Circuit Monkey

500-0026-01

Rotary Encoder

Switch Channel™ RE11
Based Rotary Encoder
Rev. A

User Guide

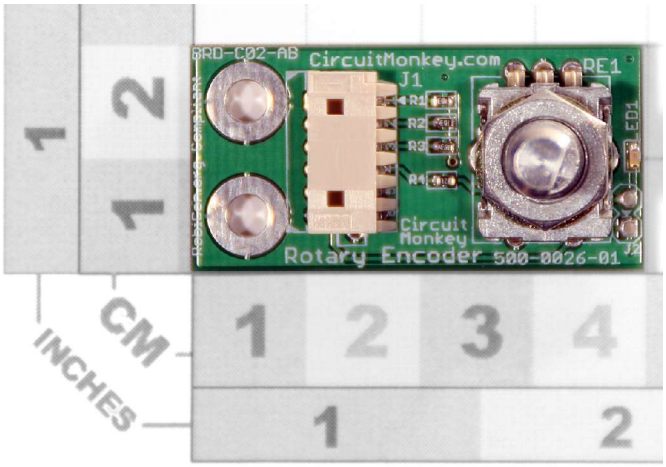
Version: 1.0
2010/01/15

Illustrations and photographs by Mark J Koch
This document was created with *OpenOffice 3*

Table of Contents

Overview

Circuit Monkey [http://circuitmonkey.com] presents part number ***500-0026-01***, a rotary encoder board based on the *Switch Channel RE11* encoder. The *RE11 Rotary Encoder* provides continuous rotation measurement for a wide variety of applications. The Input/Output signals on the rotary encoder are connected using *Molex 53015 (MicroBlade)* series for reliable and vibration resistant connections. These connectors are *RobiCon.org* compliant *(an open-source interconnect standard that we are proposing)*. The board is 40mm x 20mm and features two mounting holes placed on a 10mm apart at one end of the board. The holes are plated and connected to logic ground and accept screws of size M3 or less *(M3, 6-32, M2.5, 4-40, etc.)*.

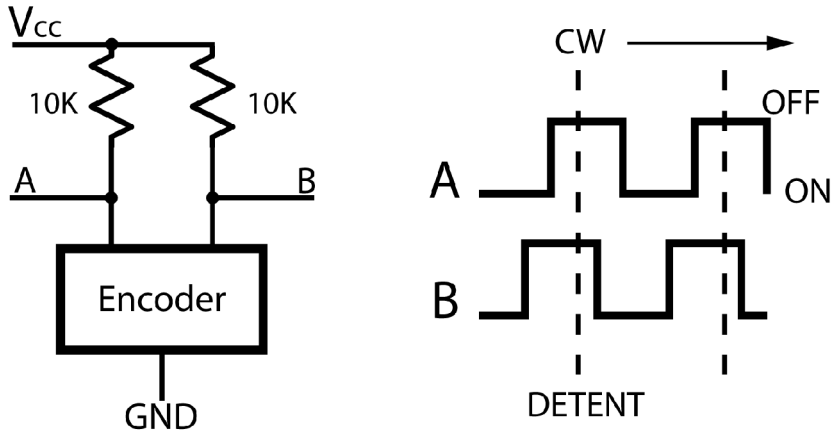


Features

- Power supply voltage; VCC = 3.3-5V.
- Operating Current: Icc=1mA(avg) . Up to 20mA with LED operating.
- RobiCon.org compliant (*Molex MicroBlade*) connector, 6-pin.
- RobiCon.org compliant board outline and mounting.
- 40mm x 20mm (WxD)
- 1.57"W x 0.79"D
- Height varies with encoder part used.

Block Diagram

A block diagram and timing diagram representing the rotary encoder board is shown below.



Note: CW = Clock-Wise

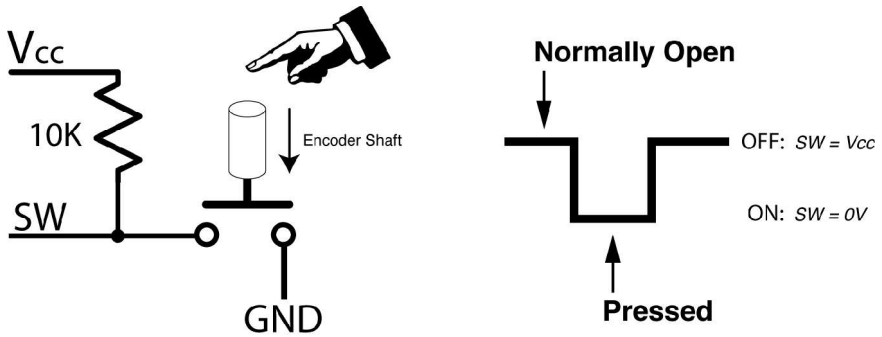
Rotary Encoder

The rotary encoder consists of two internal switches that actuate 90 degrees out of phase from each other. The switch state is normally open for both signals when the shaft is at a normal 'detent' position. The detent positions can be felt as the user turns the rotary shaft. During a turn event, the A and B signals will briefly close to Ground or 0V. Either A or B will close first depending on the direction the shaft is turned. When turning clock-wise, the B signal will ground first. The A and B signals are held to logic 'high' by 10K ohm pull-up resistors on the board. A and B will measure logic high when the rotary is static.

In software, a simple method of getting the state of the rotary is by **triggering (using an interrupt) on the falling edge of signal A** and **then measuring the state of signal B**. Signal B will be either 1 or 0 depending on which direction the shaft was turned.

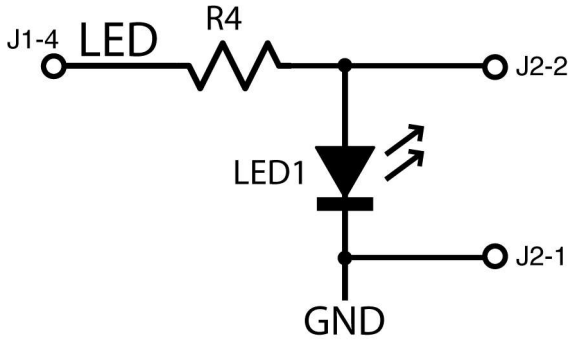
Integral Push Switch

The shaft of the rotary encoder is internally connected to a momentary push switch. Like the rotary A/B switches, the push switch is normally open and will ground when the user pushes the shaft (or attached knob). The switch signal is pulled to logic 'high' by a 10K ohm resistor on the board.



LED Indicator

The on-board LED indicator is an independent component on the board and is not logically connected to the rotary. The LED is actuated by an arbitrary I/O pin from the user's hardware. An external LED is also supported. It's features are described in the following section, *Connectors*.

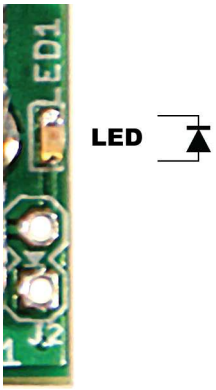


On-board LED

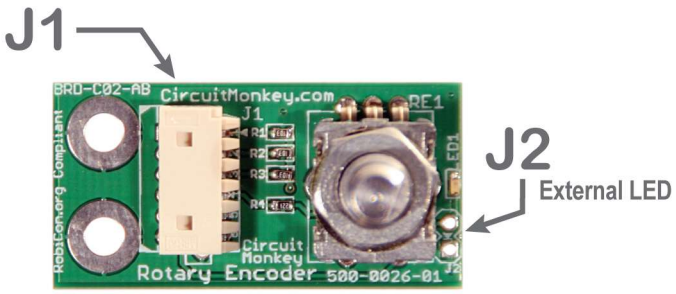
The Rotary Encoder board features a LED placed at the edge of the board and next to the J2 connector.

This LED is connected to J1 pin 4 and can be programmed to turn on or off as needed. For most applications it is useful to program it as a value, state or activity indicator.

The brightness of the LED is limited by resistor R4. The standard board ships with a 220 ohm resistor. We can customize this value to meet any of your design needs.



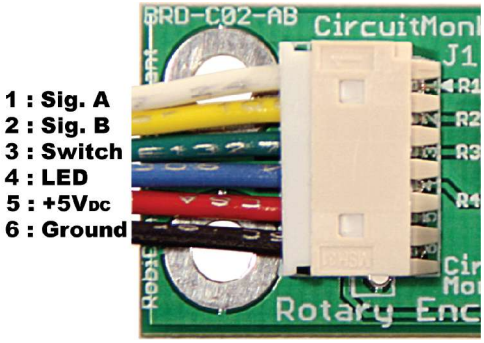
Connectors



J1 – Main Connector

This connector features six contacts, Signal A. Signal B, Switch. LED, +5V and Ground. For most applications, provide power to the board by hooking +5VDC to pin 5 and Ground on Pin 6. Connect pin 1 and 2 to your host processor. Connection of pin 3 or 4, are optional.

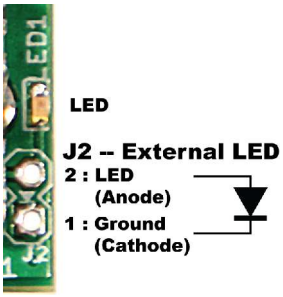
Pin	Description
1	Signal A
2	Signal B
3	Switch
4	LED
5	Vcc (+5VDC)
6	Ground



J2 – External LED

This connector features two contacts, LED and *Ground*. The LED pin (pin 2) is connected through a 220 ohm resistor from J1 pin 4. The J2 connector pads are spaced 0.1” apart and are suitable for 0.1” header type pins and also match the pin spacing on many through-hole LEDs. An external LED can be connected directly to this pin although it is recommended that the user first remove the onboard LED when doing so. This pin is activated by the source device, usually an output pin on your microcontroller.

Pin	Description
1	**LED (Anode)**
2	**Ground (Cathode)**



Applications

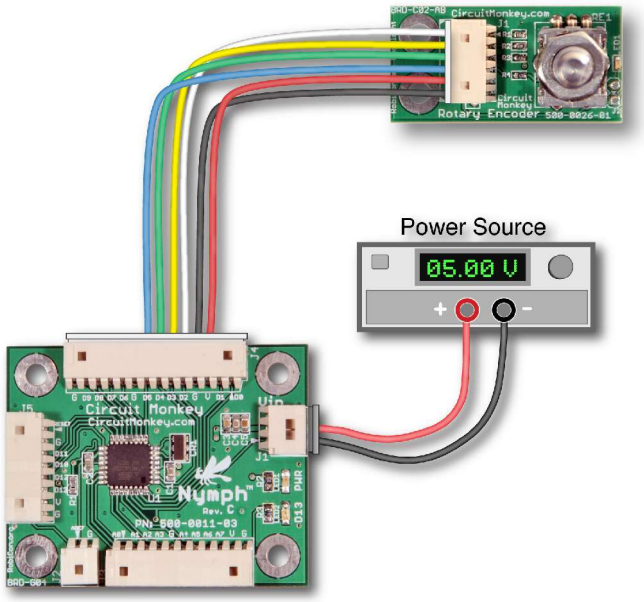
Rotary Encoder Test

This test is designed to do the following:

- Read the encoder
- Clockwise brightens the LED. Counter Clockwise dims the LED.
- Turn the Nymph/Arduino onboard LED (D13) on momentarily when the rotary switch pressed.

Hookup

Our actual test hook-up is shown below. Test code follows the hookup diagram.



Rotary Encoder J1 Pin	Description	Nymph Controller J4 Pin	Atmega328P Signal	Arduino Signal
1	**Rotary Signal A**	*5*	*PD2*	*D2*
2	**Rotary Signal B**	*6*	*PD3*	*D3*
3	**Rotary Switch**	*7*	*PD4*	*D4*
4	**LED**	*8*	*PD5*	*D5*
5	**+5V (Vcc)**	*3*		
6	**Ground**	*4*		

Rotary Encoder Test Code (for Atmega328P – Nymph Compatible)

Using the preceding hook-up diagram, one can test the functionality of the rotary encoder board with the following code. It was designed to run on a *ATmega168/328* board. We tested with our own *Nymph* product but this code should also run on an *Arduino/Freeduino* or similar board, as they are also *Atmel AVR* based.

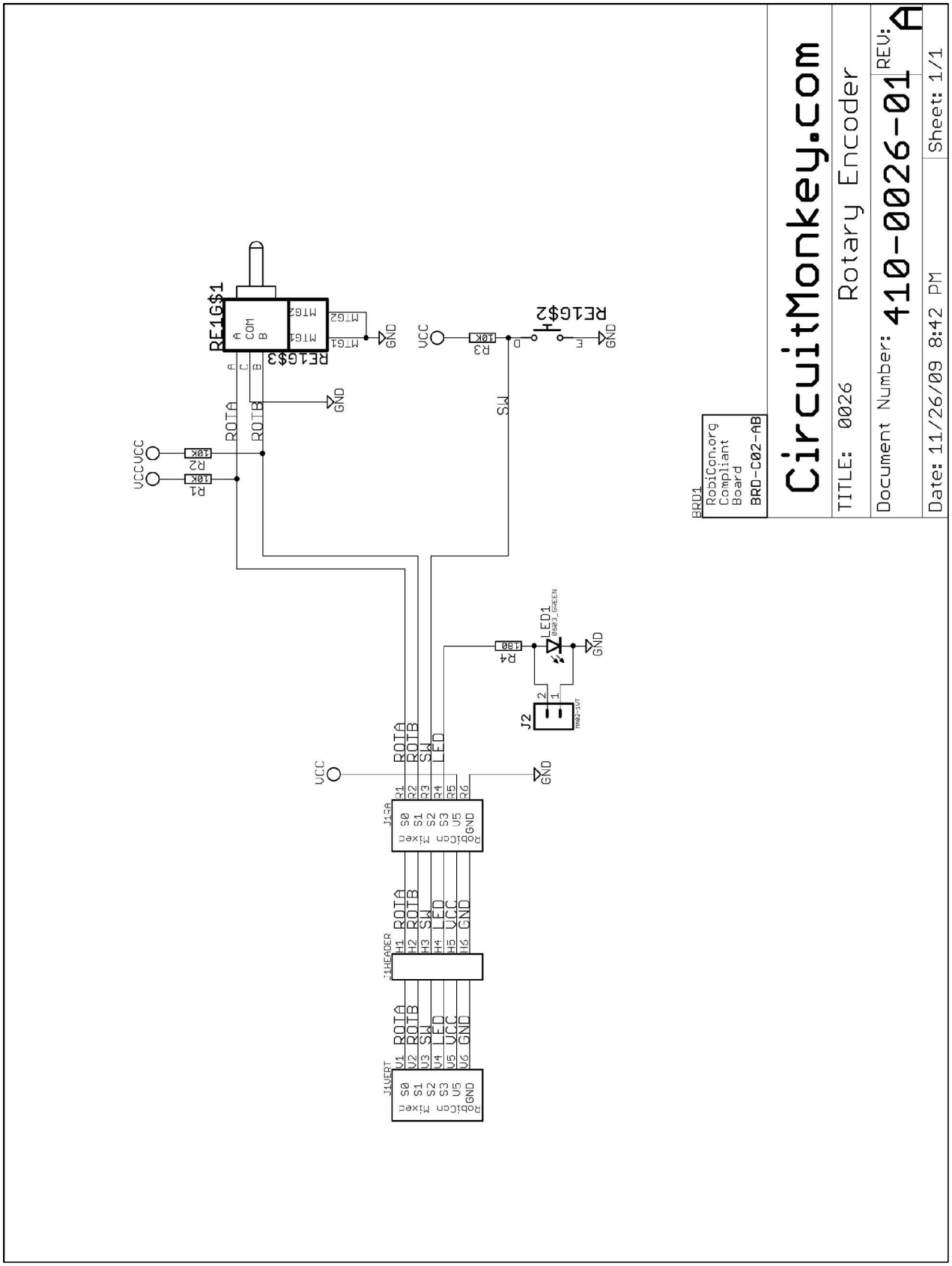
```
/**
 **
  Circuit Monkey
  500-0026-01  --  Rotary Encoder Board Test

  Designed for Nymph (ATmega328) Microcontroller

  Author:  Mark J Koch

  Description:  The 500-0026-01 Rotary Encoder board features a digital
  rotary encoder and LED on a RobiCon.org compliant printer circuit board
  (PCB).  The board features a 6-pin RobiCon compliant connector which
  supplies Ground, Power (+5VDC), digital LED signal and analog potentiometer
  value.

  500-0026-01 Pin-out:
      1 - Rotation Signal A
      2 - Rotation Signal B
      3 - Switch
      4 - LED
      5 - Vcc (+5VDC)
      6 - GND

  Nymph Hookup:

      J4-3  Vcc
      J4-4  GND
      J4-5  Rotation Signal A  -- PD2
      J4-6  Rotation Signal B  -- PD3
      J4-7  Switch  -- PD4
      J4-8  LED -- PD5

  How this code works:
      1. Initialize Output LEDs.
      2. Initialize Inputs ( Sig A, Sig B and Switch )
      3. Initialize Pin Interrupts (listens to Sig A and Switch)
      4. Main loop runs poor-man's PWM for LED brighness based on
         'rotv' value (ROT_MIN to ROT_MAX).
      5. A change in Sig A or SW calls the SIGNAL() routine.
         a. Wait 5mS for "chatter" or debounce.
         b. SIGNAL checks if switch was pressed. Processes state and
            returns if so.  LED on D13 will light when SW is pressed.
         c. If SW was not pressed, we look for a falling edge of Sig A.
         d. If Sig B is high then the knob was turned Counter Clock-Wise.
            (CCW) else it was turned Clock-Wise.

================================================================================
License:   BSD

Copyright (c) 2009, Circuit Monkey
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

    * Redistributions of source code must retain the above copyright notice,
```

```
      this list of conditions and the following disclaimer.
    * Redistributions  in binary form   must reproduce the  above copyright
      notice,  this list of conditions and  the following disclaimer in the
      documentation and/or other materials provided with the distribution.
    * Neither the name of Circuit Monkey  nor the names of its contributors
      may be used to endorse or promote products derived from this software
      without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,  BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY   DIRECT,  INDIRECT,  INCIDENTAL,  SPECIAL,  EXEMPLARY,  OR
CONSEQUENTIAL  DAMAGES  (INCLUDING,  BUT  NOT  LIMITED TO,  PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES;  LOSS OF USE, DATA, OR PROFITS;  OR BUSINESS
INTERRUPTION)  HOWEVER CAUSED  AND ON ANY THEORY OF LIABILITY,  WHETHER IN
CONTRACT,  STRICT  LIABILITY, OR TORT  (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
================================================================================
 **/
#include <avr/io.h>
#include <avr/interrupt.h>
#include <stdio.h>
#include <util/delay.h>

#define F_CPU 16000000UL  // 16MHz

// Misc Defines
#define OFF     0
#define ON      1

#define ROT_MIN 0
#define ROT_MAX 64

volatile int rotv;

//  Facade for _delay_ms
//  Helps reduce binary size since _delay_ms needs a float value
//  and we want to use an int value.  Calling the float version
//  all over your code would bloat your hex binary.
void delayMs(unsigned int ms)
{
    while(ms){
        _delay_ms(0.96);
        ms--;
    }
}

//  Facade for _delay_us
//  Helps reduce binary size since _delay_us needs a float value
//  and we want to use an int value.  Calling the float version
//  all over your code would bloat your hex binary.
void delayUs(unsigned int us)
{
    while(us){
        _delay_us(0.96);
        us--;
    }
}

void initRotaryEncoder() {
    DDRD &= ~(1<<DDD2); // PD2 input -- Rot A
    DDRD &= ~(1<<DDD3); // PD3 input -- Rot B
```

```
    DDRD &= ~(1<<DDD4); // PD4 input -- switch
}

void initLED() {
    DDRD |= (1<<DDD5); // PD5  output
}

void initD13LED() {
    DDRB |= (1<<DDB5); // PB5  output
    setD13LED(OFF);
}

void rot(int val) {
    rotv+=val;
    if ( rotv > ROT_MAX ) rotv = ROT_MAX;
    if ( rotv < ROT_MIN ) rotv = ROT_MIN;
}

void setLED(int b) {
    if ( b )  {
        PORTD |= (1<<DDD5);   // Set the bit
    } else {
        PORTD &= ~(1<<DDD5);   // UnSet the bit
    }
}

void setD13LED(int b) {
    if ( b )  {
        PORTB |= (1<<DDB5);   // Set the bit
    } else {
        PORTB &= ~(1<<DDB5);   // UnSet the bit
    }
}

void initInterrupt() {
    PCICR |= (1<<PCIE2);       // Use Pin Interrupts. Monitor Port D.

    PCMSK2 |= ( 1<<PCINT18 );  // Monitor PCINT18 == PD2 == Rot A
    PCMSK2 |= ( 1<<PCINT20 );  // Monitor PCINT20 == PD4 == Switch

    sei();    // Turn on interrupts
}

SIGNAL(PCINT2_vect) {
    delayMs(5);            // Wait for chatter.  Datasheet says 2-5mS.
    if ( !(PIND & (1<<DDD4)) ) {            // Was the switch pressed?
        setD13LED(ON);
        return;  // It was the switch that was pressed. Return when done.
    } else {
        setD13LED(OFF);
    }

    if ( !(PIND & (1<<DDD2)) ) {           // Check for falling edge of Sig A.
        if ( PIND & (1<<DDD3) )  rot(-1);  // Check if Sig B is high or low.
        else rot(1);
    }

}

int main(void)
{
    rotv = ROT_MIN;
    initLED();
    initD13LED();
```

```
    initRotaryEncoder();
    initInterrupt();

    while (1) {
        setLED(ON);
        delayUs(100*rotv);
        setLED(OFF);
        delayUs(100*(ROT_MAX-rotv));
    }
    // Never gets here.
    return(0);
}
```

Schematic


CircuitMonkey.com
TITLE: 0026 Rotary Encoder
Document Number: 410-0026-01
REV: A
Date: 11/26/09 8:42 PM
Sheet: 1/1
BRD1
RobiCon.org Compliant Board
BRD-C02-AB
RE1G$1
RE1G$2
RE1G$3
R1
R2
R3
R4
10K
180
LED1
J2
J1HEADER
J1VERT
J13A
VCC
GND
ROTA
ROTB
SW
LED

Parts List

J1 Stuffing Options

Part	Description
Molex 53014-0610	Vertical *RobiCon,* 6-pin
Molex 53015-0610	Right-Angle *RobiCon,* 6-pin
Molex 22-28-4065	Straight Pin Header, 6-pin

J2 Stuffing Options

Part	Description
Molex 22-28-4025	Straight Pin Header, 2-pin
Kingbright WP63ID (or similar)	LED, Through Hole type. The on-board 220ohm dropping resistor is calculated to run the LED at ~15mA. If you need more brightness consider an alternate resistor value for your application. We'll be happy to stuff any other resistor value you want at no charge (if we have it in stock).

RE1 Stuffing Options

Part	Description
Switch Channel RE11 Series *SwitchChannel.com*	*Switch Channel* RE11 Series with at least these attributes: `RE11**-V1*12-****S` *V1 = Vertical Only, S = 10.6mm footing*
BI Technologies EN11 Series *BITechnologies.com*	BI Technologies EN11 Series with at least these attributes: `EN11-H** * * * **` *H = Top Adjust*

SMT Resistors

These resistors are 0603 (0.06” x 0.03”) type surface mount. You will need a fine point soldering iron and some skill to change these parts. However, there are many tutorials on the web describing techniques for unsoldering and soldering these small components. *Circuit Monkey* is happy to change the values to your specifications when you order. If you already have a board, we'll do it for free if you pay the return postage (usually first class in the USA). Contact us first to arrange this.

Appendix A: References and Links

Rotary Encoder (RE11)

Switch Channel (SwitchChannel.com)

Part Series: RE-11 Rotary Encoder

BI Technologies (BI Technologies)

Part Series: EN11-H**

Links

Circuit Monkey

http://www.circuitmonkey.com

RobiCon.org

A proposed open standard for Robotics Interconnect. Currently lead by the owner of Circuit Monkey.

http://www.robicon.org

Atmel

Manufacturers of the *AVR/ATmega* microcontroller chips.

http://atmel.com/products/AVR/

StackOverlfow.com

Stack Overflow -- Rotary Encoder Discussion

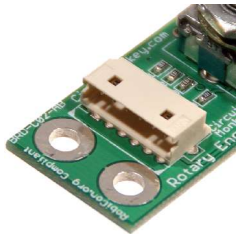
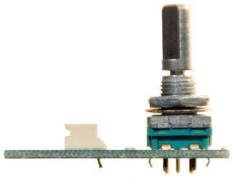
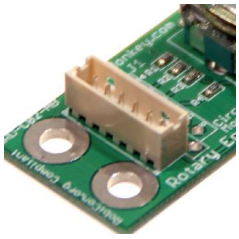
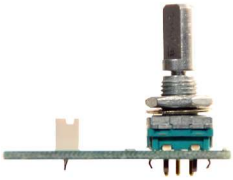
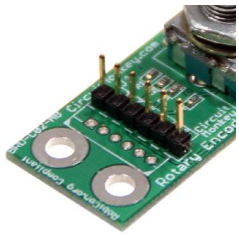
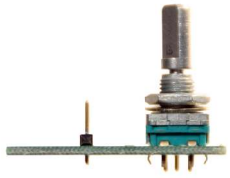
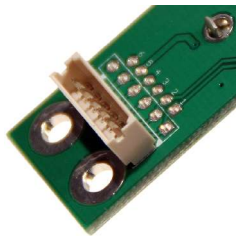
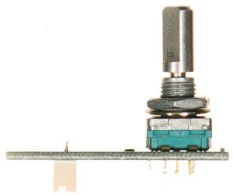
AVRFreaks.net

AVR Freaks -- Rotary Encoder Discussion

Appendix B: Alternate Configuration

J1 Connector Types and Orientation

The PCB footprint for J1 features a combination of three connector row holes for various connector options show in this table below. Please note the position of pin 1 (the square-shaped pads) regardless of configuration.

1. Right-Angle RobiCon – Top Mount – 2mm		
2. Vertical RobiCon – Top Mount - 2mm		
3. Pin Header – Top Mount – 2.54mm (0.1”) Right-angle pin header also possible.		
4. Vertical RobiCon – Bottom Mount – 2mm		
5. Pin Header – Bottom Mount – 2.54mm (0.1”)