

This is the Revision A version of the [Debug16 RoboBrick](#). The status of this project is [work in progress](#).

# Debug16 Robobrick (Revision C)

## 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](#)
- [4. Software](#)
- [5. Issues](#)

## 1. Introduction

The Debug16 RoboBrick provides the ability to output 16 8-bit registers using a total of 12 LED's. The first 4 LED's specify which register is being viewed and the remaining 8 LED's specify the register contents. Three buttons are used to increment, decrement, and reset the 4 LED register index.

Many RoboBricks have an additional debugging port that is designed to talk to the Debug16 RoboBrick. Thus, the Debug16 can be used to view the current state of many of the other RoboBricks.

## 2. Programming

The Debug16 RoboBrick supports the [standard shared commands](#) in addition to the following commands:

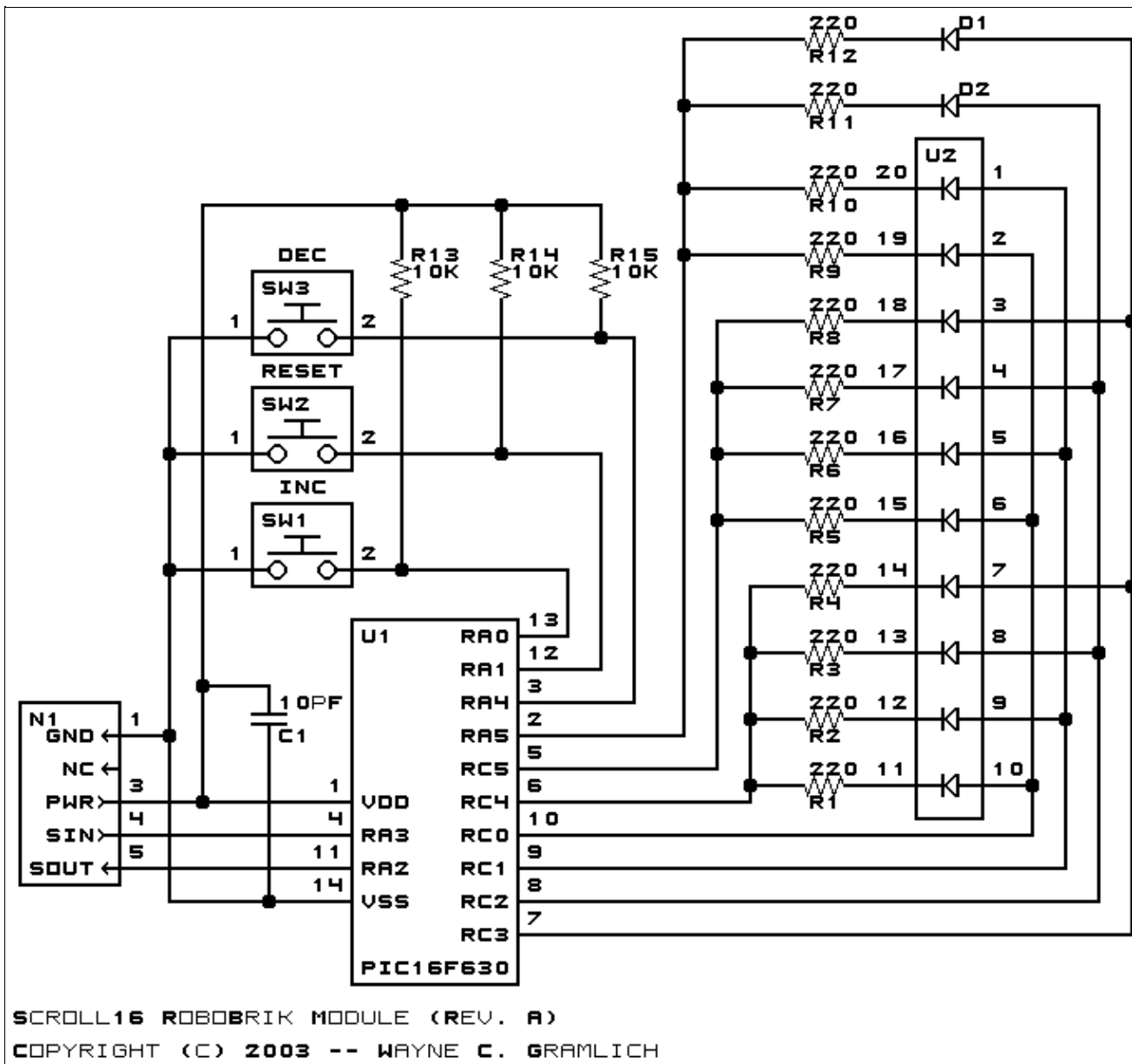
Command	Send/Receive	Byte Value								Discussion
		7	6	5	4	3	2	1	0	
Set Index and Low	Send	0	0	x	x	x	x	l	l	Set internal index register to <i>xxxx</i> and set low order 2 bits of register to <i>ll</i> .
Set Index and High	Send	0	0	h	h	h	h	h	h	Set high order bits of internal index register to <i>hhhhhh</i> .
<a href="#">Shared Commands</a>	Send	1	1	1	1	1	<i>a</i>	<i>b</i>	<i>c</i>	Send shared command <i>abc</i> to RoboBrick.

## 3. Hardware

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

### 3.1 Circuit Schematic

The schematic for the Debug16 RoboBrick is shown below:



The parts list kept in a separate file --- [debug16.ptl](#).

## 3.2 Printed Circuit Board

The printed circuit board files are listed below:

[debug16\\_back.png](#)

The solder side layer.

[debug16\\_front.png](#)

The component side layer.

[debug16\\_artwork.png](#)

The artwork layer.

[debug16.gbl](#)

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

[debug16.gtl](#)

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

[debug16.gal](#)

The RS-274X "Gerber" artwork layer.

[debug16.drl](#)

The "Excellon" NC drill file.

[debug16.tol](#)

The "Excellon" tool rack file.

## 4. Software

The Debug16 software is available as one of:

[debug16.ucl](#)

The  $\mu$ CL source file.

[debug16.asm](#)

The resulting human readable PIC assembly file.

[debug16.lst](#)

The resulting human readable PIC listing file.

[debug16.hex](#)

The resulting Intel<sup>®</sup> Hex file that can be fed into a PIC12C5xx programmer.

The Debug16 test suite is available as one of:

[debug16\\_test.ucl](#)

The  $\mu$ CL source file.

[debug16\\_test.asm](#)

The resulting human readable PIC assembly file.

[debug16\\_test.lst](#)

The resulting human readable PIC listing file.

[debug16\\_test.hex](#)

The resulting Intel<sup>®</sup> Hex file that can be fed into a PIC16F628 programmer.

## 5. Issues

Any fabrication issues are listed here.

---

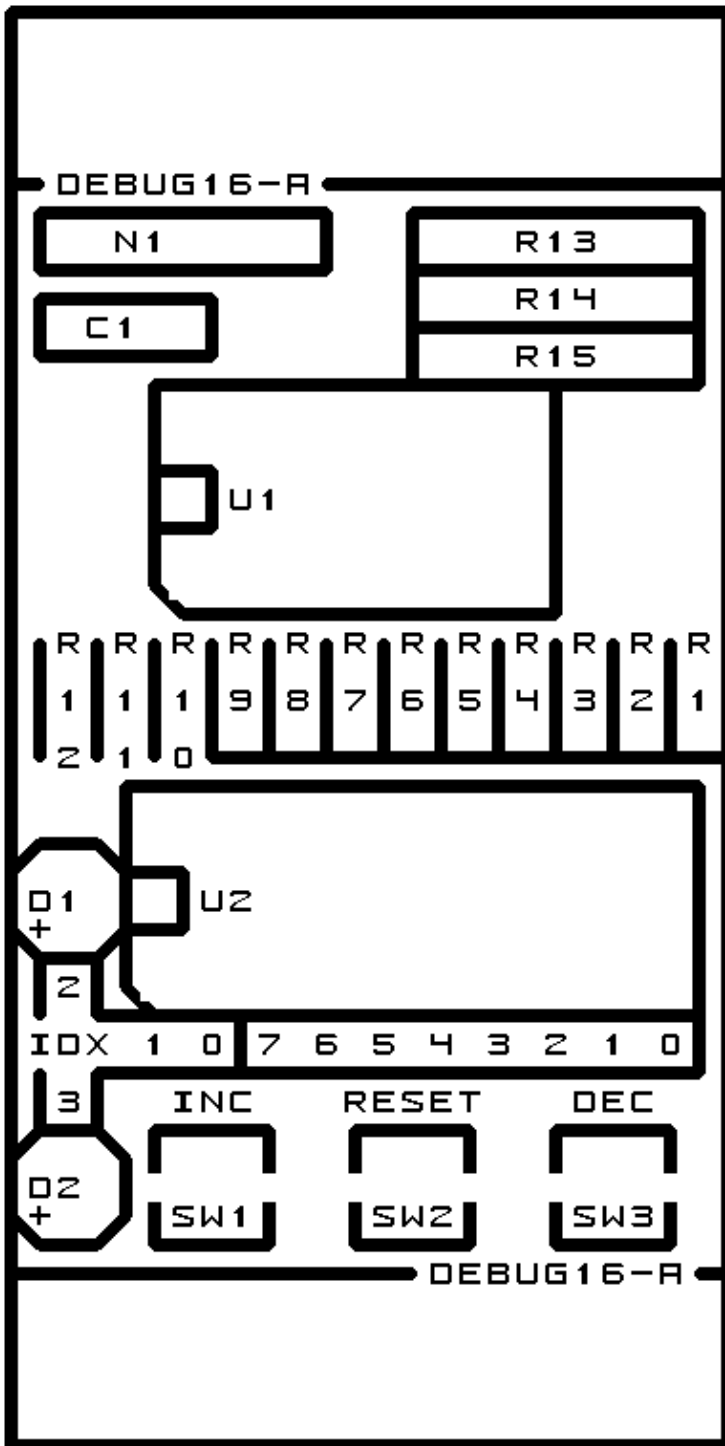
[Copyright](#) (c) 2000–2002 by [Wayne C. Gramlich](#). All rights reserved.



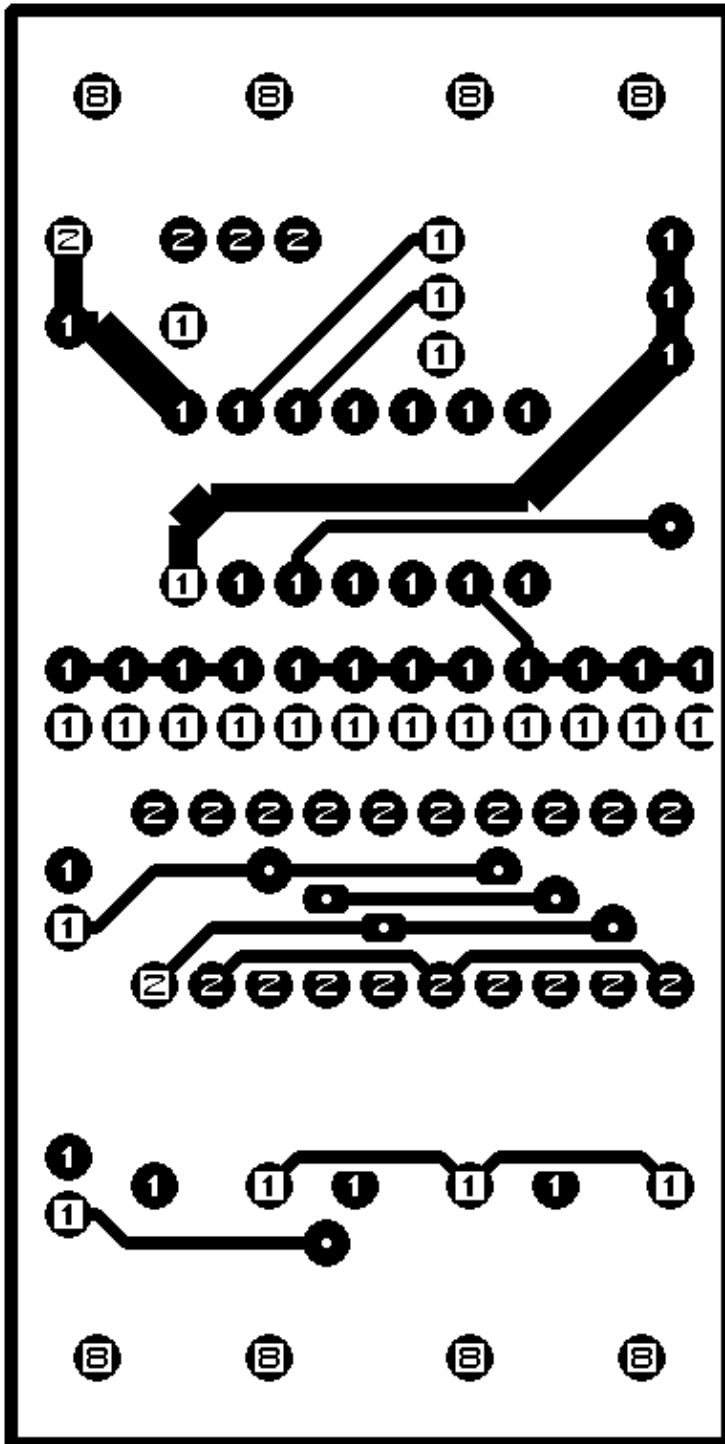
## A. Appendix A: Parts List

```
# Parts list for Debug16 RoboBrik (Rev. A)
#
C1: Capacitor10pF - 10 pF Ceramic Capacitor [Jameco: 15333]
D1-2: LEDGreen - Small Green LED [Jameco: 34606]
N1: Header1x5.Debug16 - 1x5 Male Header [5/40 Jameco: 160881]
R1-12: Resistor220.Vertical - 220 Ohm 1/4 Watt resistor [Jameco: 30470]
R13-15: Resistor10K - 10K Ohm 1/4 watt resistor [Jameco: 29911]
SW1-3: TactileButton - Normally Open Small Button [Digikey: EG2532CT-ND]
U1: PIC16F630.Debug16 - Microchip PIC16F630 [Digikey: PIC16F630-I/P-ND]
U2: LED10Bar.Green - 10 Bar Green LED [Digikey: MV54164-ND]
```

## B. Appendix B: Artwork Layer



### C. Appendix C: Back (Solder Side) Layer



### D. Appendix D: Front (Component Side) Layer

