

This is the Revision A version of the [Birod5 RoboBrick](#). The status of this project is [work in progress](#).

BIROD5 Robobrick (Revision A)

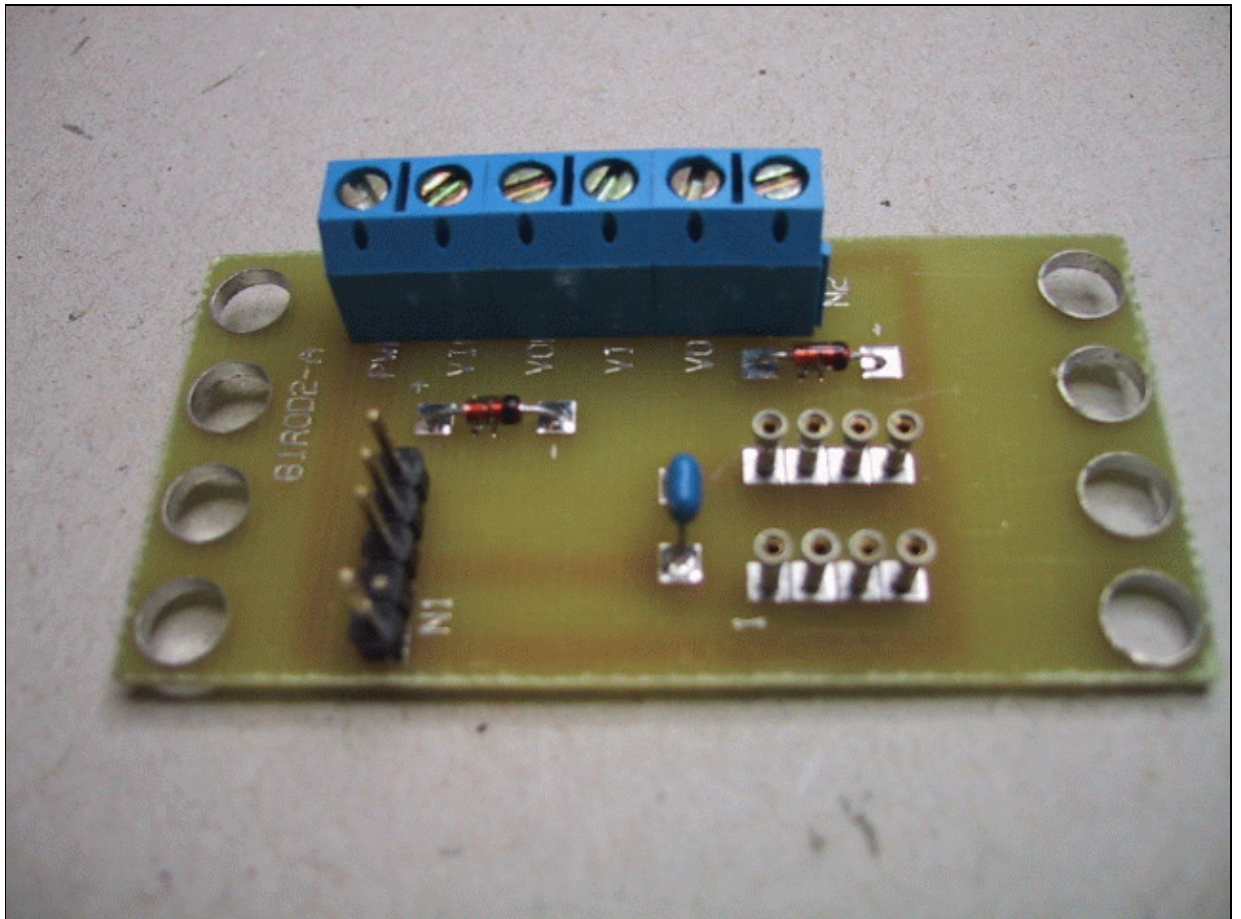
Table of Contents

This document is also available as a [PDF](#) document.

- [1. Introduction](#)
- [2. Programming](#)
- [3. Hardware](#)
 - ◆ [3.1 Circuit Schematic](#)
 - ◆ [3.2 Printed Circuit Board](#)
- [4. Software](#)
- [5. Issues](#)

1. Introduction

The BIROD5 RoboBrick is used to connect to up to 2 of the [Sharp](#)® GP2D05 IROD (InfraRed Optical Distance) measuring sensors. This version of the Sharp chip provides a single bit of information for when the sensor is within a fixed distance an object.



2. Programming

The basic operation is to send a query to the BIROD5 RoboBrick to read the 2 bits of data. The programmer can download a complement mask to cause any of the bits to be complemented prior to reading.

There is an enable mask that allows the programmer to individually enable and disable one or the other IROD. In addition, there is an alternate bit which, when set, causes the IROD's to be alternately strobed. The alternate bit would be set if you are concerned that one sensor might interfere with another.

The BIROD5 RoboBrick supports [RoboBrick Interrupt Protocol](#). The interrupt pending bit is set whenever the formula:

$$L \& (\sim I) \mid H \& I \mid R \& (\sim P) \& I \mid F \& P \& (\sim I)$$

is non-zero, where:

- I is the current input bits XOR'ed with the complement mask (C)
- P is the previous value of I
- L is the low mask
- H is the high mask
- R is the raising mask
- F is the falling mask

and

- ~ is bit-wise complement
- | is bit-wise OR
- & is bit-wise AND

Once the interrupt pending bit is set, it must be explicitly cleared by the user.

The BIROD5 RoboBrick supports both the standard [shared commands](#) and the [shared interrupt commands](#) in addition to the following commands:

Command	Send/ Receive	Byte Value								Discussion
		7	6	5	4	3	2	1	0	
Read Inputs	Send	0	0	0	0	0	0	0	0	Return input values <i>ab</i> (after XOR'ing with complement mask)
	Receive	0	0	0	0	0	0	<i>a</i>	<i>b</i>	
Read Raw	Send	0	0	0	0	0	0	0	1	Return raw data <i>ab</i> (without XOR'ing with complement mask)
	Receive	0	0	0	0	0	0	<i>a</i>	<i>b</i>	
Read Alternate Bit	Send	0	0	0	0	0	0	1	0	Return alternate bit <i>a</i>
	Receive	0	0	0	0	0	0	0	<i>a</i>	
Read Enable Complement Mask	Send	0	0	0	0	0	0	1	1	Return enable mask <i>ee</i> and complement mask <i>cc</i>
	Receive	0	0	0	0	<i>e</i>	<i>e</i>	<i>c</i>	<i>c</i>	
Read High Low Mask	Send	0	0	0	0	0	1	0	0	Return high mask <i>hh</i> and low mask <i>ll</i>
	Receive	0	0	0	0	<i>h</i>	<i>h</i>	<i>l</i>	<i>l</i>	

BIROD5 RoboBrick (Revision A)

Read Raising Falling Mask	Send	0	0	0	0	0	1	0	1	Return raising mask <i>rr</i> and falling mask <i>ff</i>
	Receive	0	0	0	0	<i>r</i>	<i>r</i>	<i>f</i>	<i>f</i>	
Set Alternate Bit	Send	0	0	0	0	0	1	1	<i>a</i>	Set the alternate bit to <i>a</i> .
Set Enable Complement Mask	Send	0	0	0	1	<i>e</i>	<i>e</i>	<i>c</i>	<i>c</i>	Set enable mask to <i>ee</i> and complement mask to <i>cc</i> .
Set High Low Mask	Send	0	0	1	0	<i>h</i>	<i>h</i>	<i>l</i>	<i>l</i>	Set high mask to <i>hh</i> and low mask to <i>ll</i> .
Set Raising Falling Mask	Send	0	0	1	1	<i>r</i>	<i>r</i>	<i>f</i>	<i>f</i>	Set raising mask to <i>rr</i> and falling mask to <i>ff</i> .
Read Interrupt Bits	Send	1	1	1	0	1	1	1	1	Return the interrupt pending bit <i>p</i> and the interrupt enable bit <i>e</i> .
	Receive	0	0	0	0	0	0	<i>e</i>	<i>p</i>	
Set Interrupt Bit Commands	Send	1	1	1	1	0	<i>c</i>	<i>c</i>	<i>c</i>	Execute shared set interrupt command <i>ccc</i> .
Shared Commands	Send	1	1	1	1	1	<i>c</i>	<i>c</i>	<i>c</i>	Execute shared command <i>ccc</i> .

3. Hardware

The hardware consists of a circuit schematic and a printed circuit board.

3.1 Circuit Schematic

The schematic for the Birod5 RoboBrick is shown below:

[birod5.gbl](#)

The RS-274X "Gerber" back (solder side) layer.

[birod5.gtl](#)

The RS-274X "Gerber" top (component side) layer.

[birod5.gal](#)

The RS-274X "Gerber" artwork layer.

[birod5.drl](#)

The "Excellon" NC drill file.

[birod5.tol](#)

The "Excellon" NC drill rack file.

4. Software

The BIROD5 software is available as one of:

[birod5.ucl](#)

The μ CL source file.

[birod5.asm](#)

The resulting human readable PIC assembly file.

[birod5.lst](#)

The resulting human readable PIC listing file.

[birod5.hex](#)

The resulting Intel[®] Hex file that can be fed into a PIC12C5xx programmer.

The BIROD5 test software is available as one of:

[birod5_test.ucl](#)

The μ CL source file.

[birod5_test.asm](#)

The resulting human readable PIC assembly file.

[birod5_test.lst](#)

The resulting human readable PIC listing file.

[birod5_test.hex](#)

The resulting Intel[®] Hex file that can be fed into a PIC12C5xx programmer.

5. Issues

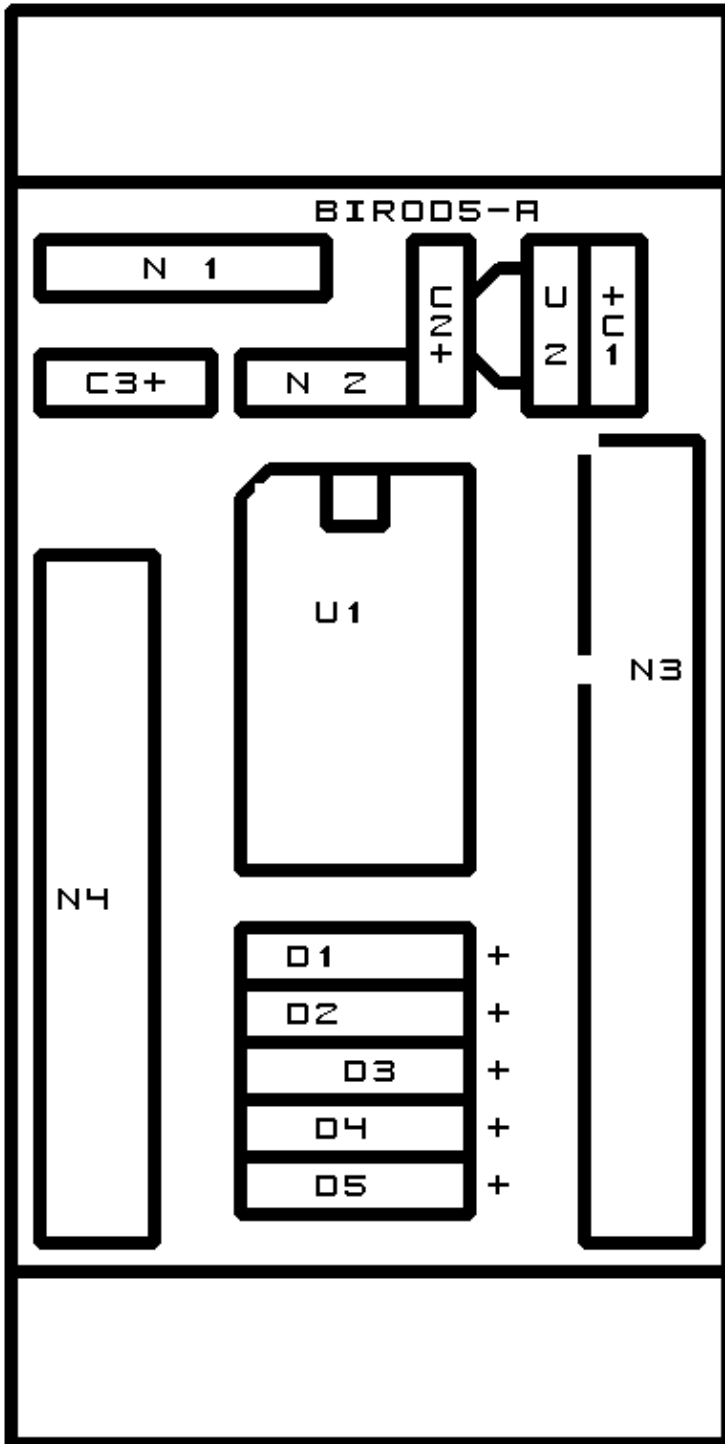
Any fabrication issues will be listed here.

[Copyright](#) (c) 2000–2002 by [Wayne C. Gramlich](#). All rights reserved.

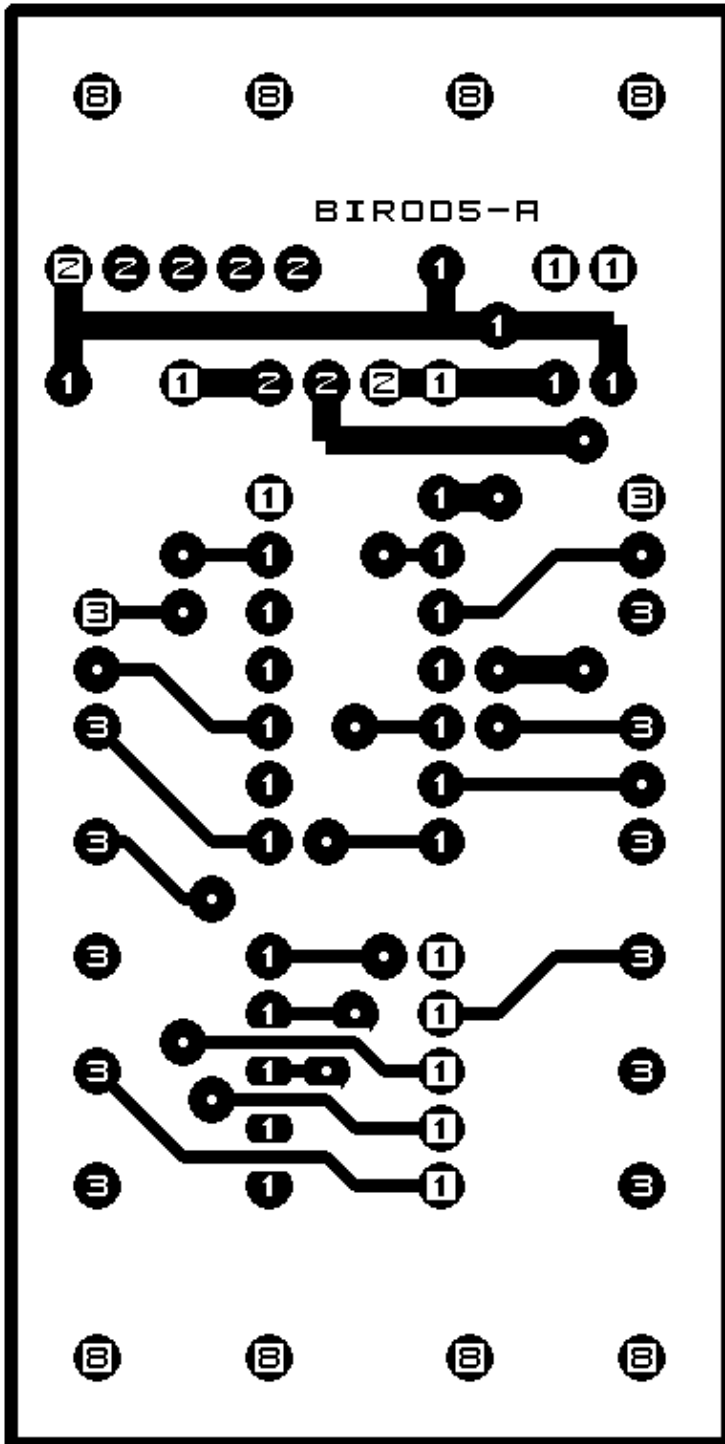
A. Appendix A: Parts List

```
# Parts list for BIROD5 RoboBrick (Rev. A)
#
C1: Capacitor470nF - .47 uFF Tantalum Capacitor [Jameco: 33531]
C2: Capacitor22uF - 22 uF Tantalum Capacitor [Jameco: 94094]
C3: Capacitor10pF - 10 pF Ceramic Capacitor [Jameco: 15333]
D1-5: 1N4148 - Small Signal Diode [Jameco: 36038]
N1: Header1x5.RBSlave - 1x5 Male Header [5/40 Jameco: 160881]
N2: Header1x3.BIROD5 - 1 x 3 Male Header [3/40 Jameco: 160881]
N3: TerminalStrip7.BIROD5 - 7 Junction Terminal Strip [2 Jameco: 189667]
N4: TerminalStrip6.BIROD5 - 6 Junction Terminal Strip [2 Jameco: 189667]
U1: PIC16C505.BIROD5 - Microchip PIC16C505 [Digikey: PIC16C505-04/P-ND]
U2: LM2940CG-5 - 5 Volt Low Dropout Voltage Regulator [Jameco: 107182]
```

B. Appendix B: Artwork Layer



C. Appendix C: Back (Solder Side) Layer



D. Appendix D: Front (Component Side) Layer

