This is the Revision A verion of the AnalogIn4 RoboBrick. The status of this project is finished. This revision has been replaced by the Revision B version of this RoboBrick.
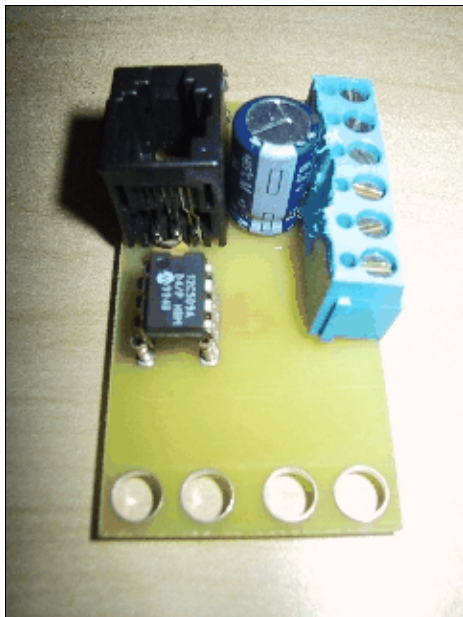
AnalogIn4 Robobrick (Revision A)

Table of Contents

This document is also available in PDF format.

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.

A picture of the AnalogIn4−A RoboBrick is shown below:

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 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 coversion results in

a 1. Whenever the digital conversion is lower than the low threshold register, the down conversion results in a 0. A hysterisis effect can be introduced by having some spread between the high and low threshold values.

After the down coversions 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
- 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.

In addition to the common shared commands and the shared interrupt commands, the AnalogIn4 RoboBrick supports following commands:

Read Pin
    Read the 8−bit value for a particular pin.
Read Binary Values
    Read and return the down convert binary values (after XOR'ing with complement mask.)
Read Raw Binary
    Read and return the raw binary values without applying the complement mask.
Read Complement Mask
    Read and return the complement mask.
Read High Mask
    Read and return the high mask.
Read Low Mask
    Read and return the low mask.
Read Raising Mask
    Read and return the raising mask.
Read Falling Mask
    Read and return the falling mask.
Set High Threshold
    Set high threshold for a specified pin.
Set Low Threshold
    Set low threshold for a specified pin.

Read High Threshold

> Read and return high threshold for a specified pin.

Read Low Threshold

> Read and return low theshold for a specified pin.

Set Complement Mask

> Set the complement mask.

Set High Mask

> Set the high mask.

Set Low Mask

> Set the low mask.

Set Raising Mask

> Set the raising mask.

Set Falling Mask

> Set the falling mask.

These commands are summarized in the table below:

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

Set Low Mask	Send	0	1	0	1	*l*	*l*	*l*	*l*	Set low mask to *llll*
Set Raising Mask	Send	0	1	1	0	*r*	*r*	*r*	*r*	Set raising mask to *rrrr*
Set Falling Mask	Send	0	1	1	1	*f*	*f*	*f*	*f*	Set falling mask to *ffff*
Read Interrupt Bits	Send	1	1	1	0	1	1	1	1	Return the interrupt pending bit *p* and the interrupt enable bit *e*.
	Receive	0	0	0	0	0	0	*e*	*p*	
Set Interrupt Commands	Send	1	1	1	1	0	*c*	*c*	*c*	Set Interrupt Command *ccc*.
Shared Commands	Send	1	1	1	1	1	*c*	*c*	*c*	Execute common shared command *ccc*

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:
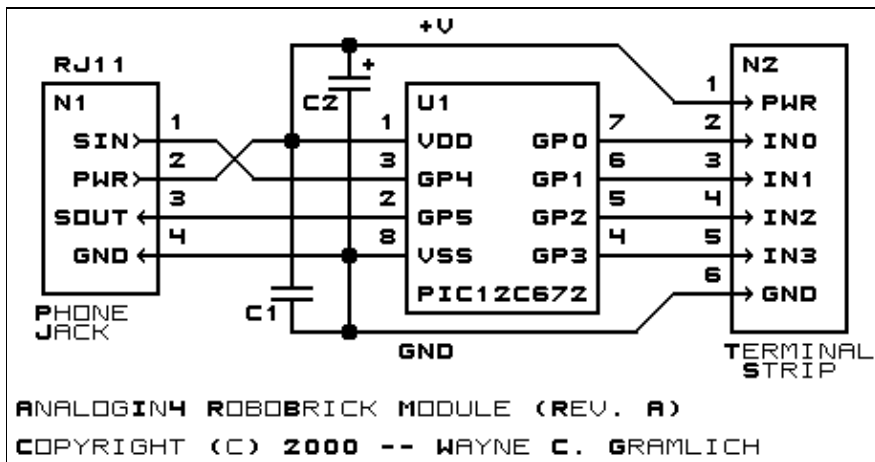
The parts list kept in a separate file –– analogin4.ptl.

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.

4. Software

The AnalogIn4 software is available as one of:

analogin4.ucl
>   The µCL 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 PIC programmer.

5. Issues

The following issues have come up:

- Make holes bigger for terminal strips.

A. Appendix A: Parts List

```
# Parts list for AnalogIn4 RoboBrick (Rev. A)
#
C1: Capacitor10pF - 10 pF Ceramic Capacitor [Jameco: 15333]
C2: Capacitor2200uF - 2200 uF 6.3V Electrolytic Capacitor [Jameco: 133145]
N1: RJ11Female4_4.RBSlave - Female RJ11 (4-4) Phone Jack [Digikey: A9071-ND]
N2: TerminalStrip6.AnalogIn4 - 6 Junction Terminal Strip [2 Jameco: 189667]
U1: PIC12C672.AnalogIn4 - Microchip PIC12C672 [Digikey: PIC12C672-04/P-ND]
```

B. Appendix B: Artwork Layer
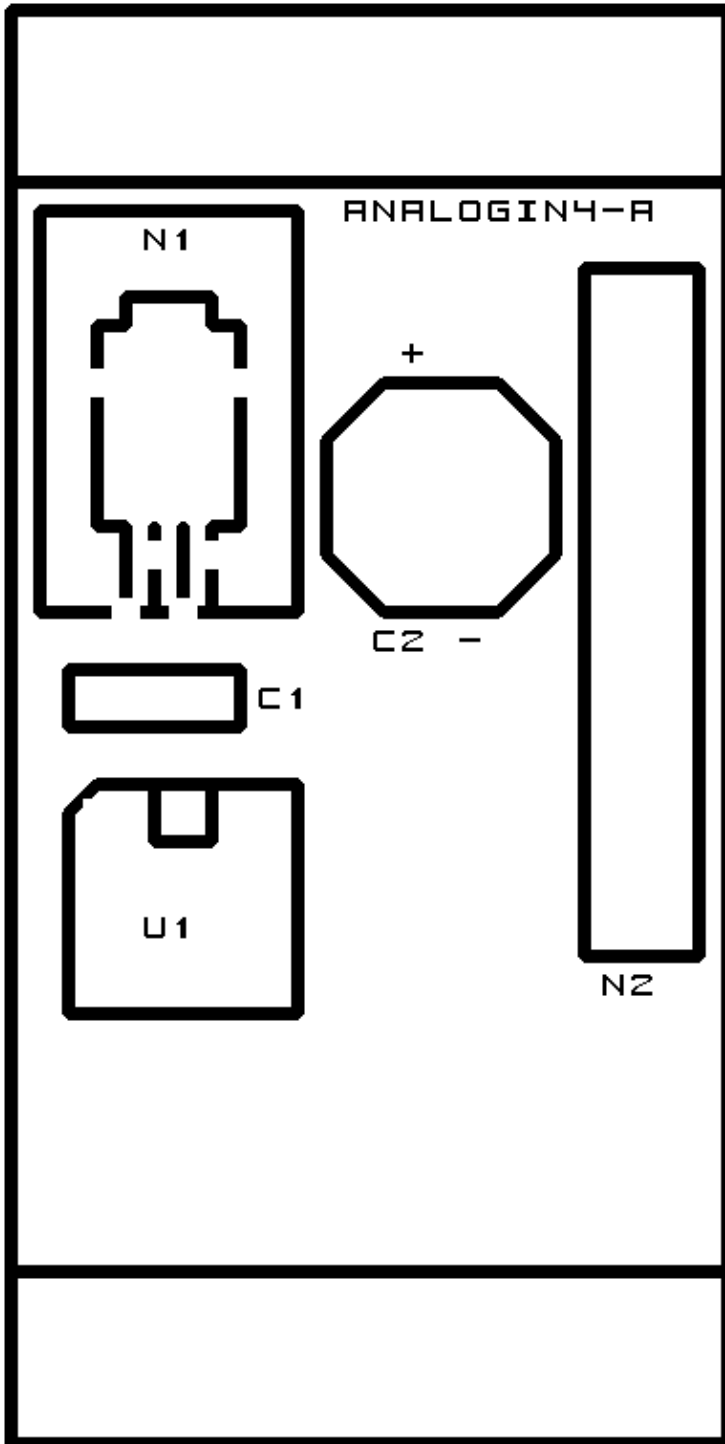
C. Appendix C: Back (Solder Side) Layer
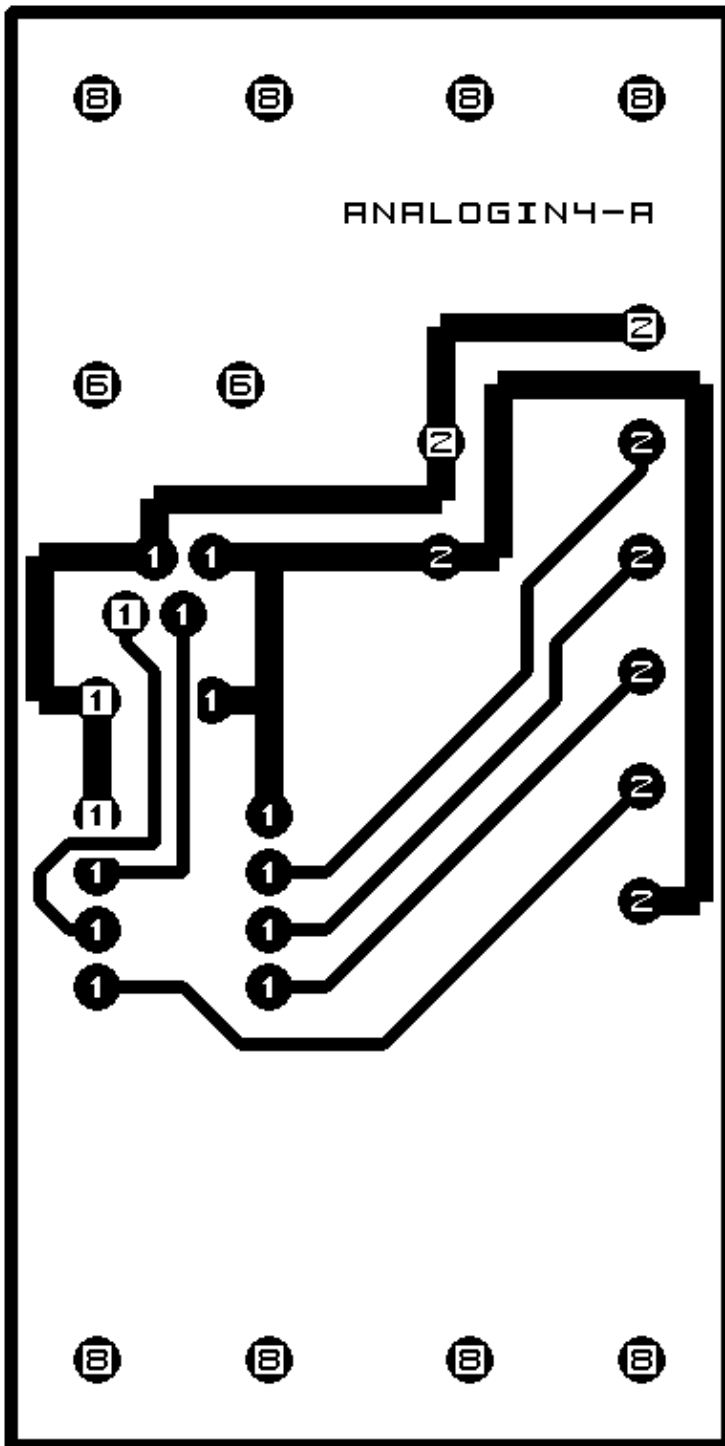
D. Appendix D: Front (Component Side) Layer

ANALOGIN4-A