

This is the Revision D version of the [AnalogIn4 RoboBrick](#). The status of this project is [work in progress](#).

AnalogIn4 Robobrick (Revision D)

Table of Contents

This document is also available in [PDF](#) format.

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

1. Introduction

The AnalogIn4 RoboBrick allows for the input of up to 4 analog voltages between 0 and 5 volts with a resolution of 8 bits.

2. Programming

The AnalogIn4 RoboBrick is continuously reading the analog inputs from its four A/D pins. The controlling program can just read the results of the digital conversion, or it can have the result down converted into a single binary bit. Each pin has a threshold high and threshold low register that is used for the down conversion. Whenever the digital conversion exceeds the high threshold register, the down conversion results in a 1. Whenever the digital conversion is lower than the low threshold register, the down conversion results in a 0. A hysteresis effect can be introduced by having some spread between the high and low threshold values.

There AnalogIn4 RoboBrick operates in either regular mode or Vref mode. In regular mode, all four inputs are A/D converted between 0 and 5 volts. In Vref mode, input 1 is used as Vref, the highest expected input voltage, and the remaining three inputs are A/D converted between 0 and Vref.

After the down conversions to binary bits, the result is 4-bits of binary data. A complement mask can be used to selectively invert individual bits in the 4-bit data.

The AnalogIn4 RoboBrick supports [RoboBrick Interrupt Protocol](#) for those lines that are being used as inputs. The interrupt pending bit is set whenever the 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

AnalogIn4 RoboBrick (Revision D)

- H is the high mask
- R is the raising mask
- F is the falling mask

and

- \sim 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.

In addition to the [common shared commands](#) and the [shared interrupt commands](#), the AnalogIn4 RoboBrick supports following commands:

Command	Send/ Receive	Byte Value								Discussion
		7	6	5	4	3	2	1	0	
Read Pin	Send	0	0	0	0	0	0	<i>b</i>	<i>b</i>	Read pin <i>bb</i> and respond with 8-bit value <i>vvvvvvvv</i>
	Receive	<i>v</i>	<i>v</i>	<i>v</i>	<i>v</i>	<i>v</i>	<i>v</i>	<i>v</i>	<i>v</i>	
Read Binary Values	Send	0	0	0	0	0	1	0	0	Return the binary values <i>abcd</i> (after XOR'ing with complement mask)
	Receive	0	0	0	0	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	
Read Raw Binary	Send	0	0	0	0	0	1	0	1	Return the raw binary values <i>abcd</i> (no XOR with complement mask)
	Receive	0	0	0	0	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	
Reset	Send	0	0	0	0	0	1	1	0	Reset everything to zero
Read Complement Mask	Send	0	0	0	0	1	0	0	0	Return and return the complement mask <i>cccc</i>
	Receive	0	0	0	0	<i>c</i>	<i>c</i>	<i>c</i>	<i>c</i>	
Read High Mask	Send	0	0	0	0	1	0	0	1	Return and return the high mask <i>hhhh</i>
	Receive	0	0	0	0	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	
Read Low Mask	Send	0	0	0	0	1	0	1	0	Return and return the high mask <i>llll</i>
	Receive	0	0	0	0	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	
Read Raising Mask	Send	0	0	0	0	1	0	1	1	Return and return the raising mask <i>rrrr</i>
	Receive	0	0	0	0	<i>r</i>	<i>r</i>	<i>r</i>	<i>r</i>	
Read Falling Mask	Send	0	0	0	0	1	1	0	0	Return and return the falling mask <i>ffff</i>
	Receive	0	0	0	0	<i>f</i>	<i>f</i>	<i>f</i>	<i>f</i>	
Read Vref Mode	Send	0	0	0	0	1	1	0	1	Read and return the Vref mode bit <i>v</i>
	Receive	0	0	0	0	0	0	0	<i>f</i>	
Set Vref Mode	Send	0	0	0	0	1	1	1	<i>v</i>	Set the Vref mode to <i>v</i> (0=regular 1=Vref Mode)
Read High Threshold	Send	0	0	0	1	0	0	<i>b</i>	<i>b</i>	Read and return high threshold for pin <i>bb</i> of <i>hhhhhhhh</i>
	Receive	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	
Read Low Threshold	Send	0	0	0	1	0	1	<i>b</i>	<i>b</i>	Read and return low threshold for pin <i>bb</i> of <i>llllllll</i>
	Receive	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	
Set High Threshold	Send	0	0	0	1	1	0	<i>b</i>	<i>b</i>	Set high threshold for pin <i>bb</i> to <i>hhhhhhhh</i>
	Send	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	

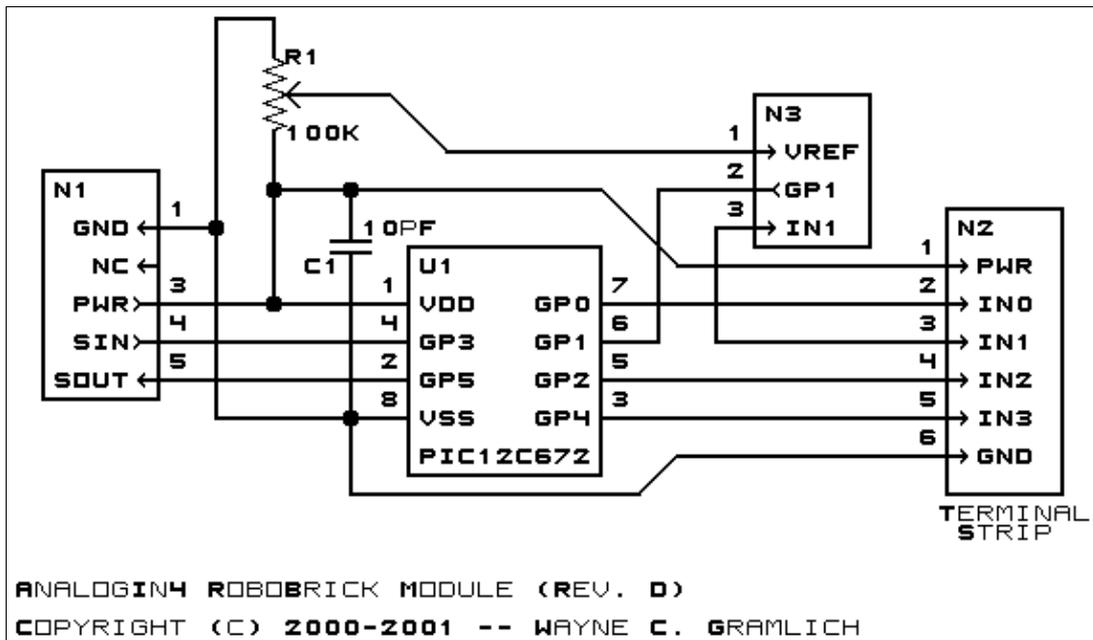
Set Low Threshold	Send	0	0	0	1	1	1	<i>b</i>	<i>b</i>	Set low threshold for pin <i>bb</i> to <i>lllllll</i>
	Send	<i>l</i>								
Set Complement Mask	Send	0	0	1	0	<i>c</i>	<i>c</i>	<i>c</i>	<i>c</i>	Set complement mask to <i>cccc</i>
Set High Mask	Send	0	1	0	0	<i>h</i>	<i>h</i>	<i>h</i>	<i>h</i>	Set high mask to <i>hhhh</i>
Set Low Mask	Send	0	1	0	1	<i>l</i>	<i>l</i>	<i>l</i>	<i>l</i>	Set low mask to <i>llll</i>
Set Raising Mask	Send	0	1	1	0	<i>r</i>	<i>r</i>	<i>r</i>	<i>r</i>	Set raising mask to <i>rrrr</i>
Set Falling Mask	Send	0	1	1	1	<i>f</i>	<i>f</i>	<i>f</i>	<i>f</i>	Set falling mask to <i>ffff</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 Commands	Send	1	1	1	1	0	<i>c</i>	<i>c</i>	<i>c</i>	Set Interrupt Command <i>ccc</i> .
Shared Commands	Send	1	1	1	1	1	<i>c</i>	<i>c</i>	<i>c</i>	Execute common 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 AnalogIn4 RoboBrick is shown below:



The parts list kept in a separate file -- [analogin4.ptl](#).

3.2 Printed Circuit Board

The printed circuit board files are listed below:

[analogin4_back.png](#)

The solder side layer.

[analogin4_front.png](#)

The component side layer.

[analogin4_artwork.png](#)

The artwork layer.

[analogin4.gbl](#)

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

[analogin4.gtl](#)

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

[analogin4.gal](#)

The RS-274X "Gerber" artwork layer.

[analogin4.drl](#)

The "Excellon" NC drill file.

[analogin4.tol](#)

The "Excellon" tool rack file.

3.3 Construction Instructions

The [construction instructions](#) are in separate file to be a little more printer friendly.

4. Software

The Analogin4 software is available as one of:

[analogin4.ucl](#)

The μ CL source file.

[analogin4.asm](#)

The resulting human readable PIC assembly file.

[analogin4.lst](#)

The resulting human readable PIC listing file.

[analogin4.hex](#)

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

The following Analogin4 test software is available:

[analogin4_test.ucl](#)

The μ CL source file.

[analogin4_test.asm](#)

The resulting human readable PIC assembly file.

[analogin4_test.lst](#)

The resulting human readable PIC listing file.

[analogin4_test.hex](#)

The resulting Intel[®] Hex file that can be fed into a PIC16F84 programmer.

5. Issues

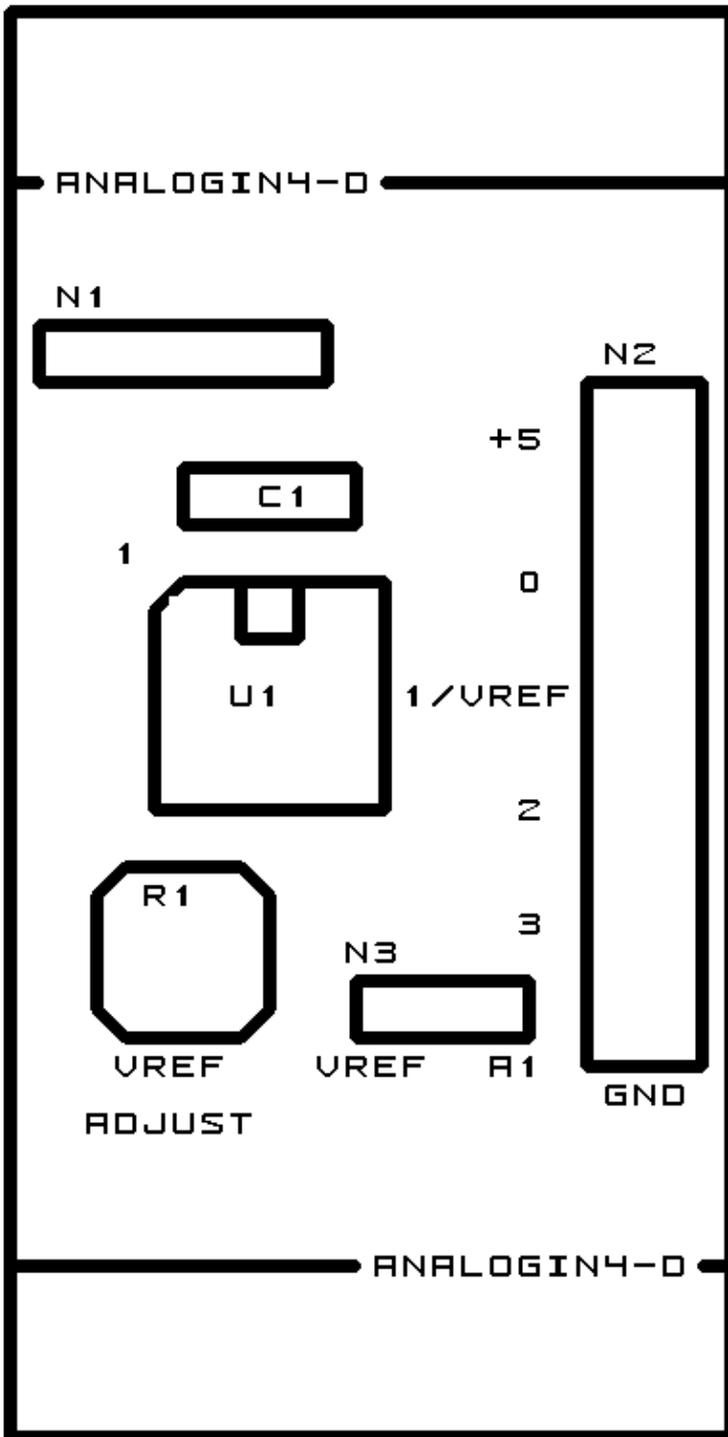
Any fabrication issues that come up will be discussed here.

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

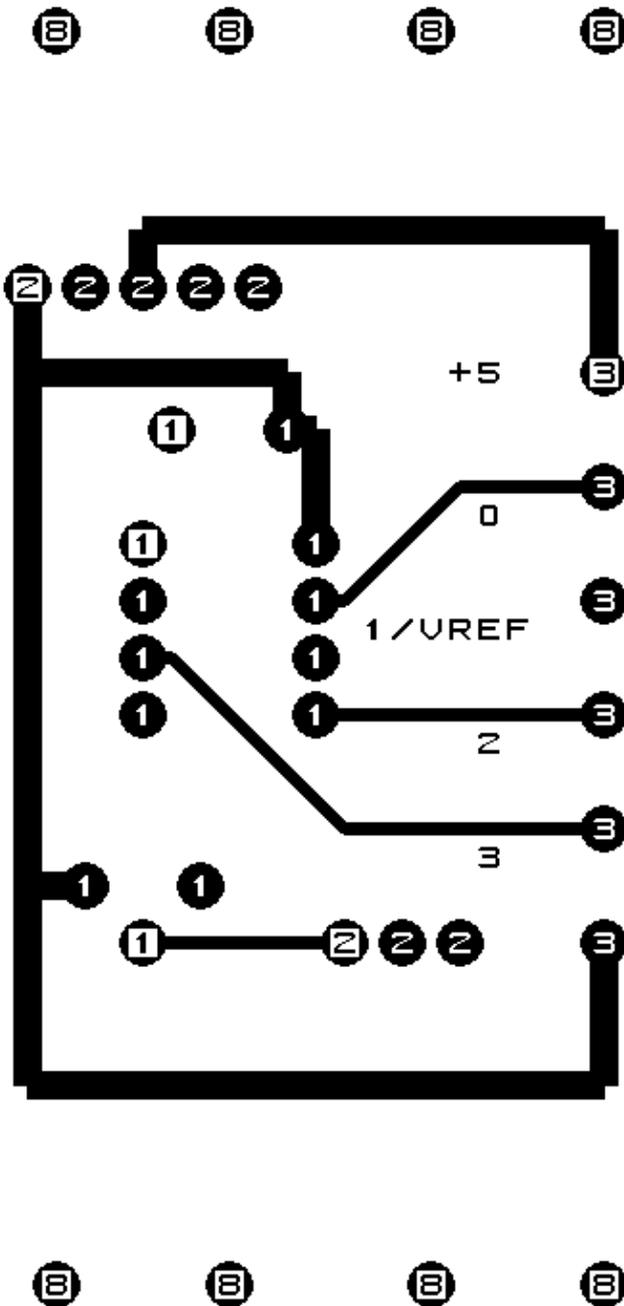
A. Appendix A: Parts List

```
# Parts list for AnalogIn4 RoboBrick (Rev. D)
#
C1: Capacitor10pF - 10 pF Ceramic Capacitor [Jameco: 15333]
N1: Header1x5.RBSlave - 1x5 Male Header [5/40 Jameco: 160881]
N2: TerminalStrip6.AnalogIn4 - 6 Junction Terminal Strip [2 Jameco: 189667]
N3: Header1x3.AnalogIn4 Header1x3 - 1x3 Male Header [3/40 Jameco: 160881]
# The shorting block goes onto N3; it is not present on the PCB.
N4: ShortingBlock - Shorting Block [Jameco: 22023]
R1: ResistorTrimPot100K.AnalogIn4 - 100K Trim Potentiometer [Jameco: 95484]
U1: PIC12C672.AnalogIn4B - Microchip PIC12C672 [Digikey: PIC12C672-04/P-ND]
# Each RoboBrick should have a connector cable:
N5-6: CableHeader1x5 - 1 x 5 Female Shell [Jameco: 163686]
N7-14: CablePinFemale - Female Pin [Jameco: 100765]
N15-16: CablePinMale - Male Pin [Jameco: 145357]
```

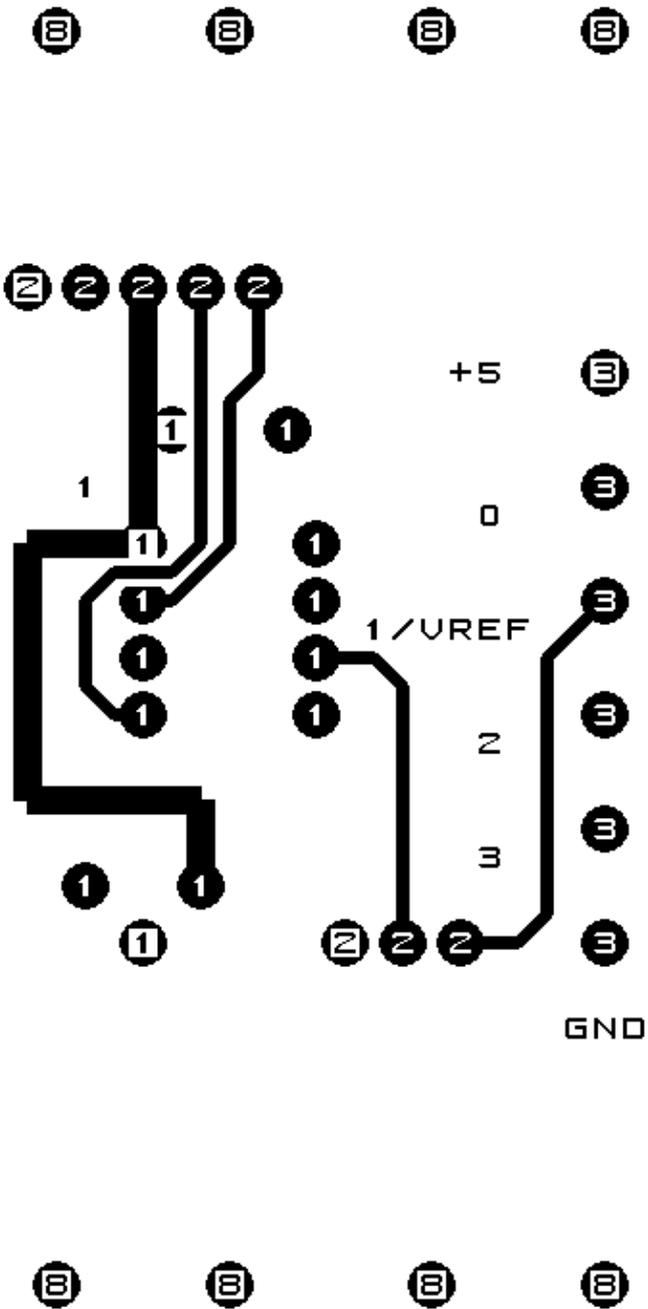
B. Appendix B: Artwork Layer



C. Appendix C: Back (Solder Side) Layer



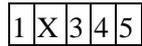
D. Appendix D: Front (Component Side) Layer



E. Appendix E: Construction Instructions

The instruction steps for building the AnalogIn4 (Rev. D) RoboBrick are listed below:

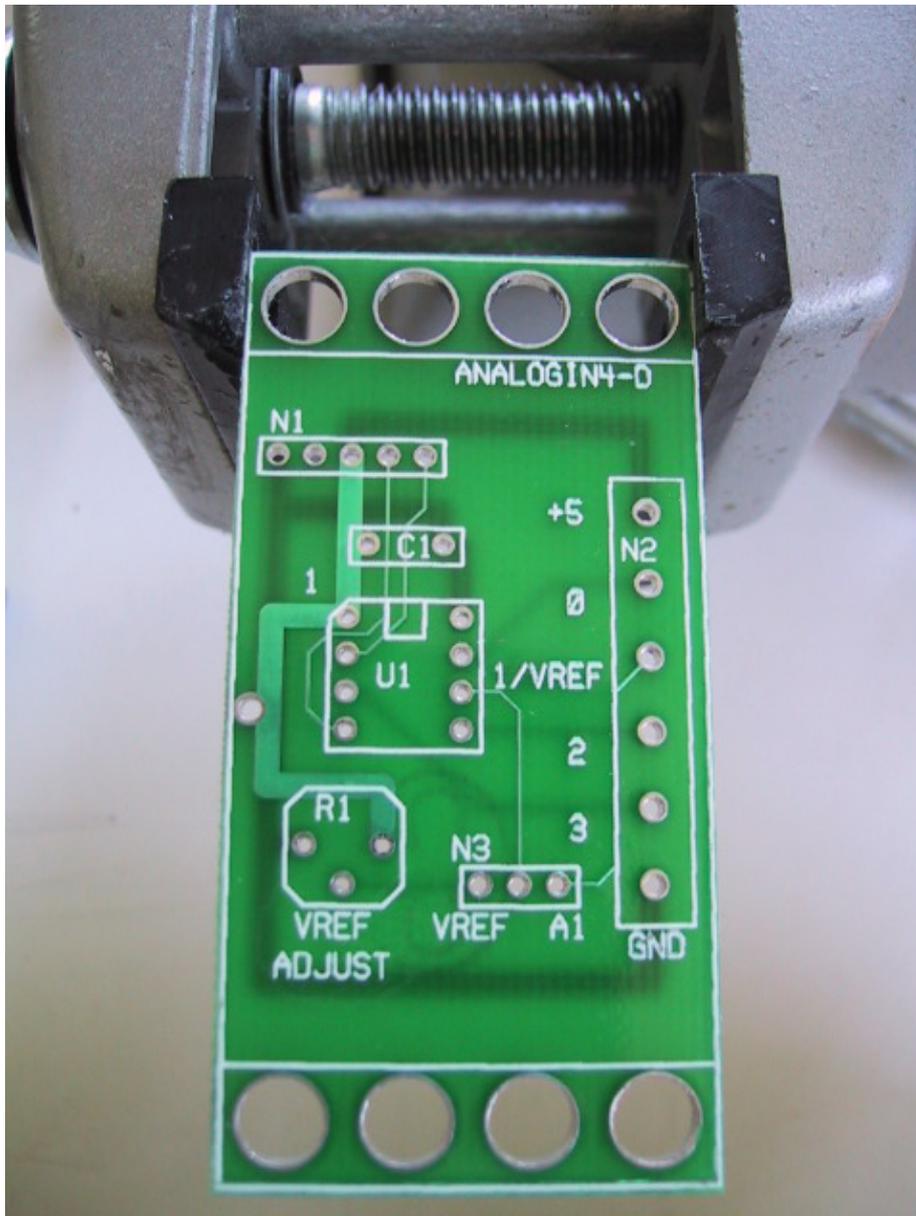
1. Orient the board so that it is vertical. By convention, north is the upper edge, south is the lower edge, west is the left edge, and east is the right edge. N1 should be in the north west corner. [\[Step 1\]](#).
2. Take a 1×5 male header and using some diagonal cutters, snip off pin 2 using the diagram below:



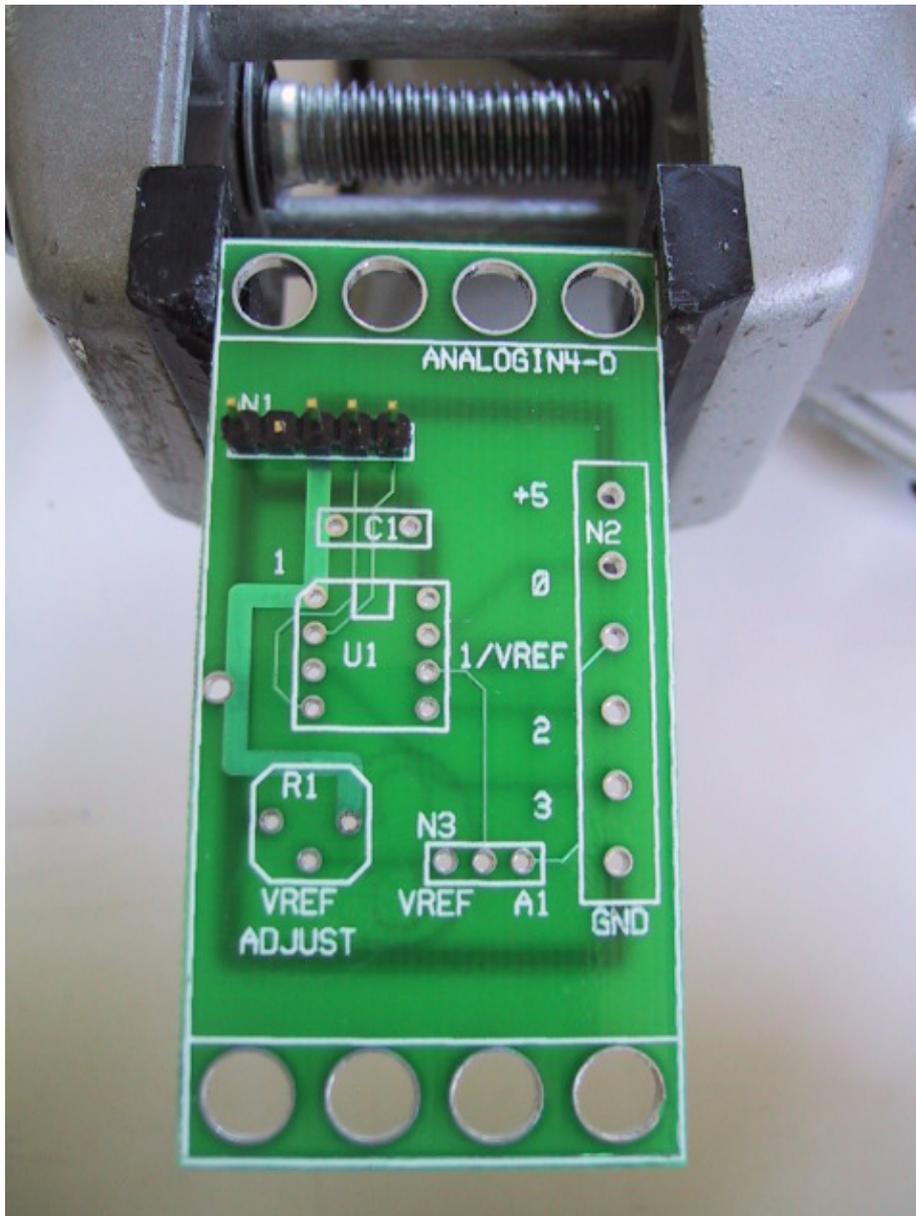
- Pin 2 is the pin marked with an `X'. Install the 1×5 male header into N1 with pin 1 to the left. First, solder only one pin and verify that the connector is pointing straight up. If not, use the soldering iron to melt solder around the pin you just solder and reposition the header so that it is pointing straight up. When you are happy with the position, solder in the remaining 4 pins. [\[Step 2\]](#).
3. Take a 1×3 male header, orient it so that it is horizontal, and insert it into N3. As in the previous instruction, solder 1 pin, verify that it is completely vertical, and solder the remaining 2 pins in. [\[Step 3\]](#).
 4. Find the 10pF capacitor and insert it into C1. Spread the leads, solder 1 pin, verify that it is positioned correctly, solder the remaining lead, and snip off the excess leads. [\[Step 4\]](#).
 5. Find the 100K trim potentiometer and insert it into R1. There is only one orientation where all three leads will fit straight in. Solder one lead, verify that it is flat on the board, solder the remaining 2 leads, and snip off the excess leads. [\[Step 5\]](#).
 6. Find the 6–terminal terminal strip and insert it into N2. If all you can find is some 2–terminal strips, a 6–terminal strip can be assembled from three 2–terminal strip by sliding them together. Make sure that the wire holes point towards the east. Solder in 1 pin, verify position, and solder the remaining pins. [\[Step 6\]](#).
 7. Install the preprogrammed PIC12C672 into U1 with the notch facing up. (We have highlighted the notch in the construction picture with some white marker.) Solder in one pin, verify that the IC is flat against the board, re–heat and reposition if necessary, and solder in the remaining pins. It is very important that the chip does not get overheated. Use a low wattage soldering iron. Also, solder in one pin at a time and then touch the chip with your finger; if it is hot to touch let the chip cool down before you solder in another pin. [\[Step 7\]](#).
 8. Install the shorting block on N3 on the two pins to the east. [\[Missing step8 picture\]](#).

The assembly of the AnalogIn4 (Rev. D) RoboBrick is complete.

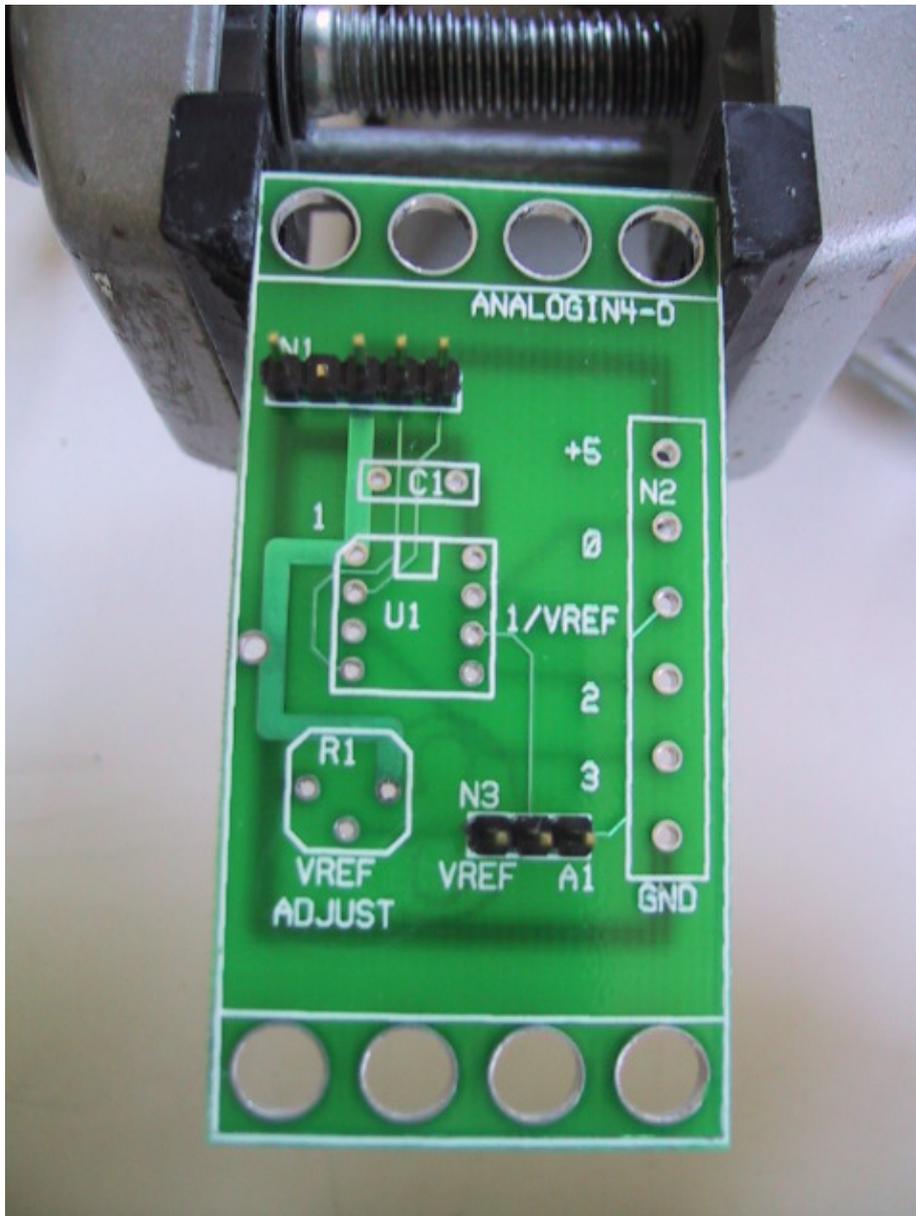
Step 1



Step 2



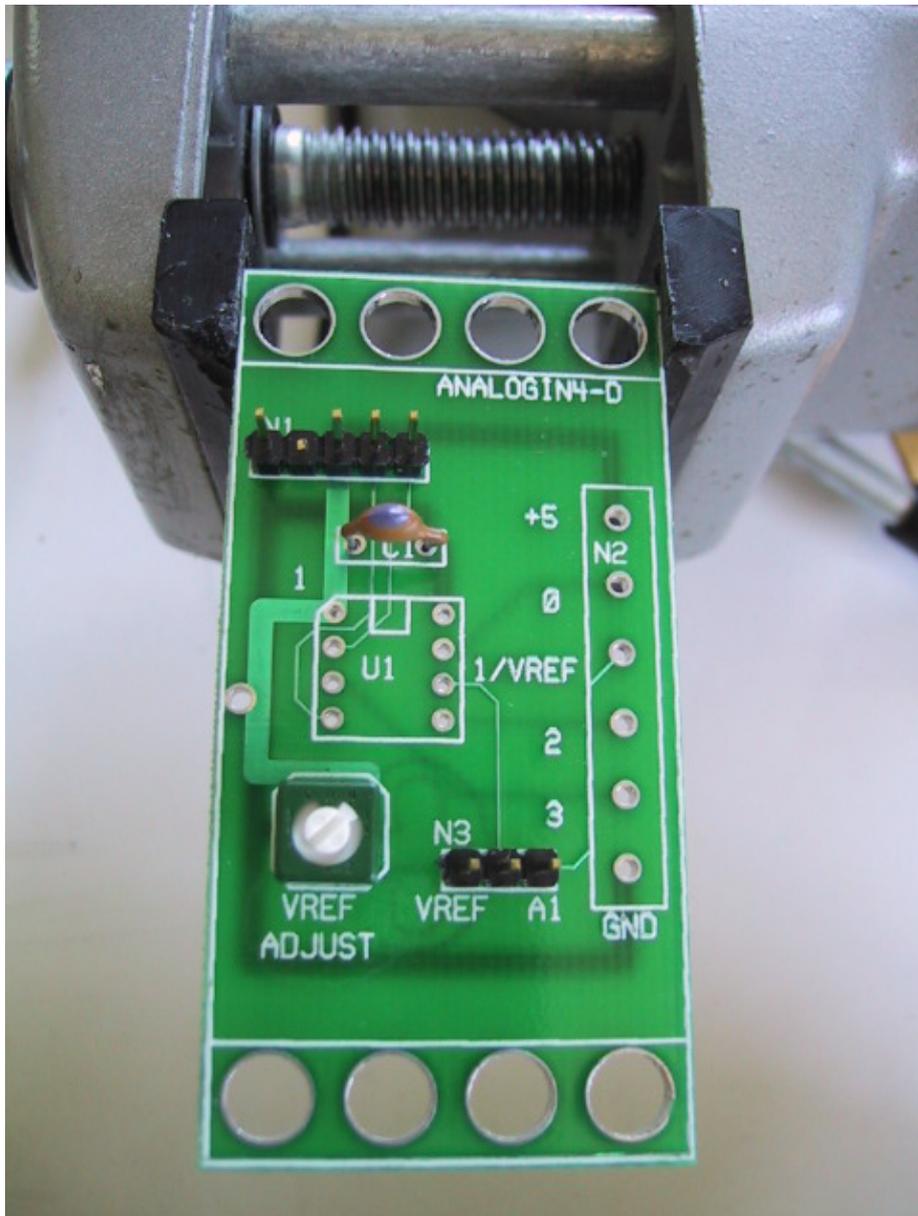
Step 3



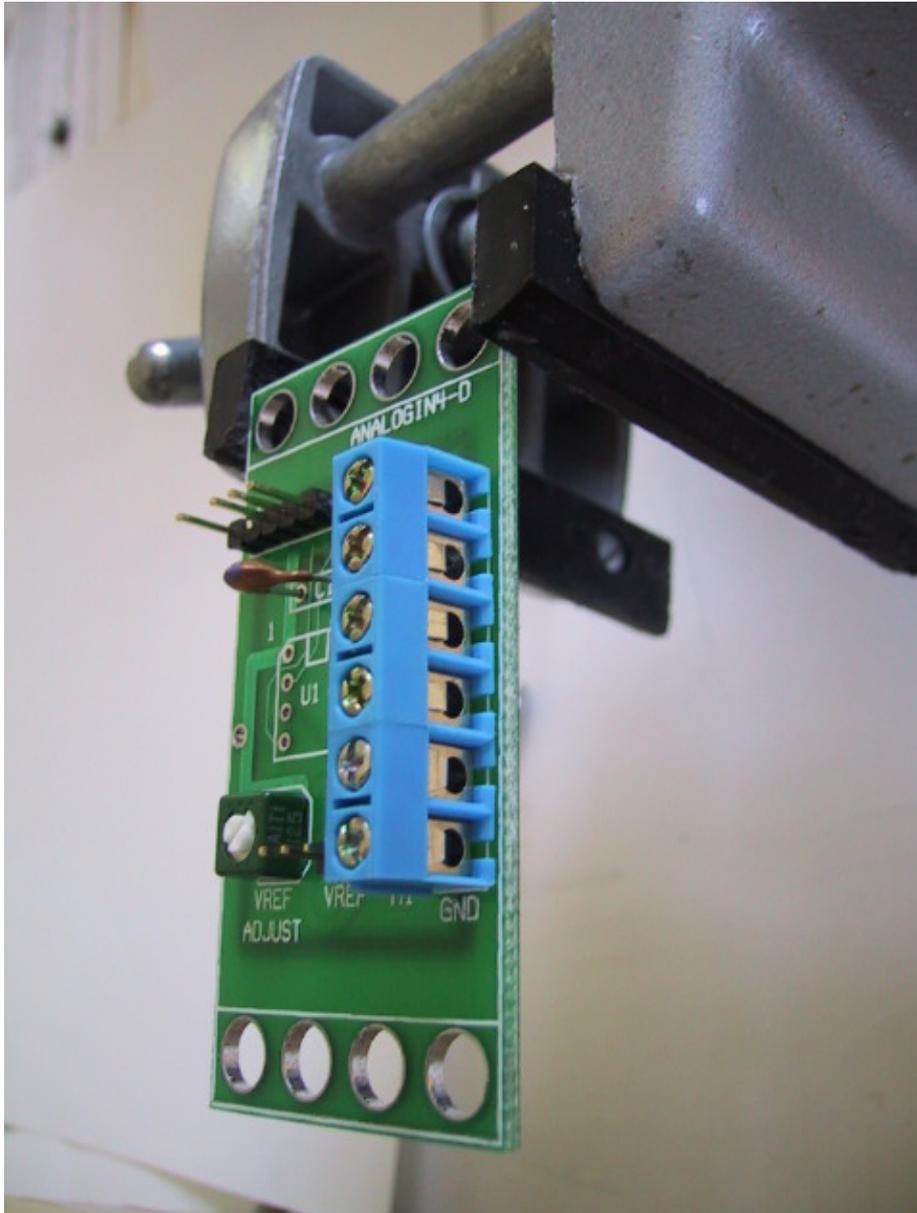
Step 4



Step 5



Step 6



Step 7

