NOAA Technical Memorandum GLERL-154

SYSTEM POWER CONTROLLER: A LOW POWER CIRCUIT BOARD FOR THE CONTROL AND MONITORING OF SUBSYSTEM POWER IN DATA COLLECTION SYSTEMS

Ronald Muzzi
Stephen Constant
John Lane

NOAA, Great Lakes Environmental Research Laboratory
4840 S. State Rd., Ann Arbor, MI 48108

February 2012

UNITED STATES
DEPARTMENT OF COMMERCE

John Bryson
Secretary

NATIONAL OCEANIC AND
ATMOSPHERIC ADMINISTRATION

Jane Lubchenco
Under Secretary for Oceans & Atmosphere
NOAA Administrator

NOTICE

 This is GLERL Contribution No. 1611.

This publication is available as a PDF file and can be downloaded from GLERL's web site: www.glerl.noaa.gov. Hard copies can be requested from GLERL Information Services, 4840 S. State Rd., Ann Arbor, MI 48108. pubs.glerl@noaa.gov.

NOAA's Mission – To understand and predict changes in Earth's environment and conserve and manage coastal and marine resources to meet our nation's economic, social, and environmental needs.

NOAA's Mission Goals:

- Protect, restore and manage the use of coastal and ocean resources through an ecosystem approach to management
- Understand climate variability and change to enhance society's ability to plan and respond
- Serve society's needs for weather and water information
- Support the Nation's commerce with information for safe, efficient, and environmentally sound transportation
- Provide critical support for NOAA's Mission

TABLE OF CONTENTS

System Power Controller: A low power circuit board for the control and monitoring subsystem power in data collection systems

Ronald Muzzi
Stephen Constant
John Lane

ABSTRACT

Power efficiency is important in remote data collection systems that are typically solar-powered. Systems such as the Real-time Coastal Observation Network (ReCON) require controlling power to sensors and components with currents of up to 2 amps and voltages up to 48 volts. Though there are some off-the-shelf solutions available, these systems either consume unnecessary power, or are limited in the amount of current and voltage that they can handle. The design of the System Power Controller board provides eight channels of semiconductor-switched power that can handle the current and voltage requirements while consuming a low amount of quiescent power. Each power channel provides high-side switching of up to 60 volts and currents of up to 2 amps. A low-power microprocessor using an RS-232 serial interface allows programming of the channels, including time delayed events. Three analog input channels allow measuring system voltages, such as solar panel, battery, and system bus voltages. The controller includes a watchdog timer with relay output which allows a full power reboot of the data collection system. The system can be operated with any voltage from 7 to 60 volts, and the entire controller only consumes 0.14 watts of power when powered at 12 volts. The controller board conforms to the PC/104 standard form factor so that it can be mounted on the top of a standard PC/104 stack. The system has proven reliable in several years of field use and has been incorporated as an integral component of the ReCON data collection network.

1. BACKGROUND

Power efficiency is an important requirement of a remote multi-sensor data collection platform, such as the ReCON buoy[1]. These systems are typically off the power grid and therefore limited by battery capacity, available solar energy, or other energy harvesting methods. One method of improving power efficiency is to operate the various components of the system, such as sensors and communication links, on a duty cycle. The duty cycle is adjusted for each component according to its particular requirements. Sensors measuring slowly changing oceanographic parameters do not need to be sampling at the same rate as sensors measuring rapidly changing parameters. The communications link, which usually consumes a large amount of power to transmit the data to a remote receiver, only needs to be powered during data transmissions. Some components, such as sensors, may only consume a few milliamps of power, while others, such as communications, may consume a few amps of power. In addition, these components often operate at various voltages, typically ranging anywhere from 5 to 60 volts. An efficient power controller module is needed to manage the power of these components. This controller must include multiple power channels, at least eight, that are independently programmed. The controller would be programmed by the data collection platform processor, responding to commands sent from the processor over an RS-232 serial link[2]. Since the PC/104 is a common form factor[3] used in data collection and other embedded systems, it is preferred that the controller board also conform to this form factor.

1 RUBERG, S.A., R.W. MUZZI, S.B. BRANDT, J.C. LANE, T.C. MILLER, J.J. Gray, S.A. CONSTANT, and E.J. Downing. A wireless internet-based observatory: The Real-time Coastal Observation Network (ReCON). Proceedings, Marine Technology Society/IEEE Oceans 2007 Conference, Vancouver, British Columbia, Canada, September 30-October 5, 2007, 6 pp. (2007). http://www.glerl.noaa.gov/pubs/fulltext/2007/20070045.pdf

2 The Electronic Industries Association (EIA) standard RS-232 (Recommended Standard) defines the electrical characteristics for serial asynchronous communications between devices, e.g., what is commonly referred to as a computer serial port.

3 For information on the PC/104 Consortium see http://www.pc104.org/.

At this time, though there are off-the-shelf solutions that can approximate these power control functions, we have not been able to find any that meet all of the above requirements. There are three common methods of controlling power: relays, low-end switching semiconductors, and high-end switching semiconductors. Off-the-shelf systems are available employing all of these methods. Systems using relays, however, consume power ranging on the order of tens of milliamps per channel. This, however, is the same order of magnitude as some of the sensors themselves, so that the use of relays would not be very power efficient. Systems using low-end switching cause problems with oceanographic sensors since they introduce an offset into the negative side of the power supply, resulting in an unknown voltage differential between the sensor ground and the system ground. This in turn affects the sensor measurements. The available off-the-shelf systems using high-end switching semiconductors are limited to a single voltage, usually 12 volts, and limited in their current output, typically 100 milliamps or less. These limitations are overcome in our design by allowing separate input voltages for each channel and selecting semiconductors capable of handling both high voltages, up to 60 volts, and high currents, up to 2 amps.

The design would then also include some extra features that would benefit the needs of data collection systems. Several analog input channels would be included to monitor system voltages such as solar panel and battery voltage. A watchdog timer would be included with a relay to allow “rebooting” the system if the system should lock-up. And an open-drain power channel would be included to provide a logic-control channel.

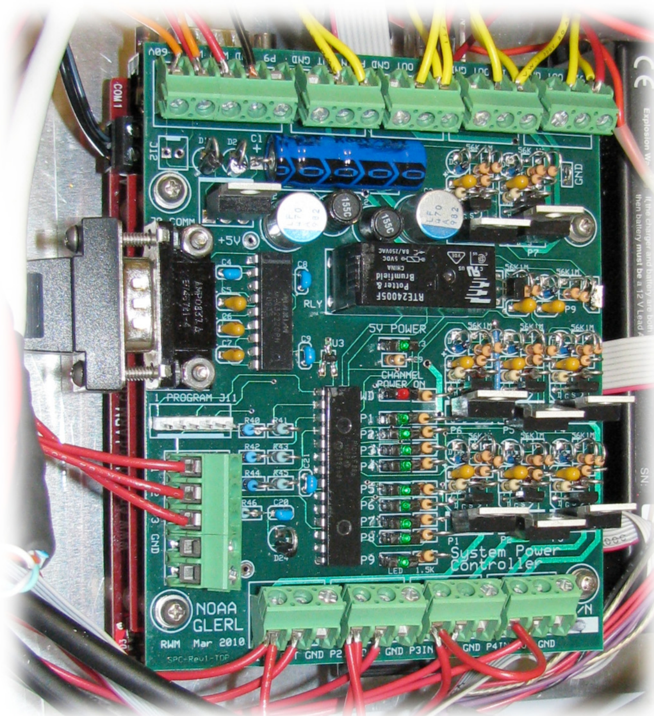

Figure 1. System Power Controller wired in ReCON buoy on PC/104 stack.

2. DESIGN

The controller was designed so that it would conform to the PC/104 form factor[4] and mount on a PC/104 stack as the top board, although the overall dimensions are slightly larger than the PC/104 specifications. It was decided not to use the PC/104 bus for power or communications, though available, because that would limit the controller operations to only be able to function when the PC/104 processor board was running and healthy. Keeping the power and communications separate also allows the controller to power cycle the PC/104 processor board if needed.

For optimum reliability and simplicity, the controller is powered directly from the system battery. Since some of our systems run on 12 volts and others on 48 volts, we designed the controller to have a wide-range input voltage of 7-60

4 For information on the PC/104 Consortium see http://www.pc104.org/.

volts. This would cover the low end of about 9 volts from a well-drained 12 volt battery to about 54 volts from a solar panel charging a 48 volt battery.

Communications to the controller is through an RS-232 serial connection. The controller will receive specific commands to turn power channels on and off, turn power channels on and off after a delay, read the analog input channels, and reset the watchdog timer. The controller uses non-volatile memory to store power-up channel states and calibration values for the analog inputs.

The power channels were designed with MOSFET[5] p-channel transistors to provide high-end voltage switching that are capable of handling up to 60 volts and up to 2 amps. High-end switching is preferred over low-end switching, because low-end switching can produce ground differentials. The amount of voltage drop resulting from high-end switching is insignificant, since systems are already designed to accommodate the large voltage swings seen in the battery/solar panel charge/discharge cycle. In addition, each power channel will have its own input and output terminal to allow the use of different voltages on different channels as needed.

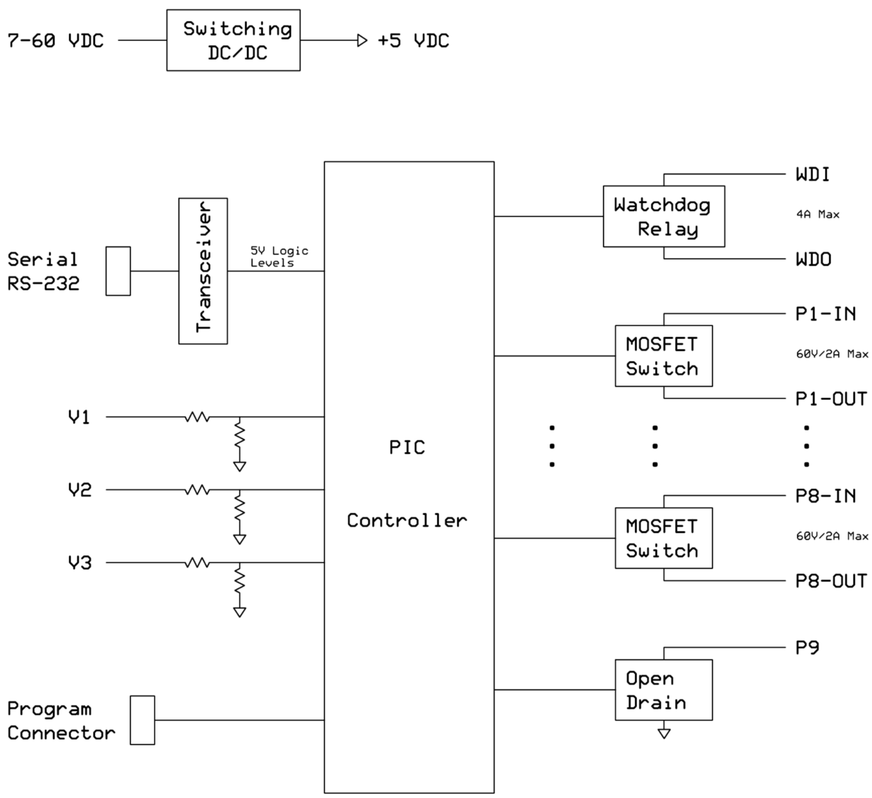

Figure 2. Function Block Diagram.

The controller includes three analog inputs to allow monitoring of various system voltages, such as solar panel, battery, and bus voltages.

A watchdog timer is provided that can be used to power cycle the system in the event of a software lockup. If the watchdog timer doesn't receive regular resets from the data logger through the serial port within the given timeout period, it pulses a relay that momentarily turns off the power to the data logger or entire system. This allows the system to self-reboot in the case of a system lockup. The relay operates in a fail-safe condition, that is, the normally-closed state.

Since the controller board will be populated in-house, the board was designed to use thru-hole components rather than surface mount components since the thru-hole components can be more easily soldered. The only exception to this was for the three-pin voltage reference, which was not available in a thru-hole package.

5 MOSFET means Metal–Oxide–Semiconductor Field-Effect Transistor.

2.1 Board Form Factor

The standard size for a PC/104 form factor board is specified[6] as 3.6 x 3.8 inches (9.1 x 9.7 cm). It soon became apparent that a board of that size would not provide enough area for all the components that we desired to place on it. Since the stack of PC/104 boards that we were using contained one oversized board measuring 3.8 x 4.8 inches (9.7 x 12.2 cm), we decided to approximate that size in order to provide the desired area, and therefore, we used a design size of 3.775 x 4.750 inches (9.6 x 1.9 cm). This would allow the boards to stack uniformly.

2.2 Electrical Design

The electrical design[7] was developed in-house to meet our power controlling requirements and still be versatile enough to be used in a wide range of applications.

2.2.1 Power Switching

To provide power switching, a FQP47P06 p-channel MOSFET transistor (Q2) was selected to provide reliable, low quiescent current switching between the IN terminal and the OUT terminal (Figure 3). Screw terminals were used to provide easy connection to the board. The MOSFET transistor has a rated RDS[8] of 0.026 Ω. In a typical application at 1 amp, this would result in a voltage drop of only 0.026 volts. The transistor is rated for 47 amps current, VDSS[9] of 60 volts, and 160 watts power dissipation. Our maximum power dissipation is $I^2 \times R = (\ 2 \text{ amps}\ )^2 \times (\ 0.026\ \Omega\ ) = 0.1$ watts, which is well handled by the TO-220 package[10] of the transistor. Since the VGSS[11] is only rated at ±25 volts, a 15 volt zener diode, D3, is placed across Q2 to limit the voltage. R4 is used to limit current flowing through the zener diode. Q1 drives the gate of Q2 low when it is powered on by the control signal. R5 is used to bias Q2 gate high when Q1 is turned off.

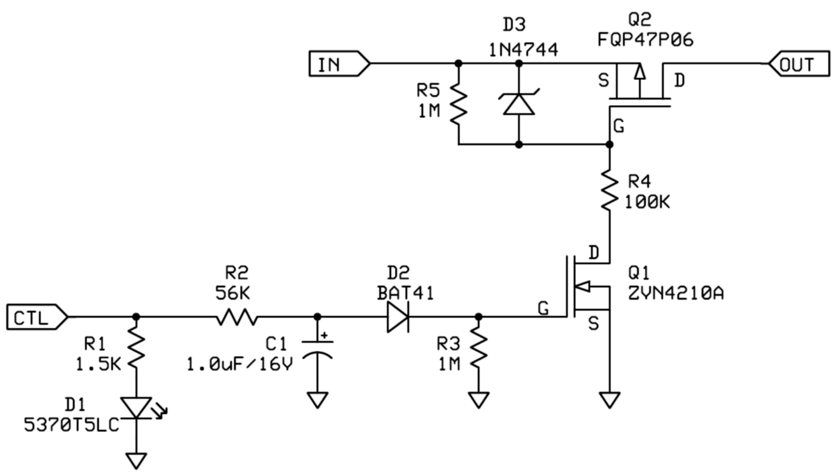

Figure 3. Power Switching Circuit.

R3 is used to bias Q1 gate low when no control input is present. Since power up spikes may be present in the control input, a low-pass filter made up of R2 and C1 is used. This provides a minimum input pulse requirement on the order of tens of milliseconds based on the time constant of 56 KΩ x 1.0 μF = 56 msec. This is more than adequate to prevent false switching during power-up transients. D2 is included as a failsafe to protect the control circuitry from high voltage, which may occur in the event of Q1 failing in a shorted mode between the gate and drain. Light emitting diode (LED) D1, along with the current limiting R1, provide a channel "on" indicator. The 5370T5LC LED was selected because it requires only

6 MOSFET means Metal–Oxide–Semiconductor Field-Effect Transistor.
7 Schematic available here: http://www.glerl.noaa.gov/ftp/publications/tech_reports/glerl-154
8 RDS is the Resistance between the Drain and the Source pins of the transistor
9 VDSS is the Voltage between the Drain and the Source pins of the transistor.
10 The TO-220 package is the Industry standard Transistor Outline number 220.
11 VGSS is the Voltage between the Gate and Source pins of the transistor.

2 mA to illuminate the LED. However, if needed, these LEDs could be left out for applications that require extremely low current requirements. Eight power channels, P1-P8, are provided with this arrangement.

One power channel, P9, is provided as an open-drain power channel for the use of controlling logic signals. It includes all of the above components, except for Q2, R4, R5, and D3.

Another power channel, P0, controls the watchdog relay (Figure 4). The input circuit is similar to the above, except that D2 is not included since there is no high voltage being applied to Q1 in this circuit.

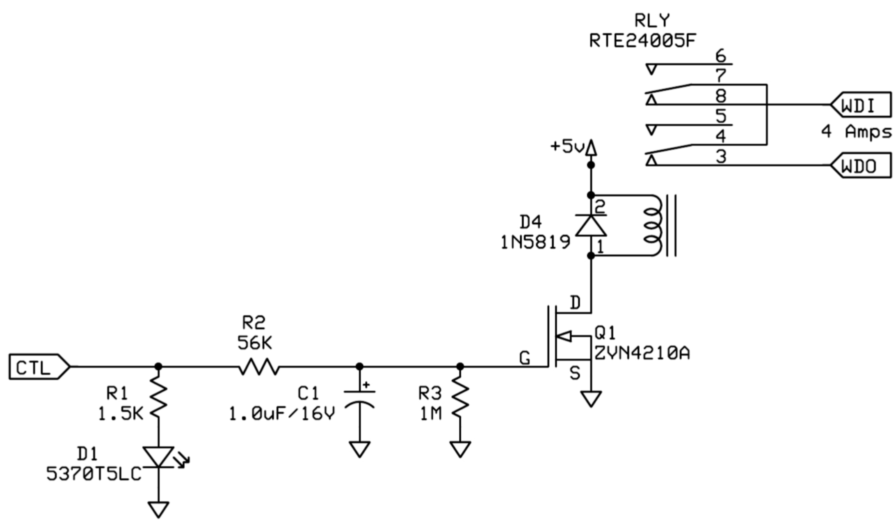

Figure 4. Watchdog Relay Control Circuit.

The drain of Q1 connects one end of the relay coil to ground. The other end of the relay coil is connected to the 5 volt supply. D4 is used across the coil as a snubber diode. Two branches of the relay are used, instead of one, in order to accommodate the switching of high voltages. The normally closed contacts are used to provide the fail-safe on condition for the watchdog reset. Since the coil only consumes power during the brief reset condition, the coil power requirements are generally not significant in the power consumption budget. A red LED was used for the relay power channel to distinguish it from the other channels, which use green LEDs because of their capabilities, low power consumption, and availability in a through-hole package.

2.2.2 PIC Processor

The Peripheral Interface Controller (PIC) 16F886 CMOS microprocessor[12] was chosen. This processor provides 24 Input/Output (I/O) lines, 10-bit Analog to Digital Converter (A/D), serial communications, three timers, an internal clock, and is available in a thru-hole package. It has 8192 words of program memory, 368 bytes of Random Access Memory (RAM), and 256 bytes of Electrically Erasable Programmable Read-Only Memory (EEPROM). It requires 2.0 to 5.5 volts to power it and consumes a maximum of 1.4 mA current at the 4 MHz operating frequency.

2.2.3 A/D Input

The 10-bit A/D input is designed for clean, slowly varying, low-impedance input sources such as solar panel voltage, battery voltage, and system bus voltages (figure 5). A 3.000 volt reference is provided by a MAX6030 low-current voltage reference.

12 CMOS is Complementary Metal–Oxide–Semiconductor.

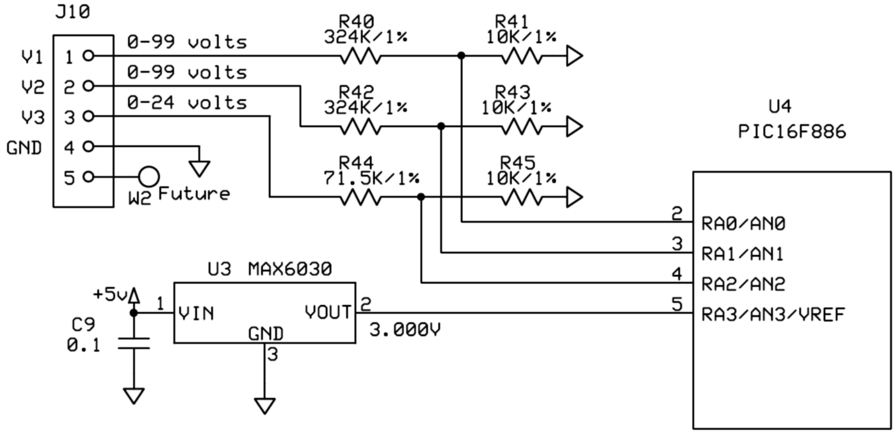

Figure 5. A/D Input Circuit.

Resistor dividers are used in a half-bridge configuration to allow measurement of high voltages. The first two analog channels use 10K and 324K Ω low thermal drift resistors to allow inputs of up to about 100 volts with a resolution of about 0.1 volts. The third analog channel uses 10K and 71.5K resistors to allow inputs of up to about 24 volts with a resolution of about 0.02 volts.

2.2.4 RS-232 Interface

The PIC microprocessor provides logic-level serial communications (Figure 6). The MAX3222 RS-232 Transceiver is used to translate between RS-232 level signals and logic-level signals. The transceiver allows input levels of up to ±25 volts and provides output levels of ±5 volts.

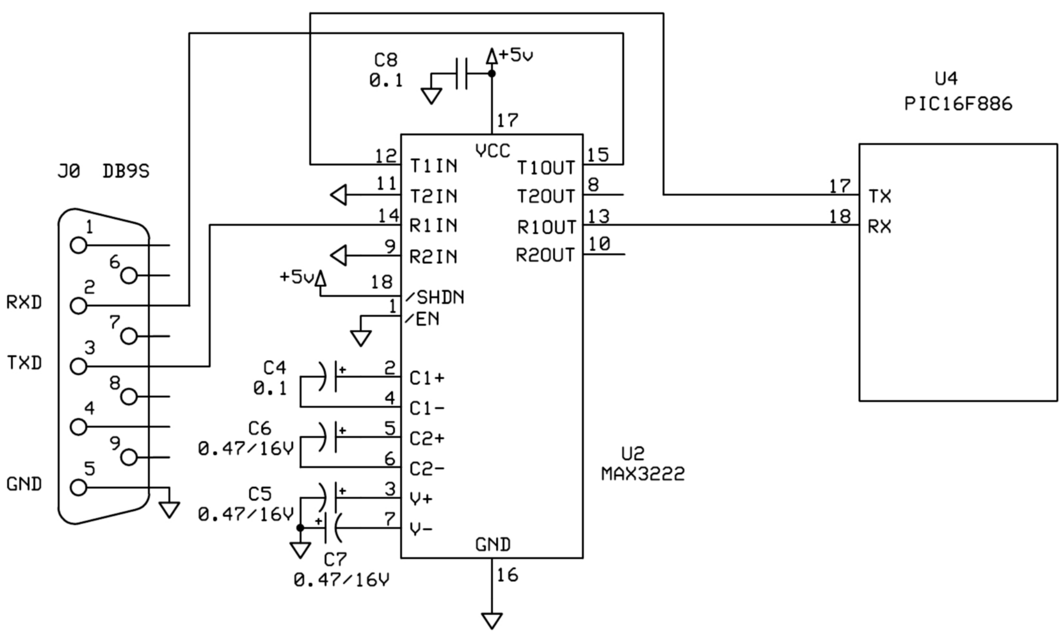

Figure 6. RS-232 Interface Circuit.

Only lines TX, RX, and GND are supported. A DE9F connector[13] is mounted directly to the board to provide a standard serial port connector.

13 DE9F is a D-subminiature series, E shell size, 9 pin, Female gender connector.

2.2.5 Switching Power Supply

For low-power operation, a wide-range input voltage switching power supply was employed to deliver 5 volts at a peak of 100 mA to the circuit (Figure 7). The LM2575HV-5.0 step-down voltage regulator was used, which provides an input voltage range from 7 to 60 volts. The components C1, C2, L1, and D2 are specified in the LM2575 data sheet. D1 is added to protect against reverse polarity inputs. The low-pass filter made up of L2 and C3 reduce the switching frequency transients to an acceptable level.

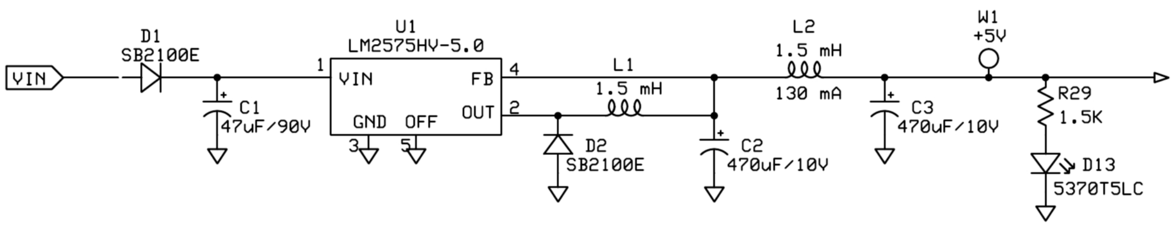

Figure 7. Switching Power Supply Circuit.

The LED, D13, provides the power-on indicator. Though the output voltage of the switching power supply will deviate slightly in direct correlation to the input voltage over the 7-60 volt range, the output voltage does not exceed the PIC microprocessor power supply limit of 5.5 volts.

2.2.6 Programming Interface

The PIC microprocessor can be programmed using a PICkit 2 USB interface[14]. This connects between a programming computer Universal Serial Bus (USB) interface and a 6-pin connector on the board (figure 8). The 6-pin connector uses standard 0.1 inch spacing between pins.

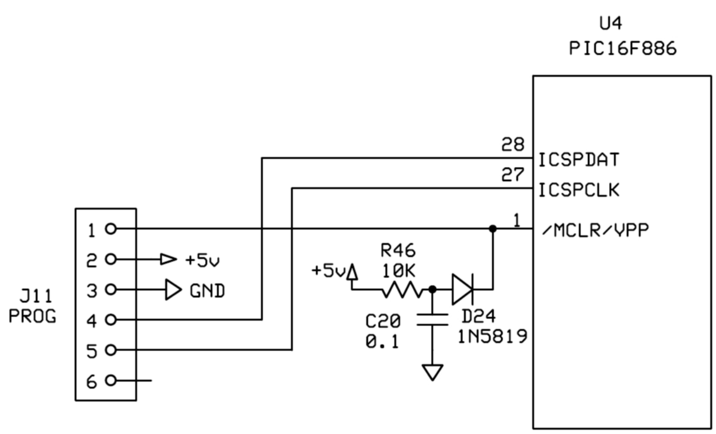

Figure 8. Programming Interface.

C20, D24, and R46 make up the required interface, as specified in the PIC16F886 data sheet.

2.3 PCB Layout Design

The printed circuit board (PCB) layout[15] was designed in-house using software provided by the circuit board manufacturer[16]. A 4-layer design was used with solder mask and silkscreen. Two of the four layers are for power and ground planes. The power plane was connected to the 5 volt supply.

14 Manufactured by Microchip Technology, Inc., http://www.microchip.com/
15 PCB layout available here: http://www.glerl.noaa.gov/ftp/publications/tech_reports/glerl-154
16 ExpressPCB, http://www.expresspcb.com.

A trace width of 0.01 inches (0.25 mm) was used for most logic and power traces. For the power switching paths requiring a capacity of 2 amps, a trace width of 0.05 inches (1.27 mm) was used. For the watchdog relay contact traces requiring a capacity of 4 amps, a trace width of 0.1 inches (2.54 mm) was used. A minimum spacing between traces of 0.02 inches (0.51 mm) was used. For traces carrying high voltages, a minimum spacing between traces of 0.03 inches (0.76 mm) was used.

Oversized pads were used on U3, the 3-pin surface mount MAX6030 Voltage Reference, in order to facilitate hand soldering of the component. Two test points were provided: one for the 5 volt supply and another for ground. A jumper was provided to connect the input voltage directly to the watchdog timer relay contacts. The silkscreen included a white box for writing the controller serial number on the board.

The controller boards are populated by hand in house. After assembly, the boards are spray coated with acrylic plastic (while covering the connectors) to provide protection against moisture for extreme environmental use.

2.4 Software Design

The PIC 16F886 microprocessor was programmed in the C language. The program[17], named "System Power Controller.c", is compiled and written to the PIC's non-volatile program memory via a PIC software development kit[18]. Comments in the program source code provide sufficient information to understand the program operation and also to facilitate making changes if needed.

2.4.1 Main Routine

The first task of the main routine is to initialize the PIC microprocessor. This includes setting the PIC registers, starting the RS-232 communications, reading stored settings from the EEPROM memory, setting power output channels to their default states, and initializing the general purpose timer interrupt. The controller then waits for and processes commands from the user received over the RS-232 port.

2.4.2 Command Processing

After connecting the controller to a terminal emulator program running at a baud rate of 9600, the `Return` key is pressed to get the command prompt and confirm that communications are working properly. The controller sends out the prompt, "`SPC>`". Pressing the `Return` key causes the prompt to be displayed again. The first letter of each command determines which routine will process it. There are separate routines for Delay, EEPROM, Help, Power, Query, Voltage, and Watchdog timer commands. The RESET command is processed by branching to the start of the main routine so that all the PIC initializations are executed again. Commands are processed immediately, with the exception of some of the power commands. Power commands that include delay times are entered into a table to provide independent timing to each power output channel.

---

17 Software program available here: http://www.glerl.noaa.gov/ftp/publications/tech_reports/glerl-154
18 PIC Software Development Kit is available from Microchip Technology, Inc.

Table 1. Controller Command Syntax List

```
System Power Controller 2.1 Commands:

  Dn             = Delay secs (0-2^32)
  P              = Show all pins
  Pn(L/H)        = Set pin 0-9 Lo/Hi
  Pn(L/H)d       = Wait d secs (0-65535), set pin L/H
  Pn(L/H)ddd,t = Wait ddd secs (0-999), pulse L/H for t secs (0-2^32)
  Q              = Query
  RESET          = Power up reset
  V(n)           = Show A/D (1-3)
  W(n)           = Show/set watchdog secs (1-2^32)
  Wx             = Watchdog: R=Reset, OFF=Stop, C=Clear reset #
  H              = Help List

EEPROM Commands:

  E         = Display all values
  EWSn      = Write serial number n (0-65535)
  EWPn      = Write default pin states, bits 0-9
  EWTn      = Write watchdog timeout default n secs (10-65535, 0=Off)
  EWRn      = Write watchdog reset time n secs (0-65535)
  EWCaMn    = Write calibration chan a (1-3) multiplier n (0-65535)
  EWCaZn    = Write calibration chan a (1-3) zero offset n (-32768 to 32767)
  EWBs      = Write boot commands (use ';' between commands)
  EWIn      = Write eeprom initialized flag: 12345=yes, other=no
  ERaaa     = Read word from address aaa (0-255)
  EWaaa,n  = Write word n (0-65535) to address aaa (0-255)
```

Examples of commands:

`SPC> V1`	→ Show voltage on V1 terminal
`SPC> P3H`	→ Set P3 high
`SPC> P1L30`	→ Wait 30 seconds, then set P1 low
`SPC> P2H020,15`	→ Wait 20 seconds, set P2 high for 15 seconds, then set P2 low
`SPC> WR`	→ Reset watchdog timer (to prevent watchdog timeout)

The `UART_Init()` routine was written to initialize the Universal Asynchronous Receiver/Transmitter (UART) RS-232 serial communications interface. The `putch()` routine is provided to interface the PIC serial I/O to the C `stdio` library for standard output processing. Serial input is processed by the `UART_Ch_Ready()` and `UART_Ch_Get()` routines to check whether a character is in the input stream buffer and to retrieve the character. The `GetString()` routine is used to retrieve an entire line of input as a string variable and provide the processing of special characters, such as `Backspace` and `Return`.

The controller features a boot command string, stored in non-volatile EEPROM memory, for the processing of commands at power-up. The processing of these commands is handled by the `GetString()` routine, including the use of the `Ctl-C` character to abort execution of the boot commands. The ";" character is used to separate lines in the boot command string. The boot command string allows power channels to be turned on according to a timed schedule. It can also be used to allow the system to operate in a stand-alone mode, i.e., without a computer to send it commands. To operate in the stand-alone mode, the boot command string would contain various timed commands and then end with the `RESET` command, which will cause the boot command string to be executed repeatedly and indefinitely. The `Ctl-C` character is used to abort execution of the boot command string if needed.

2.4.3 Watchdog Timer

The watchdog counter is a value that is decremented every second by the Timer 2 interrupt service routine. If the watchdog counter reaches 0, then the watchdog relay is energized for the specified reset time. The watchdog relay is typically connected between the system battery and the electronics so that when the watchdog relay is energized, the electronics are powered completely off, with the exception of the SPC controller. The `EWTn` command is used to set the watchdog time-out period in seconds. For example, `EWT3840` sets a 64 minute time-out period. The `WR` command must be sent to the controller on a regular basis to reset the watchdog counter to the initial value and prevent a reset. Both the initial watchdog timeout and reset time parameters are stored in EEPROM memory.

2.4.4 Timer 2

The PIC Timer 2 provides the heart of the timing needed by the controller. Timer 2 is programmed so that it generates an interrupt at a rate of 25 times per second. The Timer 2 interrupt routine repeatedly counts down from 25 in order to provide one second timing to the timing routines. There are three sections to the timing routines: the delay counter, watchdog processing, and power channel state processing.

The delay counter simply decrements the `delay_counter` variable every second until it reaches zero. This counter is then available to various commands to provided timing as needed.

The watchdog processing uses a state variable, `wd_state`, and a counter variable, `wd_secs`. At power-up and when the watchdog is reset by a user command, `wd_secs` is set to the watchdog timeout time, e.g., 3840 seconds. The variable `wd_state` is set to 1 to indicate that it is counting down the watchdog timeout time by decrementing `wd_secs`. When `wd_secs` reaches 0, the watchdog relay is energized, `wd_state` is changed to 2, and `wd_secs` is set to the watchdog reset time, e.g., 60 seconds. The variable `wd_state` equaling 2 indicates that it is counting down the reset time (how long the system is held in the reset state) by decrementing `wd_secs`. When `wd_secs` reaches 0, the watchdog relay is deactivated, `wd_state` is set back to 1, and `wd_secs` is set back to the watchdog timeout time. To disable watchdog processing, `wd_state` is set to 0.

The power channel state processing uses arrays, one for each channel. The power channel is also referred to as a pin, since they are controlled by the output pins on the PIC processor. The control arrays include a state variable, `pin_state[]`, a level variable, `pin_level[]`, and two counter variables, `pin_wait[]` and `pin_time[]`. They are indexed 0..9 to represent P0 through P9. (P0 can be controlled both via the watchdog commands and it can also be used as a regular power channel via the P commands.) These variables determine how to process each power channel at the one second interval.

A `pin_state[]` of 0 indicates that there is no processing that needs to be done for that pin.

A `pin_state[]` of 1 is used during the wait time period before changing the pin level. The counter `pin_wait[]` is decremented during this time. When `pin_wait[]` reaches 0, the pin is set to `pin_level[]`. If there is a pin set time period indicated by a non-zero `pin_time[]`, then `pin_state[]` is changed to 2. Otherwise, `pin_state[]` is changed to 0 to complete the processing.

A `pin_state[]` of 2 is used during the pin set time period. The counter `pin_time[]` is decremented during this state. When `pin_time[]` reaches 0, the pin is set to the inverse of `pin_level[]` and `pin_state[]` is set to 0 to complete the processing.

Here's a few examples of how the pin variables would be initialized according to specific power commands:

Set power channel (pin) 3 high after a 15 second delay:
Command: `P3H15`
```
pin_state[3] = 1
pin_level[3] = 1
```

```
pin_wait[3]  = 15
pin_time[3]  = 0
```

Set power channel (pin) 5 low after a 30 second delay, then after 20 seconds set it high again:
Command: `P5L030,20`

```
pin_state[5] = 1
pin_level[5] = 0
pin_wait[5]  = 30
pin_time[5]  = 20
```

Note, however, that before changing the array variable for a specific channel, `pin_state[]` must first set to 0 to clear and prevent any pin processing while the array variables are being changed. Then after the variables `pin_level[]`, `pin_wait[]`, and `pin_time[]` are set to their desired values, `pin_state[]` is set to 1 to start the processing.

2.4.5 EEPROM Storage

System parameters and calibration values are stored in the non-volatile EEPROM memory. The `EEPROM_Write_Word()` and `EEPROM_Write_String()` routines are provided to write to EEPROM memory, and the `EEPROM_Read_Word()`, `EEPROM_Read_String()`, and `EEPROM_Read_All()` routines are provided to read from EEPROM memory. The EEPROM commands (see Table 1) are provided to allow the user to write to and read the EEPROM parameters. In the table, the size of `word` is 2 bytes and the size of `char` is 1 byte.

Table 2 – EEPROM Memory Storage

Parameter	Address	Size	Description
EEPROM_SERIAL	0	word	Serial number
EEPROM_PINS	2	word	Pin default states
EEPROM_WD_TIMEOUT	4	word	Watchdog timeout default (secs)
EEPROM_WD_RESET_TIME	6	word	Watchdog reset time (secs)
EEPROM_CAL_V1_MULT	8	word	Calibration for V1, multiplier
EEPROM_CAL_V1_OFFSET	10	word	Calibration for V1, offset
EEPROM_CAL_V2_MULT	12	word	Calibration for V2, multiplier
EEPROM_CAL_V2_OFFSET	14	word	Calibration for V2, offset
EEPROM_CAL_V3_MULT	16	word	Calibration for V3, multiplier
EEPROM_CAL_V3_OFFSET	18	word	Calibration for V3, offset
EEPROM_INIT	20	word	EEPROM initialized (12345=yes)
EEPROM_BOOT_CMDS	22	char[64]	Boot commands string

2.4.6 A/D Converter

Three analog channels of A/D are available to provide monitoring voltage values. The `ReadVolt()` routine handles the reading of the specified analog channel and applying the calibration constants. The routine takes 1024 readings of the analog channel at a rate as fast as the A/D can sample. These readings are averaged and then a linear calibration is applied using the multiplier and offset calibration value for that analog channel. The result is an integer value in millivolts.

The calibration multiplier and offset are based upon the input resistor voltage divider. For analog channels 1 and 2, a 10K/324K Ω resistor half-bridge is used. The nominal calibration multiplier for these channels is:

$$VIN = (324K + 10K) / 10K \times 1024 \times VREAD = 34202 \times VREAD$$

The 1024 multiplier is used to provide better resolution under the integer arithmetic used by the PIC microprocessor. After calibrating, the value of 34202 is replaced with the calibration multiplier constant. For analog channel 3, a 10K/71.5K Ω resistor half-bridge is used. This results in the following nominal calibration multiplier:

$$VIN = (71.5K + 10K) / 10K \times 1024 \times VREAD = 8346 \times VREAD$$

The `V` command calls the `ReadVolt()` routine twice. The first call is to allow the multiplexer to select the analog channel and provide time for the channel multiplexer to settle down. The value from the first call is discarded and the value from the second call is used.

2.5 Operation

A typical operation involves connecting the controller to an RS-232 serial port on a remote data logger. The logger would then be programmed to send a `Return` character and confirm that it received the "`SPC>`" prompt. The logger would then send appropriate commands, such as turning on and off power channels, reading the voltage inputs, and resetting the watchdog timer. The power channels would be connected to various components of the system. The watchdog relay would be inserted between the battery and the system so that it provides a full power reset of the system. The controller, however, would be connected directly to the battery so that it is always powered.

2.6 Calibration & Initialization

The EEPROM values are initialized with the following commands:

```
EWSn          → Write serial number n, e.g., EWS4 writes serial number 4.
EWP0          → Set default pin states to all off.
EWT3840       → Set watchdog timeout to 64 mins.
EWR60         → Set watchdog reset time to 60 secs.
EWC1M34202    → Set default calibration values for A/D channels.
EWC1Z0
EWC2M34202
EWC2Z0
EWC3M8346
EWC3Z0
EWB           → Set empty boot commands string.
EWI12345      → Code to indicate EEPROM values are OK.
RESET         → Apply changes.
```

The controller is tested by operating each sub-system and confirming proper operation. A checkout procedure with log sheet[19] was written to guide and document the test. The switching transistors are tested at full current rating (2A) and full voltage rating (60V) simultaneously. The A/D channels are calibrated against a known voltage reference at four voltage input values spread throughout the operating range. A linear regression is done on the result to provide calibration constants, multiplier and offset, for each A/D channel. These calibration constants are stored in the EEPROM memory.

3. RESULTS AND LESSONS LEARNED

As of the writing of this report, we have two years of field use of the controller, involving 12 boards. For the most part, these controllers have worked successfully, providing rugged power control and monitoring of the ReCON platforms they have been used to control.

19 Checkout sheet available here: http://www.glerl.noaa.gov/ftp/publications/tech_reports/glerl-154

3.1 Electronic Design Lessons

The original design used on the first four controllers that we built did not include the zener diode over-voltage protection (see D3 of Figure 3) across the gate and source of the MOSFET power switching transistor. One of the power channels, operating at 48 volts, failed in the locked-on state because of this. Though it did not fail on the workbench or immediately in the field, the stressing of the gate-source junction over an extended period of time caused the transistor to finally fail. The zener diode was added to prevent this stress.

The MOSFET transistors were spaced too closely together on the board so that it was possible for the metal heat-sink tabs of adjacent transistors to touch each other. This was corrected by slightly bending alternate transistors so that the tabs wouldn't be directly next to each other.

3.2 Software Lessons

Two software bugs were discovered during the first year of use. One was a rounding error when outputting the voltages. The voltages are stored as integer values in millivolts but they are output in volts with two significant digits. The original code did not round the least significant digit, which results in biasing the value lower than actual. Code was written to round the value before displaying. The first attempt, however, contained a bug that caused values such as 12999 mV to display as 12.100 instead of 13.00. The code was corrected after the first field year.

The other software bug would occur in the case of a serial overrun on the RS-232 serial input. The PIC manual[20] states that after a serial overrun occurs, no further characters can be received until the CREN (receive enable) bit is cleared. The first versions did not handle this error. Code was added to test the OERR serial overrun flag and, if an overrun has occurred, to clear and then set again the CREN bit. This is, however, a rare error condition, and did not occur during any of our use of the boards.

After the first two field years, the software was improved to provide boot commands at power-up. This was provided because one of our ReCON systems could not handle the large current spike at power-up with all the default components powered on. Sequentially powering on the various components would reduce this power-up spike. The boot commands were added to allow a timed, sequential powering of the default channels.

The current software version, 2.2, has been optimized to fit in the provided program space of 8192 bytes. However, the program consumes 8084 bytes, which is 98.7% of the total available program space. Any future corrections, improvements, or updates to the code would need to be very efficient in order to fit in the small amount of program space remaining. One place where more program space could be freed up is in the Help command. The EEPROM command list has already been removed from the Help command. If more space is needed, the help command could be removed from the program altogether.

4.0 CONCLUSION

The System Power Controller board provides a rugged and versatile means of controlling and monitoring power in a data collection system and has proved reliable in two years of field operations.

20 Microchip PIC16F882/883/884/886/887 Data Sheet, Section 12.1.2.5, Microchip Technology Inc., 2009.