQ system provides digital and analog I/O capabilities for embedded systems; specifically, it was designed to provide data acquisition for TRIGA™ Research Reactors (TRIGA is a registered trademark of General Atomics).

## Purpose

The purpose of this document is to describe the hardware design for the DAQ system. The intended audiences for this document are the engineering, product assurance and management personnel involved in the hardware development.

## Scope

The scope of the software, which is based on the DAQ Hardware Requirements Specification (DAQ HRS) per the requirements that have been allocated to the DAQ system hardware. From this document the hardware will be developed.

## Contents of the Document

The general description including product perspective, product functions, user characteristics and general constraints is included in Section 2.

The design descriptions are included in Section 3.

The HDD to HRS traceability (reverse traceability) is documented in a separate document (Excel spreadsheet).

## Document Conventions

All tags shall take the form:

<whitespace> [DAQ\_HDD\_xxx]

where "xxx" is a three-digit number reserved for HDD usage.

For HDD tags, should the need arise to insert a new HDD tag between two other values (e.g., add a requirement between 030 and 031) then a decimal fractional number shall be appended to the HDD tag number (e.g., 031.5). Any number of decimal point suffixes can be added, if needed (e.g., 030.5.2).

Examples:

From the HDD:

<whitespace> [DAQ\_HDD\_053]

Note: because an external script may be used to extract requirements from this document, it is very important that actual requirements in this document begin on a new line (with nothing but white space preceding the requirement) and that the requirement take exactly the form shown above. The regular expression used by the requirement extraction script is the following:

<whitespace>\* '[DAQ\_HDD\_' {0-9}{0-9}{0-9}('.' {0-9}+)\* ']' <whitespace>\* ':' <whitespace>\* .\* '\n'

Where ".\*" represents an arbitrary sequence of characters not including a new line and "<whitespace>\*" represents zero or more tab or space characters. The actual string "<whitespace>" appears in front of the examples in this section so that they will *not* be captured by this script; the string "<whitespace>" should not appear in front of actual requirements (though actual whitespace is certainly permissible).

# Hardware Architecture

The DAQ digital data acquisition system (Hereafter, "DAQ") consists of several main components:

1. DAQ\_IF (DAQ interface board). This circuit board interfaces to a single-board computer such as a Netburner MOD54415 Evaluation Board, a Raspberry Pi 3 Model B, or a Teensy 3.2.
2. PPDIO96 (96-input digital I/O board). This board connects to the PPDIO96 bus connector on the DAQ\_IF board and provides 96 digital I/O pins. Up to six PPDIO96 boards can be connected together in a daisy chain off one DAQ\_IF board.
3. PPOPTO-12 (12-channel digital input opto-isolation). This board provides 12 channels of digital opto-isolation. This board connects to one of the 12-input bank connectors on the PPDIO96. Up to eight PPOPTO-12 boards can be connected to a single PPDIO96 board providing 96 opto-isolated inputs.
4. PPBreakout (12-channel breakout board for PPDIO96). This board connects to one of the 12-input bank connectors on the PPDIO96. It provides 12 sets of two-terminal screw terminals for the 12 I/O pins on that PPDIO96 bank. The PPBreakout board is useful when feeding inputs to the PPDIO96 that do not require isolation or for connecting digital outputs from the PPDIO96 to the rest of the system.
5. PPRELAY-12 (12-channel mechanical relay). this board connects to the PPDO bus connector on the DAQ\_IF board and provides 12 mechanical relay digital outputs (with NC/NO terminals). In addition, for digital I/O pins capable of sinking up to 150 mA are also available. In theory, an unlimited number of these boards (and PPSSR-16 boards) can be daisy-chained off the PPDO connector. In practice, fan-out limits the system to 10 or fewer boards.
6. PPSSR-16 (16-channel Solid-State Relay). This board connects to the PPDO bus connector on the DAQ\_IF board (or daisy-chains with other PPSSR-16 and PPRELAY-12 boards). It provides 16 solid-state relay controlled outputs. In theory, an unlimited number of these boards (and PPRELAY-12 boards) can be daisy-chained off the PPDO connector. In practice, fan-out limits the system to 10 or fewer boards.
7. PPAIO-16/4 (16 analog inputs, 4 analog outputs). This board connects to one of the four 6-pin I2C connectors on the DAQ\_IF board and provides 16 single-ended 16-bit analog inputs (or 8 double-ended inputs) and four 12-bit analog outputs. The analog inputs support up to a 0-5V range (with programmable gain amplifiers allowing the full-range input of smaller signals). The analog outputs include amplifier circuitry to support outputs in the range -10V to +10V (easily programmable as -5V to +5V, if desired).
8. PPAC4 (4-channel analog conditioning). This board provides analog isolation and conditioning. The four inputs to this board are single-ended analog signals in the range -10 to +10V. The four outputs from this board are double-ended analog signals (0-4.096V) that provide a full 16-bit range to the PPAIO-16/4 analog inputs. This board also contains isolation amplifiers that isolate the double-ended output from the single-ended input signal.
9. PPAC420 (8-channel 4-20mA analog conditioning). This board accepts 4-20 mA current loop inputs and produces a single-ended -1.25 to +5V output. Typically, this outputs would be fed as inputs to the PPAC4 (to provide isolation and full-range ADC conversion), though if isolation is not required the inputs could be fed directly into the single-ended inputs on the PPAIO-16/4 board.

## Design Considerations

### Assumptions and Dependencies

In the most common use case, the DAQ system will be connected to some host computer via Ethernet, USB, or RS-232 Serial. While it is possible for the DAQ system to operate at a stand-alone data acquisition system, such usage will be rare.

### Related Software or Hardware

When running with a Netburner MOD54415 module, the system will be running the µC/OS real-time operating system. When running on a Raspberry Pi, the system will likely be running a variant of the Linux operating system. When running with a Teensy 3.2 module, the software will (likely) be using the Arduino-style library modules provided for the Teensy 3.2.

Because the DAQ system operates using I2C and SPI interfaces, it is quite possible to hook up other I2C and SPI devices. This document will not consider such related hardware.

### End-User Characteristics

There are three types of "end-users" associated with the DAQ system: system end users, technicians, and system designers.

System end users (those individual using the final system) may not even be aware of the DAQ system – they are only interested in using the system as a whole and the DAQ system might be a small or hidden part of that whole system.

Technicians are those individuals responsible for maintaining and calibrating the system. Their responsibilities will likely include maintaining and calibrating the components of the DAQ system. As such, they are likely to be concerned with the electronic design and operation of the DAQ system boards.

System designers are those indivduals who design a system around the DAQ system. Clearly, they will require the most knowledge and experience with the DAQ system hardware.

### System Interfaces

The DAQ system provides for up to three system interfaces (depending on the computer module controlling the DAQ\_IF board): Ethernet (Netburner and Raspberry Pi), USB (Teensy 3.2), and RS-232 (Netburner, Raspberry Pi, and Teensy 3.2).

### Fail-Safe

The DAQ system operates in a fail-safe fashion. After a power on restart the PPDIO96 comes up with all the pins programmed as inputs; the PPRELAY-12 and PPSSR-16 digital output boards come up with all the relays turned off (NC in the "C" state, NO in the "O" state, and solid-state relays in the high-impedance state).

The DAQ-IF includes a watch-dog timer that trips a relay if it is not refreshed within 5-10 seconds. If the firmware on the CPU module hangs up or otherwise fails to refresh the watchdog timer within the specified amount of time, a relay will be actuated. The NC/NO outputs from this relay are available for use in the rest of the system (application-defined) to handle the software anamoly. In addition, this signal also appears on the PPDIO96 and PPDO busses. Activation of this signal will place the PPDIO96, PPRELAY-12, and PPSSR-16 boards in the fail-safe state.

Finally, a software programmable line (reset) appearing on the PPDIO96 and PPDO busses will also put the boards into their fail-safe state. Note that the reset line will also reset the watchdog timer if it has expired.

The DAQ\_IF board uses a separate digital output line from the CPU module to refresh the watchdog timer. The CPU module must pulse this line (low-to-high edge) at least once every timeout period to keep the watchdog timer from timing out. Note that the watchdog timer is a hardware (RC-based) circuit, it does not depend on software to produce a watchdog timeout.

When designing an end-system around the DAQ system, you should carefully consider the fail-safe design of the DAQ system and wire up any safety-critical systems in an appropriate fashion (e.g., SCRAM loops in a nuclear power reactor).

### Hardware Basis

The DAQ hardware is based the data acquisition needs commonly found in TRIGA™ Research Reactors. The design considered the maximum data acquisition requirements for the most complex reactor in existence and then tripled or quadrupled those requirements (in terms of input and output channels) and that became the basis for the design. Because of the generous requirements (up to 576 digital inputs [or outputs], up to 160 relay-controlled outputs, up to 128 analog inputs, and up to 32 analog outputs) the DAQ system should be more than capable for the vast majority of data acquisition projects someone could come up with.

### User Interfaces

This document assumes that the DAQ system will be controlled by some host PC. As such, most of the user interface design associated with the DAQ system will occur on the host PC. However, certain user-interface components are present on the DAQ hardware. In particular, all boards will contain a power LED indication power applied to the board (or not). Furthermore, most digital I/O devices provide LEDs for each bit of digital I/O indicating whether the digital I/O pin is active or inactive.

### Hardware Interfaces

The DAQ system provides several hardware interfaces to external (to the DAQ) components and systems. As noted earlier, the DAQ system provides Ethernet, USB, and RS-232 interfaces to host computer systems. On the hardware side there are also interfaces to digital inputs, interfaces to digital outputs, interfaces to analog inputs, interfaces to analog outputs, and interfaces to other modules using the I2C and SPI busses.

#### Ethernet Interfaces

The Netburner and Raspberry Pi 3 Model B single-board computers (SBCs) both provide on-board Ethernet interfaces. These SBCs both contain an operating system with a full TCP/IP stack allowing easy communication across the Ethernet using standard socket communications.

The Teensy 3.2 module does not have a built-in Ethernet interface. However, it is possible to purchase an SPI-based Ethernet interface and wire it to the SPI bus (PPDIO96 bus) on the DAQ\_IF interface board if Ethernet access is desired when using a Teensy 3.2 SBC. Note, however, that such connections are beyond the scope of this document.

#### USB Interfaces

The Teensy 3.2 SBC includes a small micro-USB connector. You may connect the DAQ\_IF board (with Teensy 3.2 installed) to a host computer using a USB-to-micro-USB cable. To the host computer, the Teensy 3.2 looks like a really fast COM port.

In theory, the Netburner can also be programmed to act as a USB device. This document will not consider that option.

The Raspberry Pi 3, while it supports USB, cannot be used as a USB device (it's a USB host). Newer Raspberry Pi Zero units can be programmed as a device (using USB On-the-go) but this document will not consider that option.

### Software Interfaces

There are software libraries available for the Teensy 3.2 and Raspberry Pi 3 Model B SBCs that allow you to easily program all the I/O chips in the DAQ system. However, the main software written to support the DAQ system runs on the Netburner MOD54415 Evaluation board plugged into the DAQ\_IF board. The software interface for this system is completely described in the SRS (Software Requirements Specification), SDD (Software Design Description), and DAQ Operations manual.

### General Constraints

* Hardware or software environment – The majority of the software is written in the C/C++ programming language. In addition, there are various "make" files and system scripts used to build or run the system. Some test software is written in C++ running on the Teensy 3.2 or Raspberry Pi, but the majority of the system code runs under µC/OS on the Netburner.
* The software is developed under Windows using the Netburner software development kit and software development tools.
* Availability or volatility of resources – Run within the confines of the Netburner MOD54415 hardware.
* Interface/Protocol Requirements – The basic intertask communication will be handled by Ethernet socket communications.
* Data repository and distribution requirements – In a preemptive multitasking operating system, shared data objects must be protected to insure all access is serialized. Typically, this is done with semaphores, mutexes, or critical sections. To provide the necessary data access protection for globals, the code will include a global semaphore to be used by most tasks. This method will insure serialized data access.
* Memory and other capacity limitations - There will be no memory constraints other than the total amount of system RAM installed MOD54415 module. As the software runs on µC/OS, the system limitations 64 MB RAM and 2 MB Flash/ROM.
* Performance Requirements – The system shall perform all necessary tasks well within the performance capabilities of the MOD54415. During normal operations, all normal I/O and user interface tasks much be completely in less than 25% of the CPU cycles available. Occasional exceptions, lasting no more than a few seconds (less than the DAQ\_IF watchdog timeout period), are acceptable as long as these events are infrequent.
* Network Communications – The system shall use TCP/IP Ethernet communications protocols to communicate between the DAQ system and other devices.
* Testing – Testing will follow the standard Plantation Productions, Inc., practices for testing systems, as defined by the HTP (Hardware test procedures) and STP (Software Test Procedures) documents.
* Environmental -- The software shall perform properly while the unit is operating within the environmental constraints for the hardware.

#### Operating System

The DAQ software will utilize the µC/OS pre-emptive priority-based multitasking operating system kernel for microprocessors. µC/OS was not developed by Plantation Productions, Inc., and may be considered Software of Unknown Provenance (SOUP). However, µC/OS is open-source and its implementation on the Netburner modules has a long history so there is a high confidence in the reliability of this software.

The host computers contain their own software. As the software for those devices is outside the scope of this document, it shall not be considered here.

#### Criticality of the Application

The hardware may be used as part of a research reactor. The hardware's design allows the support of safety critical systems with an appropriate verification and validation (V&V) process in place.

#### Personnel Safety and Security Considerations

The hardware must be designed to protect the safety of personnel who use and maintain it.

### Goals and Guidelines

**Basic hardware goals:**

The hardware is designed to operate as programmed under normal circumstances. In anamolous situations (power up, software hang) the hardware shall enter a special fail-safe mode in which all digital I/O ports that are programmable as digital inputs or outputs revert to inputs and all strict digital output ports are programmed in the "off" (or "open") state. To fully realize this in a system, the following guidelines apply:

1. The PPDIO96 board should be programmed as input only. Although the system will automatically reset the I/O expander chips to input pins on a reset or watchdog timeout condition, keep in mind that if the pins are actually used as outputs the pins will be floating at that point and won't be in a guaranteed (fail-safe) state immediately after the reset or timeout.
2. Before writing to a digital output board (PPRELAY-12 or PPSSR-16) the system software should first verify that a watchdog timeout has not occurred (which will force all the outputs to "off" or "open"). If a timeout has occurred, then the system should reinitialize all the digital outputs to some reasonable state.

### Developmental Methods

The developmental method used to design the DAQ system hardware was based loosely on the Prototype/Iterative Model. In this Model, being a process, the following phases were followed in order:

1. Requirements Specification
2. Hardware Design
3. Circuit board creation and assembly
4. Testing (validation)
5. Repeat steps 1-4 for each new feature or site port.

# Hardware Design Description

## Design Stakeholders

The initial design of the DAQ system hardware was to support the data acquisition requirements for the Dow TRIGA Research Reactor (DTRR) at Dow Chemical in Midland Michigan. For this project, the project stakeholders are the following:

* Dow Chemical (site management)
* Dow Chemical (site operators/users)
* NRC
* Plantation Productions, Inc., project management
* Plantation Productions, Inc., software engineering/development
* Plantation Productions, Inc., hardware engineering/development
* Faircloth Engineering (systems hardware engineering development)
* Plantation Productions, Inc., software quality assurance

Although this design was originally created for DTRR and Dow Chemical, the design is sufficiently generic to be adopted for a wide variety of embedded applications. As such, the stakeholders can easily be swapped for more stakeholders more pertinent to a given application.

## Design Concerns

### Dow Site Management

The DAQ design shall provide like-for-like functionality of the existing data acquisition system for DTRR.

### DTRR Operators

The console design shall provide a reasonable approximation of the legacy QNX console to reduce the learning curve. This is achieved by providing the like-for-like functionality of the existing console using a similar user interface. In particular, the DAQ system will provide a transparent replacement for obsolete ISA-bus analog and digital I/O boards.

### NRC

The DAQ shall follow software quality assurance requirements for research reactors as specified by ANSI 15.15-1978.

### DAQ Project Management

The design shall allow implementation, testing, and deployment of the console on schedule and under budget

### DAQ Hardware Engineering/Development

The hardware engineering team shall have the following design concerns:

* The design shall provide the functionality required by the SOW, SyRS, and HRS.
* The design shall be efficient in terms of space, power usage, and functionality.
* The design and implementation shall be reliable.
* The design and implementation shall be maintainable.

### DAQ Hardware Engineering/Testing

The hardware testing team shall have the following design concerns:

* The design shall produce an appropriate set of test cases.
* The design and hardware shall be such that the system is easy to test.

# DAQ Detailed Design

## PPAIO-16/4 16-Channel Analog Input/4-Channel Analog Output Board

### Capacity, Interface, and Accuracy

1. [DAQ\_HDD\_032]

The PPAIO-16/4 board provides uses four ADS1115 modules to provide 16 channels of unsigned 15-bit single-ended analog input (four input channels per ADS1115). Alternately, a system designer may combine pairs of analog inputs to obtain up to 8 channels of signed 16-bit double-ended analog input. The ADS1115 ADCs provide a programmable gain amplifier supporting input voltages of 0..5V, 0..4.096V, 0..2.048V, 0..1.024V, 0..0.512V, or 0..0.256V. Normally, the DAQ system operates these chips in the 0..4.096V range as this is the voltage range produced by the PPAC-4 analog conditioning modules. Note that the full 16-bit input range is only available when using double-ended inputs; the single-ended inputs do not allow negative numbers and are limited to 15 bits.

The ADS is capable of processing as many as 860 samples/second. It may, however, be operated at a lower sample rate with higher accuracy. The requirements only call for 100 SPS, so operating at the ADS1115 default rate of 128 SPS is perfectly reasonable.

The PPAIO-16/4 also supports up to four MCP4725 DAC modules with additional analog amplification (on board) to convert the DAC's 12-bit 0..5V analog output to the range -10..+10V. The DAC's output sample rate is largely determined by the I2C interface, which (while slow) is certainly capable of exceeding the 100 SPS output rate specified by the requirements.

The accuracy of the ADS1115 IC greatly exceeds that of the circuit board layout. The system requirements call for 5% accuracy. In practice such a low accuracy was specified to allow automated testing using the DAC output to feed the ADC inputs. Given that such a test fixture is going from a 12-bit (unsigned) output voltage to a 15-bit (unsigned) input voltage with varying reference voltages across the power plane on the circuit board, the 5% was very liberal allowing wide variations between the two chips.

For demo/prototyping purposes, using sockets and headers for the ICs and breakout boards (BoBs) is probably acceptable, but for actual installation everything should be soldered onto PPAIO-16/4 circuit board (the BoBs should have right-angle headers soldered onto them and those headers should be soldered onto the PPAIO-16/4 PCB).

For the quietest and most accurate readings, you should use double-ended inputs and run twisted-pair cables from the signal source (or a PPAC-4 analog conditioning board) to the screw terminal inputs on the PPAIO-16/4 PCB. For convenience, the PPAIO-16/4 board includes a pair of 8-pin (2x4) headers that mate with similar connectors on the PPAC-4 boards. However, if low noise and high accuracy is desireable, do not use these connectors and ribbon cables to connect the PPAC-4 to the PPAIO-16/4 – use twisted pair (and better yet, shielded twisted pair) cabling for the interconnect.

The 8-pin header uses the following pin-out:

|  |  |  |  |
| --- | --- | --- | --- |
| Pin | Description | Pin | Description |
| 1 | AIN1+ | 2 | GND1 |
| 3 | AIN0+ | 4 | GND0 |
| 5 | AIN3+ | 6 | GND3 |
| 7 | AIN2+ | 7 | GND2 |

(If you're wondering, the out-of-sequence pinout produces cleaner trace paths on the circuit board, avoiding crossing other independent signals.)

One other thing to consider is your power supply. The PPAIO-16/4 runs off a ±12V power supply. This must be a "clean" power supply. Preferably one that is separate from those powering the digital boards in the DAQ system and especially one that is not connected to relay boards or other high-current demand circuits with lots of transients. Such boards will create spikes on the power supply lines that will affect your analog readings.

By following these rules you can obtain much better accuracy than that dictated by the PPAIO-16/4 requirements. Those requirements are suitable for the original intended purpose of the PPAIO-16/4 (various analog inputs from real-world sensors that aren't that accurate). Those requirements are easily improved upon, when needed, by carefully passing analog signals around in your design.

The ADS1115 and MCP4725 ICs and related components are surface mount devices (SMDs). A design goal for the DAQ system is to avoid using SMDs so that the boards can be tested, maintained, modified, and repaired in the field. The exception to this rule is that SMDs in the design are okay if commerically available (open hardware) breakout boards for the chips are available. Adafruit provides breakout boards for both these chips, Sparkfun also provides a BoB for the MCP4725 IC. The DAQ design uses the BoBs from both of these suppliers.

### PPAIO-16/4 I2C Interface, Port Assignment, and Daisy-Chaining

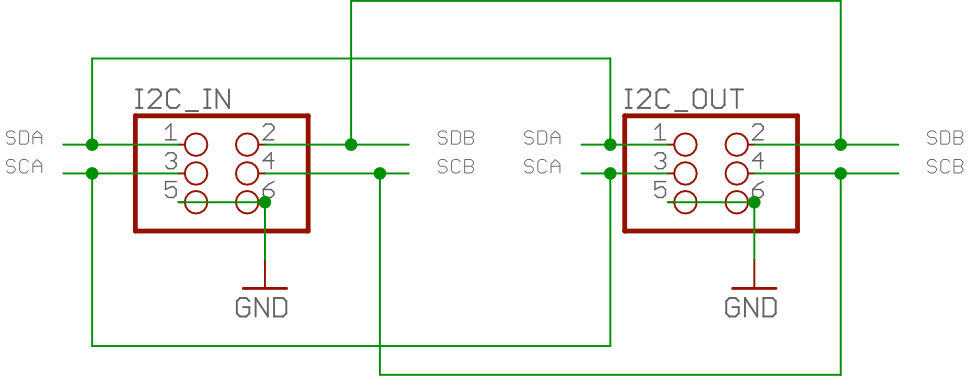
1. [DAQ\_HDD\_033]

The PPAIO-16/4 board has two I2C bus connections: an input connection and an output connection. The header contains data lines for two independent I2C busses. These connections are 6-pin (2x3) headers with the following pinouts:

|  |  |  |  |
| --- | --- | --- | --- |
| Pin | Description | Pin | Description |
| 1 | I2C SDA port A (data) | 2 | I2C SDA port B (data) |
| 3 | I2C SCL port A (clock) | 4 | I2C SCL port B (clock) |
| 5 | Gnd | 5 | Gnd |

On a given PPAIO-16/4 the ADC modules only use the A port. The DAC chips use both I2C ports.

The I2C input header pins are not routed directly to the I2C output header pins. Instead, the circuit swaps the Port A and Port B pin between the input and output headers:



This input/output I2C bus arrangement allows you to daisy-chain two PPAIO-16/4 boards on a single DAQ\_IF I2C bus connector and the two boards (subject to certain construction limitations) will operate on separate I2C busses.

The ADS1115 ADC chips are I2C bus devices that can operate at one of four separate I2C addresses. A single address pin (ADDR) on the ADS1115 connects to one of the following pins (also on the ADS1115) to select one of the following addresses:

* ADDR to Gnd: I2C address 0x48 on I2C bus A
* ADDR to Vcc (+5v): I2C address 0x48 on I2C bus A
* ADDR to SDA: I2C address 0x4a on I2C bus A
* ADDR to SCL: I2C address 0x4b on I2C bus A

There are four Adafruit ADS1115 modules on each PPAIO-16/4 board (numbered 1..4, corresponding to addresses 0x48 through 0x4b, respectively). Each Adafruit module supports 4 single-ended or two double-ended analog inputs. The corresponding AINn (*n*=0..15) input port assignments on a single PPAIO-16/5 board are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Module # | I2C address | AIN Ports (single-ended) | AIN Ports (double-ended)[[1]](#footnote-1) |
| 1 | 0x48 | AIN0..AIN3 | AIN0/AIN1, AIN2/AIN3 |
| 2 | 0x49 | AIN4..AIN7 | AIN4/AIN5, AIN6/AIN6 |
| 3 | 0x4a | AIN8..AIN11 | AIN8/AIN9, AIN10/AIN11 |
| 4 | 0x4b | AIN12..AIN15 | AIN12/AIN13, AIN14/AIN15 |

When you have two PPAIO-16/4 boards daisy-chained together, the first board (directly connected to the DAQ\_IF) responds to I2C addresses on the first (A) I2C bus, the second board (connected to the first PPAIO-16/4) responds to I2C addresses on the second (B) bus.

There are four MCP4725 DAC modules on the PPAIO-16/4 board. As these chips can only respond to two different I2C address, the four modules use both I2C busses. The Adafruit MCP4625 BoBs consume I2C addresses 0x62 and 0x63. The corresponding output port assignments are as follows:

|  |  |  |
| --- | --- | --- |
| Module # | I2C address | AOUT Ports |
| 1 | 0x62: bus A | AOUT0 |
| 2 | 0x63: bus B | AOUT1 |
| 3 | 0x62: bus A | AOUT2 |
| 4 | 0x63: bus B | AOUT3 |

The MCP4725 BoB modules from Sparkfun use different I2C addresses and have the following output port assignments:

|  |  |  |
| --- | --- | --- |
| Module # | I2C address | AOUT Ports |
| 1 | 0x60: bus A | AOUT0 |
| 2 | 0x61: bus B | AOUT1 |
| 3 | 0x60: bus A | AOUT2 |
| 4 | 0x61: bus B | AOUT3 |

To select between the two addresses (0x60/0x61 or 0x62/0x63) you connect the ADDR pin on the MCP4725 to Vcc (+5V) or Gnd. Connecting ADDR to Vcc selects the odd address of the pair, connecting ADDR to Gnd (or left floating) selects the even address:

|  |  |  |
| --- | --- | --- |
| ADDR tied to: | Adafruit address | Sparkfun address |
| Gnd | 0x62 | 0x60 |
| Vcc | 0x63 | 0x61 |

Note that the ADDR pin is accessed differently on the Adafruit and Sparkfun MCP4725 modules. On the Adafruit BoB the ADDR pin is brought out to one of the header pins; on the Sparkfun board the ADDR pin is accessible via a pair of solder pads on the circuit board.

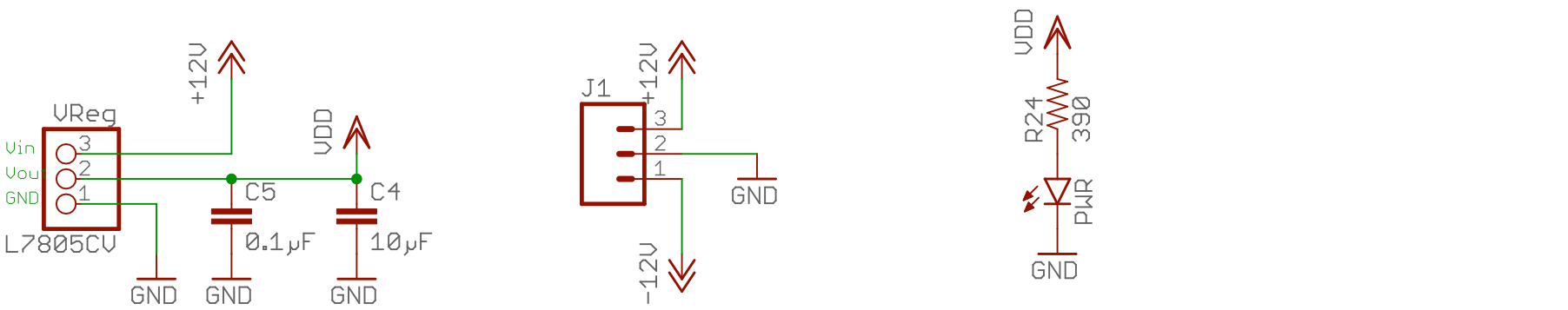
If you daisy-chain two PPAIO-16/4 boards on the same DAQ\_IF I2C bus, the first board must have Adafruit MCP4725 DAC BoBs installed and the second must have Sparkfun MCP4725 DAC BoBs installed. This prevents I2C address conflicts. Technically, you could install two Adafruit and two Sparkfun BoBs on a single board and have that board operate off a single I2C bus. While this might make better sense from a certain point of view, the current design allows you to install four PPAIO-16/4 boards in the system (no daisy-chaining) utilizing parts bought from a single vendor (Adafruit). Note that other topologies are possible. However, the software written for the DAQ system assumes all Adafruit or all Sparkfun DACs on the same board and Adafruit DACs on the first board and Sparkfun DACs on the second. As a general rule, you should connect the first four PPAIO-16/4 boards directly to the DAQ\_IF board and only daisy-chain them if you need more than four PPAIO-16/5 boards in the system (the standard software expects this topology).

Note that the PPAIO-16/4 board doesn't bother buffering the I2C lines. This is largely because there are never more than a few I2C devices on each I2C bus (e.g., 4 ADCs and 4 DACs) so the buffering isn't really necessary.

### PPAIO-16/4 Power Supply

1. [DAQ\_HDD\_034]

The PPAIO-16/4 receives ±12V power and ground on a three-pin screw terminal. As with most DAQ system boards, there is an LED (indirectly) connected to the +12V power supply to indicate when power is applied.



Because the Adafruit and Sparkfun breakout boards (BoBs) run off +5V, there is a linear voltage regulator on the board (L7805CV) that will convert the +12V supply to +5V. This regulator is good for 1A which is more than sufficient to power the LED, Bobs, and other 5V circuitry.

Because the PPAIO-16/4 involves sensitive analog circuitry (ADCs and DACs are very sensitive to voltage levels and fluctuations) it is very important to supply the board with good, clean, highly-filtered power. There should be no heavy loads (particularly heavy intermittent loads) and certainly no inductive loads (e.g., motors, magnets, or relays) on the same power supply. Even putting digital circuits on the same power supply is not a very good idea. The best solution is to have a dedicated ±12V power supply just for the analog components in the DAQ system.

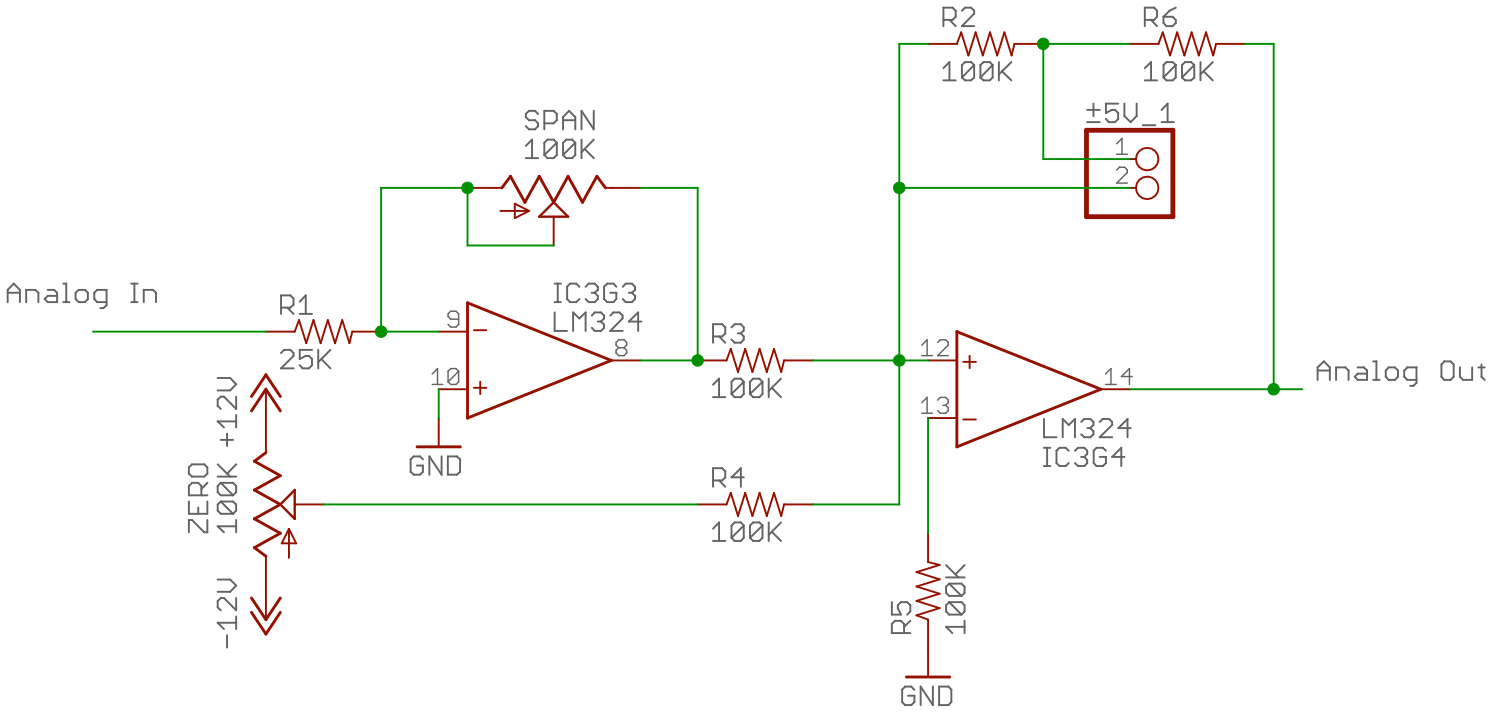
Because there are four separate power busses on the PPAIO-16/4 board (+5V, +12V, -12V, and Gnd) with only four layers on the circuit board, the non-ground voltages share the same circuit board layer. The power busses were manually laid out to deliver needed voltages wherever necessary without using vias or traces. The only exception to this rule was to use a 100 mil wide trace to deliver +12V to the 5V voltage regulator.

Note that ground planes were placed on the top and bottom layers of the circuit board to provide shielding and help eliminate noise. This is in addition to the ground plane found on the interior of the PCB.

### PPAIO-16/4 Analog Output Amplifiers

1. [DAQ\_HDD\_035]

The PPAIO-16/4 contains four opamp circuits that convert the 0..5V output from the DAC chips into either a ±5V signal or a ±10V signal (jumper programmable):



The first opamp in this circuit has an adjustable gain (nominally 2) that converts the 0..5V input (from the DAC) to 0..10V (gain is adjustable with the SPAN pot). The second opamp in the circuit (with no jumper installed) allows setting the offset such that the output is ±10V (with no jumper, the gain is 2). Installing the jumper block on the ±5V header halves the feedback resistance (which halves the gain) reducing the output voltage to ±5V. There are four copies of this circuit on the PPAIO-16/4 – one for each DAC on the board.

Note that the LM324 opamp used in this design easily provides the 1kHz bandwidth and linear response specified by the requirements. The SPAN pot provides a gain (span) adjustment for the circuit and the ZERO pot provides a voltage offset (zero) function.

## PPAC4 4-Channel Analog Conditioning Board

### Capacity and Interface

1. [DAQ\_HDD\_036]

The PPAC4 board provides four channels of analog conditioning and isolation. Each channel accepts a singled-ended ±10V analog input, isolates it, and converts the input to a double-ended/differential signal (0..4.096V on each side) suitable for input to a PPAIO-16/4 board.

The board provides four two-pin screw terminals as input connectors and also provides four two-pin screw terminals as output connectors.

There is an 8-pin (2x4) header on the board connected to the input signals. This header matches the output header pin-out on the PPAC420 (4-20 mA conversion board) and has the following pin-out:

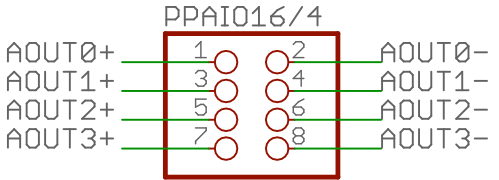
|  |  |  |  |
| --- | --- | --- | --- |
| Pin | Description | Pin | Description |
| 1 | AIN1+ | 2 | GND1 |
| 3 | AIN0+ | 4 | GND0 |
| 5 | AIN3+ | 6 | GND3 |
| 7 | AIN2+ | 7 | GND2 |

(If you're wondering, the out-of-sequence pinout produces cleaner trace paths on the circuit board, avoiding crossing other independent signals.)

There is also an 8-pin (2x4) header containing the four differential output signals. You can use a small ribbon cable to connect the PPAC4 directly to a PPAIO-16/4 board. However, keep in mind that ribbon cables tend to be noisy, so if you are interested in a low-noise connection between the PPAC4 and PPAIO-16/5 boards (or between the PPAC4 and PPAC420 boards), you should use a twisted pair cable between screw terminals on the boards. In theory, using the differential pairs will minimize (if not eliminate) cross talk in the cables, but twisted pairs rather than ribbon cables is always quieter.

PPAC4 8-pin output header pin-out (connection to PPAIO-16/4 board):

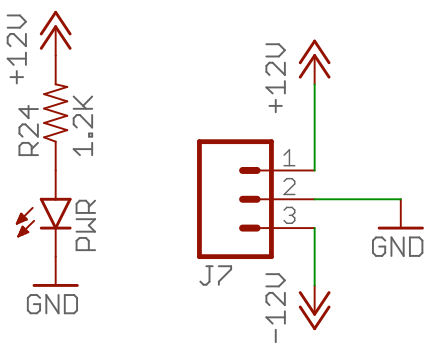
|  |  |  |  |
| --- | --- | --- | --- |
| Pin | Description | Pin | Description |
| 1 | AIN0+ | 2 | AIN0- |
| 3 | AIN1+ | 4 | AIN1- |
| 5 | AIN2+ | 6 | AIN2- |
| 7 | AIN3+ | 7 | AIN3- |



### PPAC4 Power Supply

1. [DAQ\_HDD\_037]

The PPAC4 receives ±12V power and ground on a three-pin screw terminal. As with most DAQ system boards, there is an LED connected to the +12V power supply to indicate when power is applied.



Each channel contains an isolation amplifier that requires an isolated ±12V power supply. Therefore, the board contains four isolated power supplies (DC to DC converters) that create ±12V for use by the isolation amplifiers.

Because the PPAC4 involves sensitive analog circuitry (opamp circuits are very sensitive to voltage levels and fluctuations) it is very important to supply the board with good, clean, highly-filtered power. There should be no heavy loads (particularly heavy intermittent loads) and certainly no inductive loads (e.g., motors, magnets, or relays) on the same power supply. Even putting digital circuits on the same power supply is not a very good idea. The best solution is to have a dedicated ±12V power supply just for the analog components in the DAQ system.

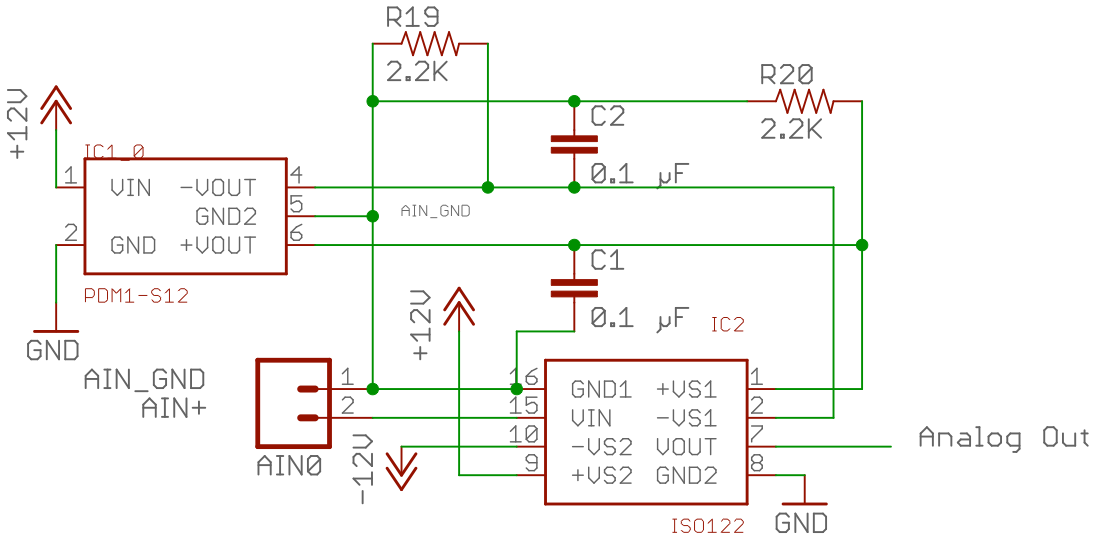
Because the PPAC4 electrically isolates the incoming signals from the amplification and power systems, the ground and power planes are somewhat complex on the circuit board. You do not want the circuit board's normal ground and power planes going underneath the isolated signals on the circuit board. This could create inductive and capacitive coupling (granted it's tiny, but present nonetheless) to be avoided in an isolated circuit. Therefore, the power and ground planes on the interior layers underneath each isolated circuit are from the isolated power supply for that particular circuit.

### PPAC4 Isolation Amplifier Section

1. [DAQ\_HDD\_038]

The PPAC4 contains an ISO122 isolation amplifier for each channel. As noted in the previous section there is an isolated ±12 DC to DC converter, running off the PPAC4's +12V, supplying power to the isolation amplifier.

The PDM1-S12 DC-to-DC converter used in the design requires at least a 10% load to produce a correct voltage output. To achieve this, the PPAC4 design places a pair of 2.2K resistors between the (isolated) ±12V leads and the (isolated) Gnd pin on each PDM1-S12. This consumes about 0.07W and the pair of them consume 0.14W. As the device is a 1W device, this corresponds to about 14% power.



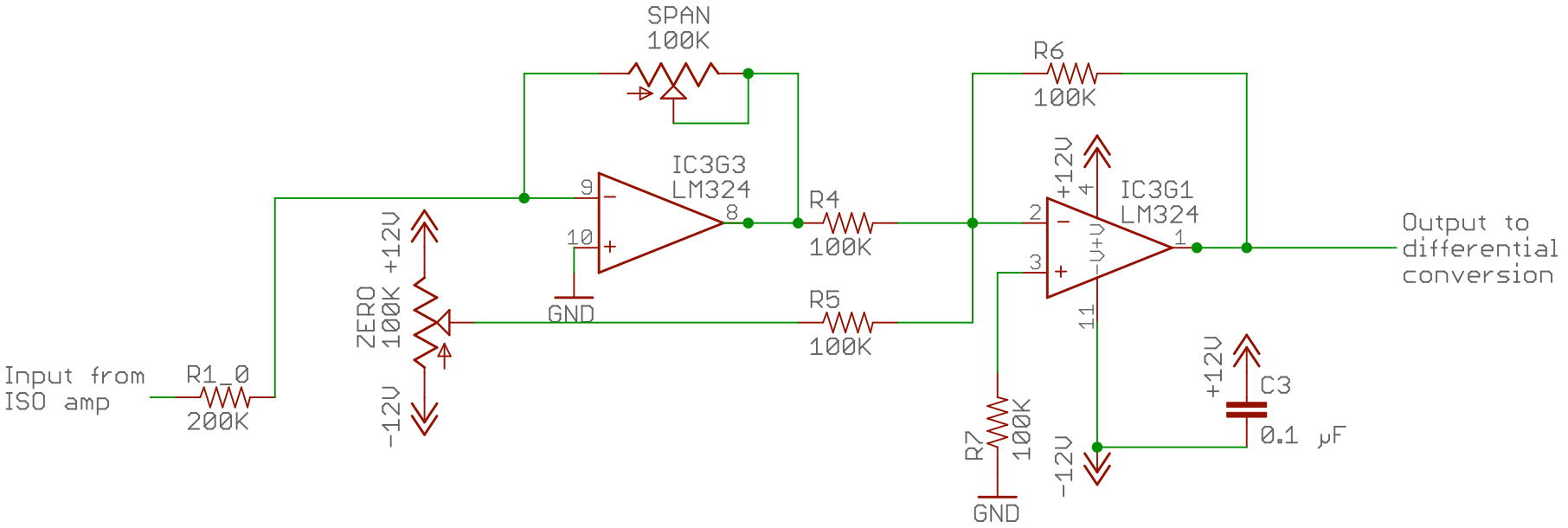
The ISO122 amplifier accepts the isolated analog input and copies it to the isolated output with a gain of 1. The ISO122 is a relatively low-end isolation amplifier. The gain will likely not be exactly one and there might be a voltage offset (positive or negative) added to the signal. The amplifier section following the isolation amplifier will compensate for these issues. The ISO122 offers a 50 kHz bandwidth; this is far more than the 1 kHz bandwidth specified by the PPAC4 requirements.

### PPAC4 Amplifier Section

1. [DAQ\_HDD\_039]

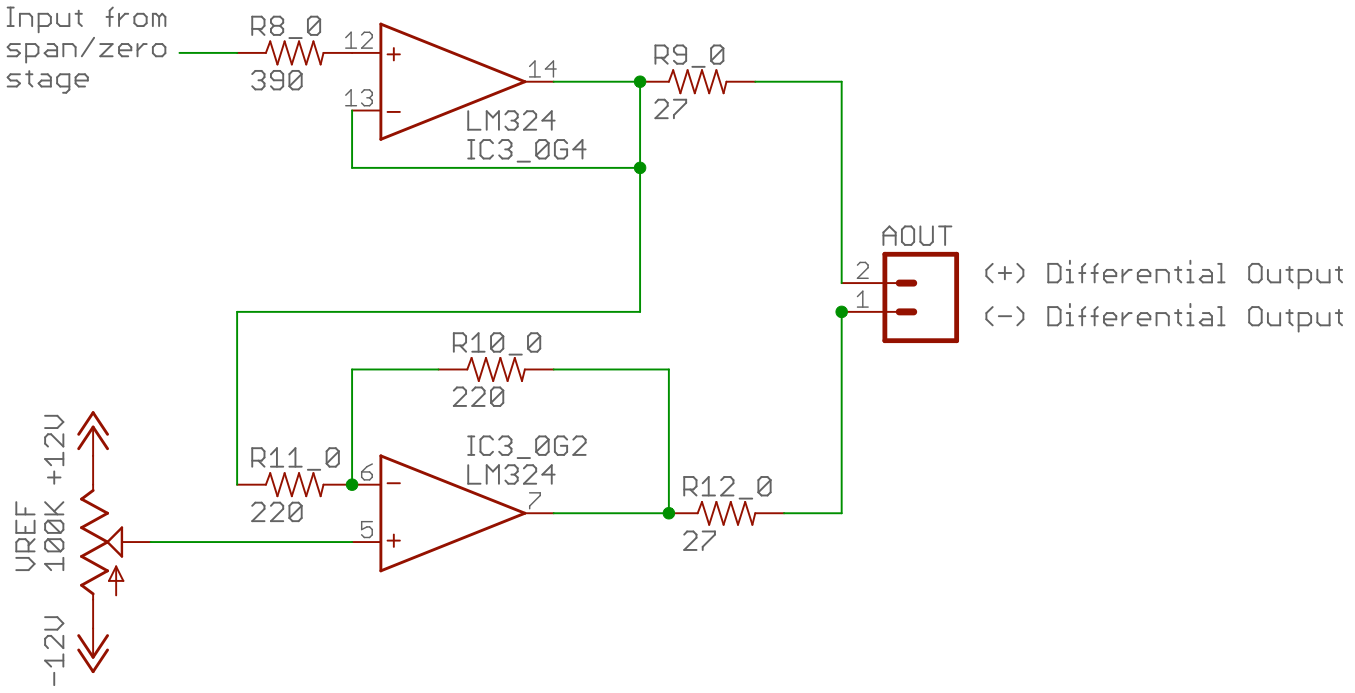
The PPAC4 contains two amplifier circuits, each consisting of a pair of opamps (all part of an LM324 quad opamp package). The first pair of opamps provides gain (span) and offset (zero) adjustments. They also convert an input signal in the range ±10V to the range ±2.048V. The second pair of opamps accept the ±2.048 output from the first amplifier stage and convert that signal to a pair of differential 0..+4.096V outputs. Note that all opamps operate in a linear fashion, thus satisfying the requirement that the transfer function is linear.

Note that the (nominal) resistor values in the first opamp of the amplifier section have a 50:200 ratio (assuming the span pot is exactly in the middle, producing 50K Ω). This produces a gain of ¼. For a ±10V input, this would produce a ±2.5V output. In fact, the PPAC4 board has to be calibrated to produce a ±2.048V output (which the second state of the amplifier section converts to a 0..4.096V differential signal). The span/gain pot handles the discrepancy. Rather than having the span pot exactly in the middle of the range (and adjusting it to obtain exactly ±2.5V), the idea is to increase the resistance by about 20% additional to bring the output voltage down to ±2.048V.



The second opamp in the circuit exists simply to adjust the offset voltage (that is, to set the zero point). Most likely, the ISO122 isolation amplifier will add some positive or negative offset to the input voltage. Putting the zero pot on a separate opamp helps make it slightly more independent of the gain/span pot. Also, the second opamp reinverts the signal (though this could have also been done in the following double-ended output conversion by simply swapping the signal lines).

The double-ended differential outputs are always positive (with respect to the power supply ground). When representing negative voltages (i.e., -10V to just below 0V) the negative diffierential terminal is greater than the positive differential terminal by the specified voltage. For example, -10V is converted to 4.096V on the negative terminal and 0V on the positive terminal. +10V is converted to 0V on the negative terminal and 4.096V on the positive terminal (note: 4.096V represents a full-scale reading, which corresponds to 10V).



The differential opamp stage (the last two opamps) contains a VREF adjustment pot. This sets the voltage that the system will use to represent 0V on the differential output. The nominal value to shoot for is 2.048V. That is, 2.048V appearing on both sides of the differential output will represent 0V (this generally implies that 4.096V on the negative side with 0V on the positive side represents -10V and 0V on the negative side with 4.096V on the positive side represents +10V, with both sides ranging from 0V to 4.096V (the range of the ADC inputs that provide the maximum precision).

## PPAC420 8-Channel 4-20 mA Analog Input Conditioning Board

### Capacity and Interface

1. [DAQ\_HDD\_040]

The PPAC420 board provides eight channels of 4-20 mA conversion. Each channel accepts a 4-20 mA input (on two-pin screw terminals) and the board produces eight single-ended voltage outputs in the range -1.25 to 20 mA. Note that a 4 mA input produces a 0V output.

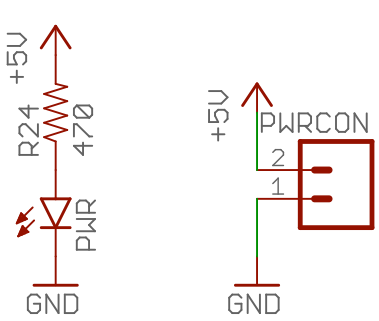
The PPAC40 board produces outputs that would normally be fed as inputs to the PPAC4 board. Because the PPAC420 provides eight channels and the PPAC4 board supports only four channels, it will take two PPAC4 boards to handle all the outputs from the PPAC420. As the outputs from the PPAC4 are differential/double-ended outputs, it will take a single PPAIO-16/4 (which supports 8 double-ended inputs) to handle all the output channels from the PPAC420.

In addition to the two-pin screw terminals, the PPAC420 also provides a pair of 8-pin (2x4) headers containing the analog output signals. These headers are compatible with the 8-pin input headers on the PPAC4 board, allowing a single ribbon cable to connect the two boards (two ribbon cables from the PPAC420 can connect to two PPAC4 boards). Of course, twisted pair wiring from the PPAC420 screw terminal outputs to the PPAC4 screw terminal inputs is probably a tiny bit quieter, but you can use ribbon cables if you don't need an ultra-quiet signal.

### PPAC420 Power Supply

1. [DAQ\_HDD\_041]

The PPAC420 receives +5V power and ground on a two-pin screw terminal. As with most DAQ system boards, there is an LED connected to the +5V power supply to indicate when power is applied.



Each channel contains an isolation amplifier that requires an isolated ±15V power supply. Therefore, the board contains eight isolated power supplies (DC to DC converters) that create ±15V for use by the RCV420 ICs used on the board. Note that each channel is electrically independent of all the others.

As is the case with nearly all DC-to-DC converters/isolated power supplies, the xxxx converters the PPAC420 uses require a minimum 10% load to produce a well-regulated output. As such there are some load resistors between the ±15V power supplies and ground to provide a minimum load.

Because the PPAC420 involves sensitive analog circuitry it is very important to supply the board with good, clean, highly-filtered power. There should be no heavy loads (particularly heavy intermittent loads) and certainly no inductive loads (e.g., motors, magnets, or relays) on the same power supply. Even putting digital circuits on the same power supply is not a very good idea. The best solution is to have a dedicated +5V power supply just for the analog components in the DAQ system.

## PPRLYIO-12 12-Channel I/O Relay Board

### Capacity and Interface

1. [DAQ\_HDD\_042]

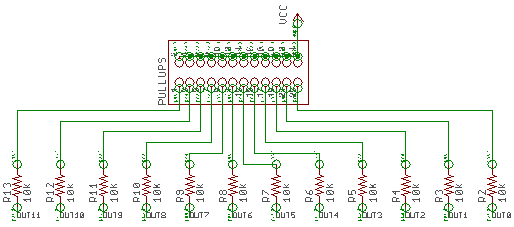
The PPRLYIO-12 board provides 12 channels of mechanical relay control for digital input/output devices. It provides 12 sets of screw terminals (three terminals each: NO/COM/NC – normally open/common/normally close) as outputs.

The board utilizes JZC-11F-05VDC – 1Z relays that contain both normally open and normally closed contacts. There are board-layout-compatible relays available that drop the NC contact (providing only a normally-open contact) as well.

The PPRLYIO-12 board provides two PPDIO96-bank-compatible 20-pin headers – one containing a set of (TTL-compatible) input signals and one containing a set of output signals (jumper programmable as open collector or TTL-compatible pins). These connectors each have the following pinout:

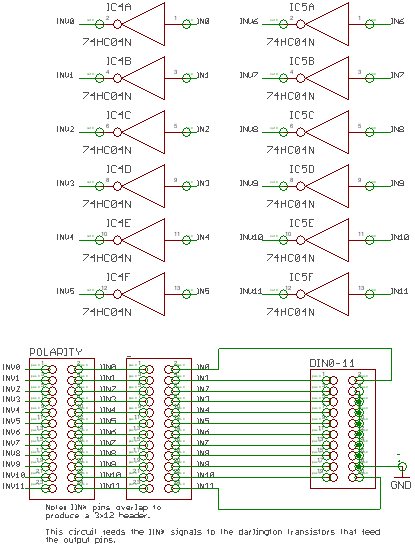
|  |  |  |  |
| --- | --- | --- | --- |
| PPRLYIO-12 Input and Output Pinouts | | | |
| Pin | Description | Pin | Description |
| 1 | D1 | 2 | D0 |
| 3 | D2 | 4 | Gnd |
| 5 | D3 | 6 | Gnd |
| 7 | D4 | 8 | Gnd |
| 9 | D5 | 10 | Gnd |
| 11 | D6 | 12 | Gnd |
| 13 | D7 | 14 | Gnd |
| 15 | D8 | 16 | Gnd |
| 17 | D9 | 18 | Gnd |
| 19 | D10 | 20 | D11 |

The output pins can be (jumper) programmed as open collector or TTL-compatible outputs. The board uses 1-½ ULN2308 8-channel darlington transistor array packages to control the output pins. An active-high signal on the darlington input creates an open circuit which ties the output pin to ground. By itself, this provide an open-collector circuit. There is a set of 12 pairs of jumpers on a header than provide pullup resistors on each outpt channel. If the jumper is in place, this connects a pullup resistor (10kΩ) to +5V for the corresponding channel. In this case, the output pin produces a TTL-compatible output signal of +5V when the darlingon is not actuated, and Gnd when the darlington is switched on. Note that this effectively inverts the output (producing +5V when the input is low and 0V when the input is high).



The input conectors go to a 3x12 (36-pin) header where jumpers can select the input (directly) or an inverted version of the input. If a jumper is installed across the IIN\* and INV\* pins, then the jumper selects the output of a 74HC04 hex inverter IC and feeds an inverted version of the signal to the darlington. If a jumper is installed across the IIN\* and IN\* pins, then an uninverted version of the signal is fed to the darlington input.

Note that if you're operating a given PPRLYIO-12 channel in TTL-compatible mode (that is, the corresponding jumper is installed on the pullups header), then the output signal will normally be inverted. Selecting the IIN\* and INV\* pins on the polarity header will invert the signal prior to the darlington so that the output produces a signal with the same polarity as the input.

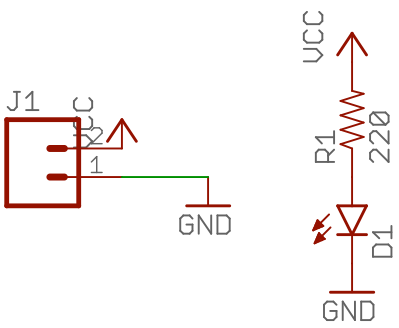


Note: the header is actually 3x12, not 4x12 as the schematic might suggest. The IIN\* holes in the circuit board overlap. The schematic was drawn this was to make it easy to show the IIN\* signals.

### PPRLYIO-12 Power Supply

1. [DAQ\_HDD\_043]

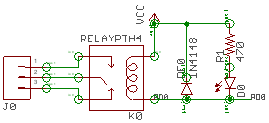
The PPRLYIO-12 I/O board runs off a +5V power supply. As with most DAQ system boards, the RLYIO-12 I/O board illuminates an LED when power is applied to the board.



### PPRLYIO-12 Relay and Data LED Activation

1. [DAQ\_HDD\_044]

The PPRLYIO-12 I/O board includes three ULN2308 8-channel darlington transistor array packages. Half the channels (12) provide buffering to the output connector. The remaining 12 channels control the relays and the channel LEDs on the PPRLYIO-12.



(the collector on the darlington transistor output connects to the RD0 signal in the schematic above.)

When the darlington input is activated (with a logic high signal) the transistor creates a closed circuit that shunts the collector (RD0, above) to ground. This turns on the relay and also turns on the associated data LED. When the darlington input is unactivated (with a logic low signal) the darlington presents an open circuit and no current flows through the LED or relay (meaning the LED is off and the relay is in the inactive position with the NO contact open and the NC contact closed).

As with any inductive circuit, there is a flyback diode across the relay coil terminals to shunt any inductive kickback away from the rest of the circuit (note that the ULN2308 darlington array also includes such diodes, but good design dictates always putting a diode right on the coil terminals, anyway).

By its design, the relay isolates the NC/COM/NO contacts from the rest of the circuit.

The board utilizes JZC-11F-05VDC – 1Z relays that provide up to 5A current handling capability (derated to 2.5A by good engineering design principles plus PCB trace limitations). These relays can also handle up to 30 VDC (actually, they are rated up to 125VAC but DAQ system design limits voltages to 30 VDC).

1. Double-ended ports connect their positive terminal to AIN0, AIN2, AIN4, AIN6, AIN8, AIN12, or AIN14. They connect their negative terminals to AIN1, AIN3, AIN5, AIN7, AIN9, AIN11, AIN13, or AIN15 (odd/even pairs). Double-ended signals do not use the Gnd pins on the screw terminals. [↑](#footnote-ref-1)