Micromint Modules

MicroBolt

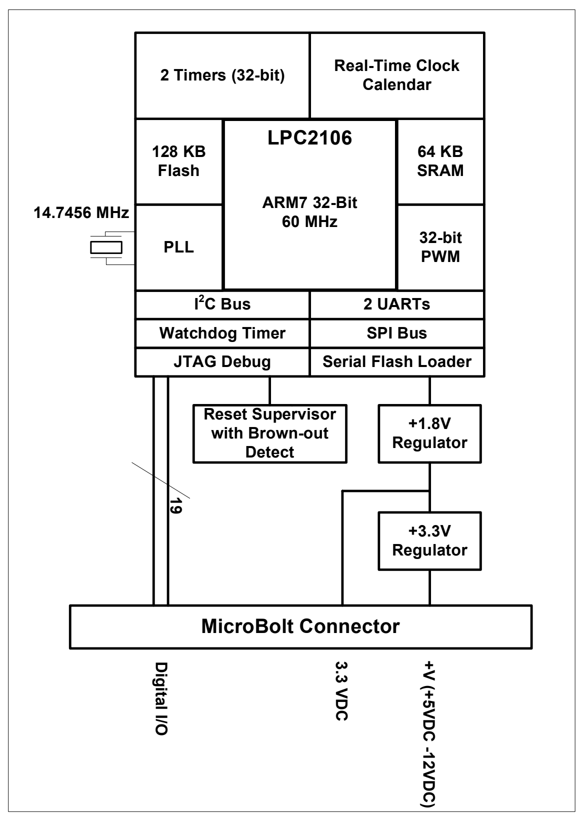
Microcomputer/Controller Featuring the Philips LPC2106

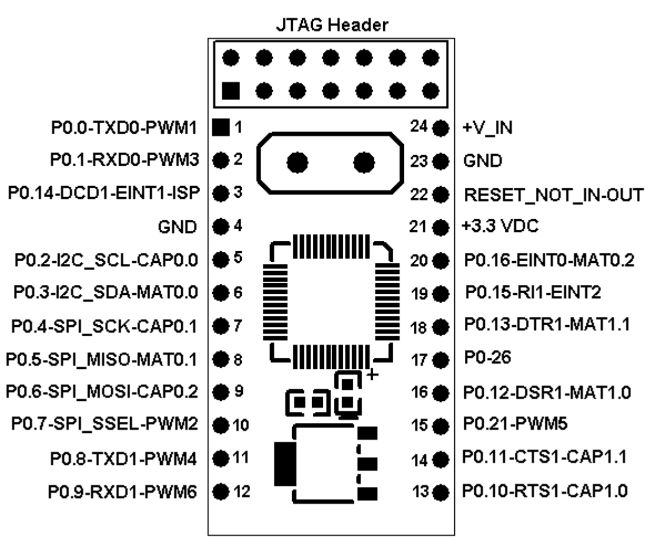
FEATURES

- Powerful 60 MHz, 32-bit ARM processing core.
- Pin compatible with 24 pin Stamp-like controllers.
- Small size – complete computer/controller with I/O less than 1 cubic inches (0.75" x 1.48" x 0.65")
- Low power, only 200 mW typical.
- Dual powered – operates on +3.3V or +5-12V (e.g., 5V @ 40 mA typical)

- Program and Data Memories
  - 128K Bytes of In-System Reprogrammable Flash with 10,000 Write/Erase Cycles
  - In-System Programming by on-chip serial flash loader or external JTAG programmer.
  - 64K Bytes SRAM

- Peripheral Features
  - I²C two-wire Serial Interface
  - Master/Slave SPI Serial Interface
  - Two programmable UARTs (TTL), one with complete modem interface signals.
  - Two 32-bit Timer/Counters with Separate Prescaler and Compare Modes
  - One 32-bit PWM giving 6 PWM Channels with Programmable Resolution from 1 to 32 Bits
  - Real-Time Clock Calendar
  - Programmable Watchdog Timer
  - Three external level sensitive interrupts
  - JTAG debugging
  - 19 Digital I/O that can sink or source 4 mA each
  - Reset supervisor with brown-out detection
  - JTAG debugger connector directly pin compatible with Nohau EMUL-ARM 14 pin header option.





ABSOLUTE MAXIMUM RATINGS

Operating Temperature: Commercial	0°C to +70°C
Storage Temperature	-50°C to +125°C
Voltage on +V (Pin 1) referenced to GND	0 to +12 VDC
Voltage on +3.3V (Pin 21) referenced to GND With pin 24 open	0 to +3.6 VDC Regulated

PIN DESCRIPTIONS

MicroBolt is a 24-pin package (0.75" x 1.48" x 0.65") with 0.1" pin and 0.6" row spacing. Most pins have multiple functions depending on system configuration. **DIO – Digital Input/Output**

Pin	Signal	Description
1	P0.0	DIO/TXD0/PWM1 (Transmitter output for UART 0 or Pulse Width Modular output 1). This pin is also used for device programming through the serial flash loader.
2	P0.1	DIO/RXD0/PWM3 (Receiver input for UART 0 or Pulse Width Modular output 3). This pin is also used for device programming through the serial flash loader.
3	P0.14	DIO/DCD1/EINT1/ISP (Data carrier detect input for UART 1 or Active low external interrupt input 1). This pin is also required for device programming through the serial flash loader and is held low during reset to put the device into the flash loading mode.
4	GND	Digital Ground
5	P0.2	DIO/ I²C-SCL/CAP0.0 (Clock for I²C serial communications or Capture input for Timer 0, channel 0). When used as I²C-SCL, an external pull-up resistor is required.
6	P0.3	DIO/ I²C-SDA/MAT0.0 (Data for I²C serial communications or Match output for Timer 0, channel 0). When used as I²C-SDA, an external pull-up resistor is required.

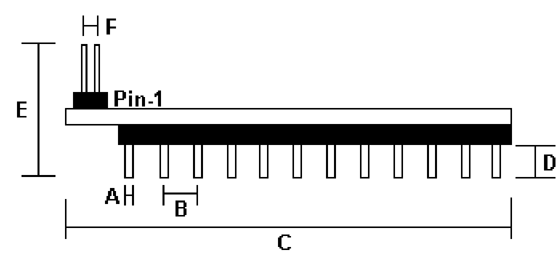
Pin	Signal	Description
7	P0.4	DIO/ SPI-SCK/CAP0.1 (Serial clock for SPI serial communications or Capture input for Timer 0, channel 1).
8	P0.5	DIO/ SPI-MISO/MAT0.1 (MISO data for SPI serial communications or Match output for Timer 0, channel 1).
9	P0.6	DIO/ SPI-MOSI/CAP0.2 (MOSI data for SPI serial communications or Capture input for Timer 0, channel 2).
10	P0.7	DIO/ SPI-SSEL/PWM2 (Slave select input for SPI serial communications or Pulse Width Modular output 2). When used as SPI-SSEL, an external pull-up resistor is required (Note: When in master mode, this pin must remain pulled up high and cannot be used as a DIO or slave select output.)
11	P0.8	DIO/ TXD1/PWM4 (Transmitter output for UART 1 or Pulse Width Modular output 4).
12	P0.9	DIO/ RXD1/PWM6 (Receiver input for UART 1 or Pulse Width Modular output 6).

Pin	Signal	Description
13	P0.10	DIO/ RTS1/CAP1.0 (Request to send output for UART 1 or Capture input for Timer 1, channel 0).
14	P0.11	DIO/ CTS1/CAP1.1 (Clear to send input for UART 1 or Capture input for Timer 1, channel 1).
15	P0.21	DIO/ PWM5 (Pulse Width Modular output 5).
16	P0.12	DIO/ DSR1/MAT1.0 (Data set ready input for UART 1 or Match output for Timer 1, channel 0).
17	P0.26	DIO.
19	P0.15	DIO/ RI1/EINT2 (Ring indicator input for UART 1 or Active low external interrupt input 2).
20	P0.16	DIO/EINT0/MAT0.2 (Active low external interrupt input 0 or Match output for Timer 0, channel 2).

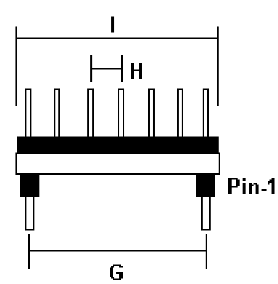
Pin	Signal	Description
21	+3.3V	Internal power supply voltage for the LPC2106. This output may be used to power minimal external circuitry or sensors. The MicroBolt may be powered by +3.3V, only through this pin, provided Pin 24 is left unconnected.
22	Reset*	Open drain RESET_NOT input and output. The Reset line functions as a normal active low reset input, but provides output functionality given it's open drain output and pull-up resistor. The MicroBolt remains in reset during power-up for 350 mS after a valid power supply voltage has been detected. Reset is also activated during power down and brown-out voltage detection. The MicroBolt Reset line can be used to provide an active low reset signal to external circuitry.
23	GND	Digital Ground
24	+V_IN	MicroBolt power supply input. +V is nominally 5-12 VDC. If pin 1 is open, the MicroBolt can be powered with +3.3 VDC directly on pin 21.

MECHANICAL AND ENVIRONMENTAL CHARACTERISTICS

Length	1.475 inches
Width	0.745 inches
Height	0.645 inches
Operating Temperature	0°C to +70°C



	Inches		Millimeters	
DIM	min	max	min	max
A	0.023	0.025	0.584	0.635
B	0.097	0.103	2.46	2.62
C	1.465	1.485	37.21	37.72
D	0.117	0.119	2.97	3.022
E	0.635	0.655	16.12	16.63
F	0.097	0.103	2.46	2.62
G	0.590	0.610	14.99	15.49
H	0.097	0.103	2.46	2.62
I	0.735	0.755	18.67	19.18



DC ELECTRICAL CHARACTERISTICS

Operating Temperature	Ta = 0°C to + 70°C
Operating Voltage	Vcc = 3.0 V to 3.6 V
	Vss = 0.0 V

Characteristic	Minimum	Typical	Maximum	Units	Condition
Supply Voltage					
(V_{CC} to Pin 21)	3.0	3.3	3.6	V	
(+V to Pin 24)	4.5	9	12	V	
Supply Current (Icc)		40	60	mA	
Input Low Voltage (V_{IL})			0.8	V	
Input High Voltage (V_{IH})	2.0		5.5	V	
Output Low Voltage (V_{OL})			0.8	V	I_{OL} = 4 mA, Vcc = 3.3V
Output High Voltage (V_{OH})	3.0			V	I_{OH} = -4 mA, Vcc = 3.3V

1.0 MicroBolt Hardware Overview

MicroBolt is an industrial oriented controller in a 1.48" by 0.75" 24-pin DIP circuit board module. It contains a 32-bit, Philips LPC2106, ARM7 enabled microcontroller that includes on chip Flash, SRAM, and other features as shown in the above block diagram.

1.1 LPC2106 Microcontroller

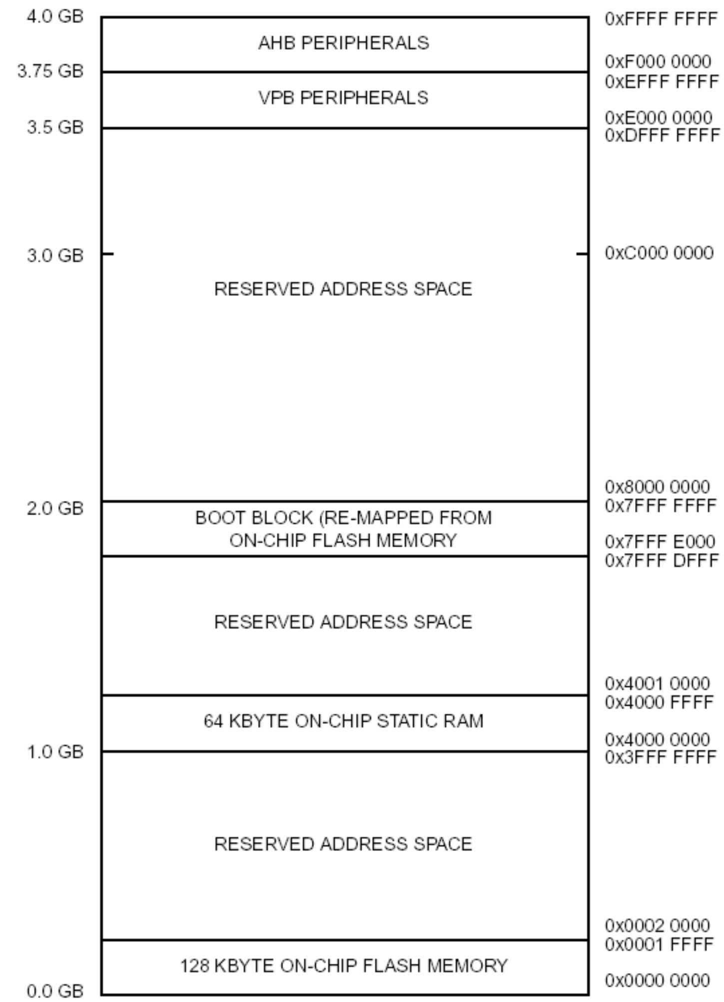
The LPC2106 has the following features available to the MicroBolt user: High speed processing via the internal 60 MHz 32 bit ARM7TDMI-S processor, 128K Bytes on-chip Flash Program Memory (128 bit wide interface/accelerator enables high speed 60 MHz operation.), 64K Bytes internal SRAM, In-System Programming (ISP) and In-Application Programming (IAP) via on-chip boot-loader software (Flash programming takes 1 ms per 512 byte line. Single sector or full chip erase takes 400 ms.), Vectored Interrupt Controller with configurable priorities and vector addresses, JTAG in system debugging, Multiple serial interfaces including two UARTs (16C550), Fast I²C (400 kbits/s) and SPI, Two 32-bit timers (7 capture/compare channels), PWM unit (6 outputs), Real Time Clock and Watchdog, Nineteen 5 V tolerant general purpose I/O pins, 60 MHz maximum CPU clock available from programmable on-chip Phase-Locked Loop, Two low power modes: Idle and Power-down, Processor wake-up from Power-down mode via external interrupt, Individual enable/disable of peripheral functions for power optimization. More details regarding the LPC2106 can be found in the Philips LPC2106 datasheet at http://www.philips.com/microcontrollers

2.0 Memory Map

MicroBolt memory is broken up into four different sections; Flash for program space, SRAM for volatile data storage, Peripherals, and the Boot block.

2.1 MicroBolt Program Space

MicroBolt has a total of 128 KB of program space.

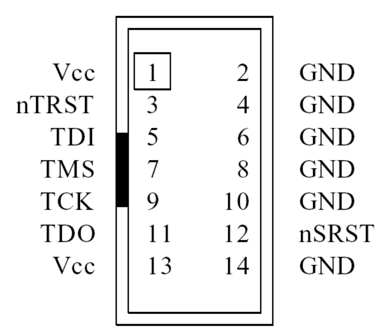


2.3 MicroBolt SRAM Data Memory

MicroBolt has 64 KB of SRAM for volatile data storage.

2.4 MicroBolt's Optional JTAG Connector

MicroBolt has the ability to have a JTAG connector installed on the module. The following diagram is the pin-out of the JTAG connector.



MicroBolt 14 Pin JTAG Header

3.0 MicroBolt Software

When it comes from the factory, the MicroBolt has no software on the board itself. Programs are developed using cross-development tools running on a desktop PC and is programmed into the MicroBolt for execution. There are several development environments from which to choose for the LPC2106 microcontroller, but the recommended IDE for the MicroBolt is the ImageCraft ICCV7 for ARM IDE with ANSI C compiler.

3.1 ImageCraft ICCV7 For ARM C Compiler

ICCV7 for ARM is an IDE with an ANSI C compiler for ARM code generation that also contains an Application Builder GUI that supports the MicroBolts LPC2106, editor, project manager, and code browser. It also contains support for ELF/DWARF debugging and help files are available as HTML Help and PDF. The IDE itself runs under Windows 95, 98, Me, NT 4.0, 2000 and XP. It is an easy to use integrated development Environment with a built-in serial communication terminal for debugging and has an editor with auto indentation and keywords highlighting. The C Compiler supports all the ANSI C types. ICCV7 for ARM also supplies MicroBolt users with a free LPC2106 library (pec.lib) from Pride Embedded, LLC. This library provides a demo project that contains the following functionality: Multiple C and header files to allow for ease of understanding, PLL setup and initialization, GPIO pin function selection and I/O direction (inputs, outputs, and special function pin setup shown), Philips ARM VIC (Vectored Interrupt Controller) setup and prioritization, Timer-0: Setup, initialization, ISR (Interrupt Service Routine), Timer-1: Setup, initialization, ISR, PWM-0: Setup, initialization, ISR, UART-0:Setup, initialization, baud rate selections, ISR on received character and string storage and parsing, transmit routine, External Interrupt-1: Setup, ISR, code for converting level-sensitive interrupt to edge-sensitive (1 interrupt instead of many), Delay routines (measured for accuracy), I2C communications, SPI communications, and more. Micromint also provides users many example projects based upon this library to get up and running quickly.

4.0 Programming the MicroBolt

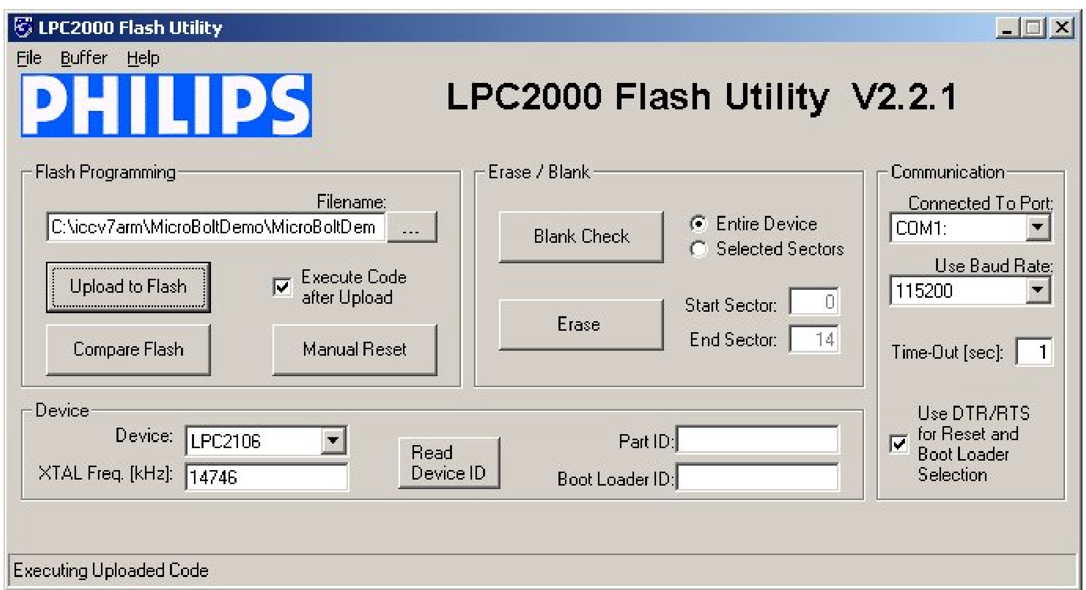
The user-programmable part of the MicroBolt uses Philip's LPC2106 FLASH micro-controller that can be reprogrammed thousands of times. These programs can be created using a number of resources, as described above. Programming the MicrBolt is done through UART 0, via the RS232 serial port, by using the LPC2106 Flash Utility IDE. Programming can also be accomplished through the on-board JTAG debugging connector, but serial port programming will be discussed here since it's the most straightforward method of programming. **Note: The MicroBolt does not contain an RS232 transceiver on-board so programming it over the serial port requires the MicroBolt development board or an external RS232 transceiver connected to UART 0.**

4.1 Using the Philips Flash Utility

The Philips Flash Utility is a free tool as provided by Philips. It is used to program a program into the MicroBolt module. It is very easy to use. There are only six steps to send a program to the MicroBolt.

Step

1. Open the Philips flash utility.
2. Select the appropriate COM Port
3. Select the appropriate reset configuration (check Use DTR/RTS if using the MicroBolt development board).
4. Click on the top "…" button to find the hex file you want to send to the MicroBolt.
5. Click on the "Upload to Flash" button to send the files.
6. Cycle power to the board or reset the module.



5.0 MicroBolt Development Board

The MicroBolt Development Board was designed, for use as an evaluation platform, for prototyping additional circuitry around applications using the MicroBolt. The development board measures 6 inches by 6 inches. Circuitry for a SPI 2-channel 12-bit Analog to Digital converter, an I^2C EEPROM, LEDs, an LCD, and two RS-232 or one RS232 and one RS-485 communication ports.

5.10 Development Board Power Supply

An unregulated 12 VDC wall transformer with a 2.5mm power plug is supplied to power the board. The plugs center tap should be negative. A diode (D1) will protect the regulator if a power supply with the wrong polarity is accidentally plugged in. The development board can power the Microbolt with a regulated +5VDC by placing a jumper on JP17 or with a regulated +3.3VDC by placing a jumper on JP18. **WARNING: Inserting jumpers on both JP17 and JP18 will damage the MicroBolt.**

5.11 MCP3202 2-Channel 12-bit ADC

The MCP3202 (U4) is a successive approximation 12-bit Analog to Digital Converter with on board sample and hold circuitry. It is programmable to provide a single pseudo-differential input pair or dual single-ended inputs.

Communication to the ADC is done using MicroBolt's SPI bus. The ADC's chip select can be connected to P0.16 of the MicroBolt by adding a jumper to JP39. The data in signal can be connected to P0.5 of the MicroBolt by adding a jumper to JP14. The data out signal can be connected to P0.6 of the MicroBolt by adding a jumper to JP13. The clock signal can be connected to P0.4 of the MicroBolt by adding a jumper to JP12. In order for the SPI bus to operate JP11 must have a jumper connected to it. JP11 pulls P0.7 of the MicroBolt up to 3.3V through a 10kΩ resistor.

5.12 I^2C EEPROM

U7 is for an I^2C EEPROM. The recommended I^2C EEPROMs are from Microchip, the 24CXX, 24LCXX, 24AAXX, or 24FCXX (24XX*) family. The devices are 128-bit through 512 Kbit Electrically Erased PROMs. The devices are organized in blocks of x8-bit memory with 2-wire serial interfaces.

Communication to the I^2C EEPROM is done by using MicroBolt's I^2C bus. The EEPROM's data line can be connected to P0.2 of the MicroBolt by adding a jumper to JP37. The EEPROM's data line can be connected to P0.3 of the MicroBolt by adding a jumper to JP38. Some I^2C EEPROMs have the ability to change their base address by connecting A1, A2, and A3 to different combinations of +3.3V or ground. This can be done by adding jumpers to JP32, JP33, and JP34. Some I^2C EEPROMs have a Write Protect pin (WP). The WP pin can be connected to +3.3V or ground by adding a jumper to JP35.

5.13 LCD Connectors

The development board has two different connectors for LCDs a 14-pin single row header and a 14-pin 2x7 dual row header. Only one LCD can be connected at a time. Any LCD with a Hitachi 44780 (or equivalent) controller is recommended for use with the development board. The LCD needs to be communicated to in nibble mode. The contrast for the display can be adjusted by adjusting the potentiometer R5.

To communicate to the LCD six digital I/O need to be connected to the display. The LCDs data bit seven can be connected to P0.13 of the MicroBolt by adding a jumper to JP29. The LCDs data bit six can be connected to P0.12 of the MicroBolt by adding a jumper to JP28. The LCDs data bit five can be connected to P0.11 of the MicroBolt by adding a jumper to JP27. The LCDs data bit four can be connected to P0.10 of the MicroBolt by adding a jumper to JP26. The LCDs E signal can be connected to P0.21 of the MicroBolt by adding a jumper to JP25. The LCDs RS signal can be connected to P0.26 of the MicroBolt by adding a jumper to JP24.

5.14 UART0

MicroBolt's UART0 is used for programming and serial communications. UART0's transmit signal connects to the RS-232 level shifter by adding a jumper to JP7. UART0's receive signal connects to the RS-232 level shifter by adding a jumper to JP10. In order to program the MicroBolt it's reset and ISP signal should also be connected to the RS-232 level shifter. JP9 connects the reset signal and JP8 connects MicroBolt's ISP signal to the RS-232 level shifter. Figure 1 demonstrates the jumper settings for programming the MicroBolt.

The development board was designed to make MicroBolt's UART0 be a DTE (Data Terminal Equipment) or a DCE (Data Communication Equipment) device by changing two jumpers. JP1 connects the transmit or the receive signal to pin three of J1's DB9 connector. JP2 connects the transmit or the receive signal to pin two of J1's DB9 connector. Figure 2 demonstrates the jumper settings set as a DTE device making the MicroBolt act similar to a computer's serial port. If the user wanted to connect the development board to a modem with the jumpers set like Figure 3 a straight through cable would have to be used. Figure 3 demonstrates the jumper settings as a DCE device making the MicroBolt act similar to a computers peripherial device. If the user wanted to connect the development board to a computer with the jumpers set like Figure 2 a straight through cable would have to be used.

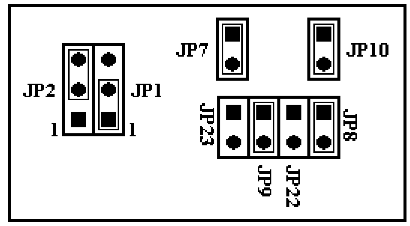


Figure 1: *Jumper Configuration for Programming*

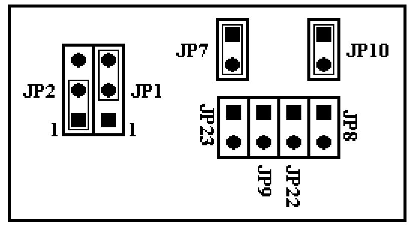


Figure 2: *Jumper Configuration for UART0 as a DTE Device*

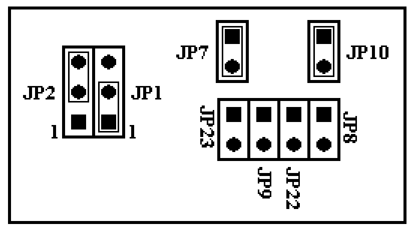


Figure 3: *Jumper Configuration for UART0 as a DCE Device*

5.15 UART1

MicroBolt's UART1 can be used for RS-232 or RS-485 serial communications. UART1's transmit signal connects to the RS-232 level shifter by installing a jumper across pins 1 and 2 of JP6. UART0's receive signal connects to the RS-232 level shifter by installing a jumper across pins 1 and 2 of JP5. Like UART0, UART1 can be configured as a DTE or a DCE device depending on how the jumpers are set. JP4 connects the transmit or the receive signal to pin three of J2's DB9 connector. JP3 connects the transmit or the receive signal to pin two of J2's DB9 connector. Figure 4 demonstrates how to set UART1 as a DTE. Figure 5 demonstrates how to set UART1 as a DCE.

UART1 can be configured for RS-485 communication by connecting the transmit and receive signals to the RS-485 driver. To connect the transmit signal to the driver a jumper needs to be installed across pins 2 and 3 of JP6. To connect the receive signal to the driver a jumper needs to be installed across pins 2 and 3 of JP5. In order for a RS-485 network to operate properly the RS-485 drivers transmitter needs to be able to be enabled and disabled. This can be accomplished by installing a jumper on JP19 to connect P0.15 to the disable/enable pin on the RS-485 driver. Figure 6 demonstrates how to set-up UART1 for RS-485 communications.

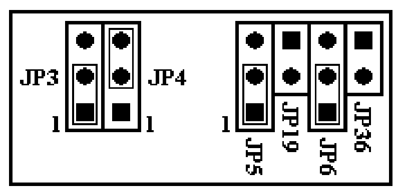


Figure 4: *Jumper Configuration for UART1 as a DTE Device*

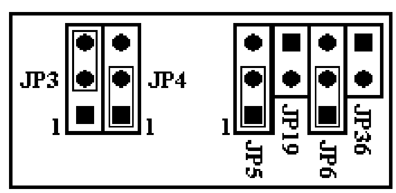


Figure 5: *Jumper Configuration for UART1 as a DCE Device*

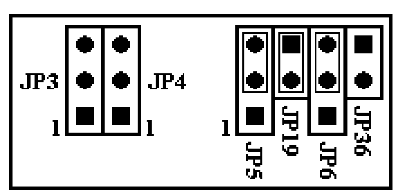
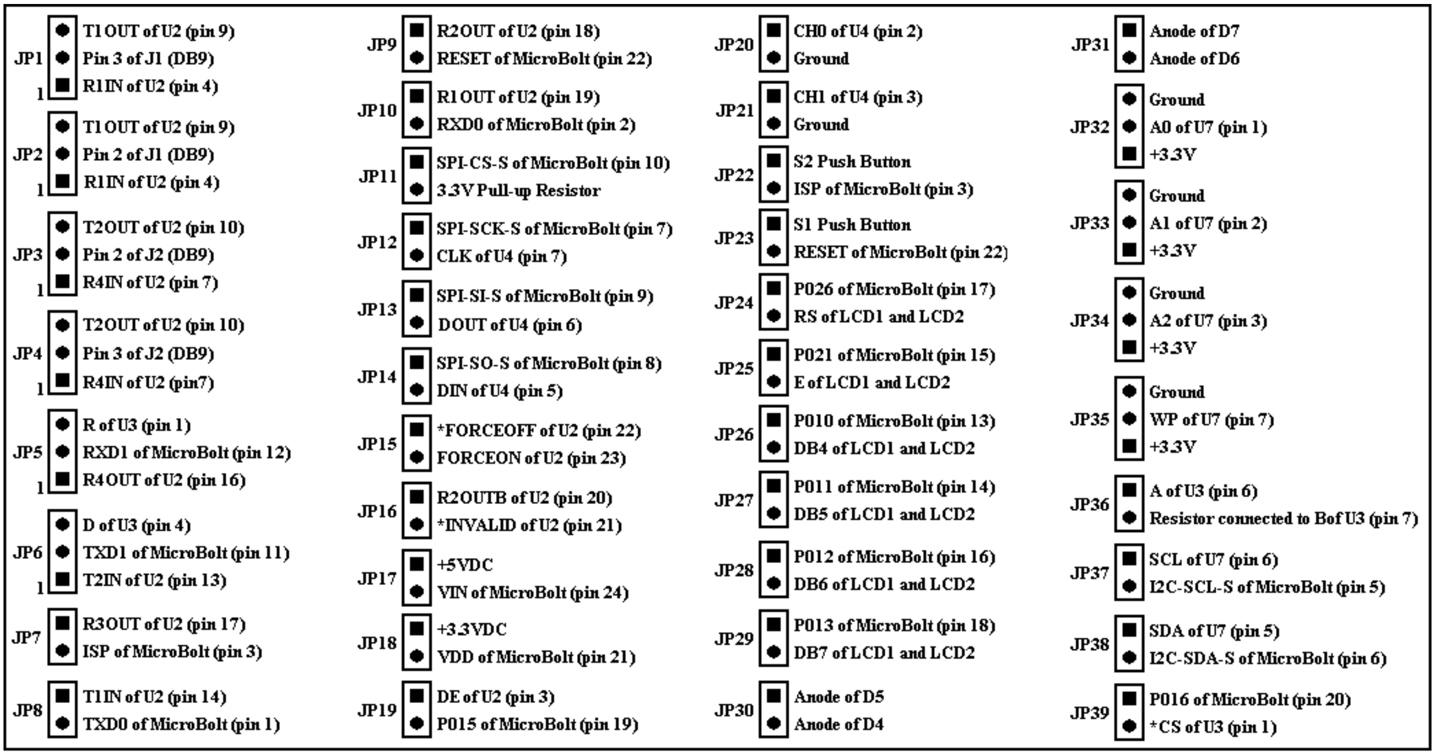
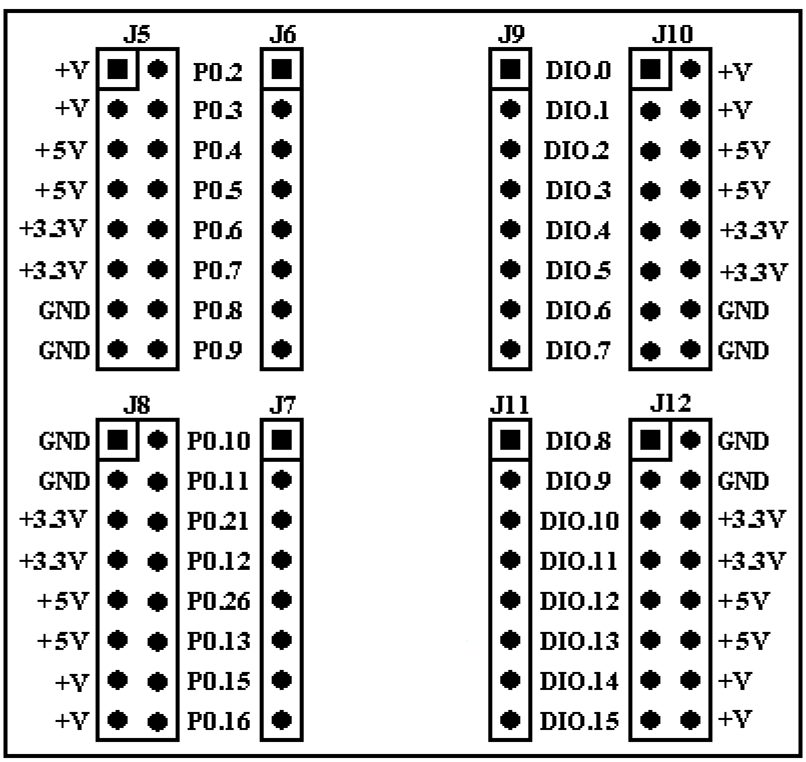


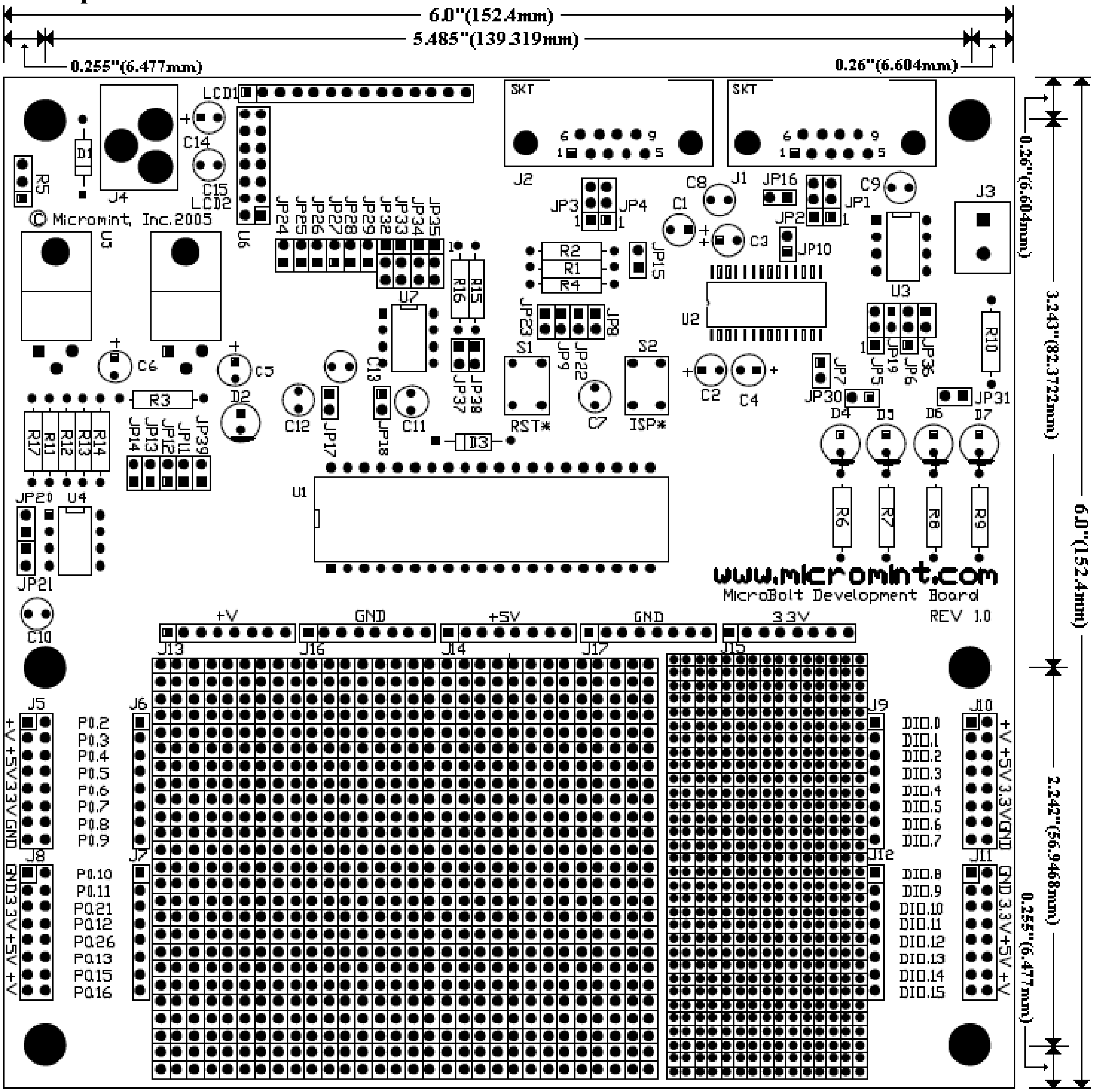
Figure 6: *Jumper Configuration for UART1 for RS-485 communication*

5.16 Header and Jumper Descriptions

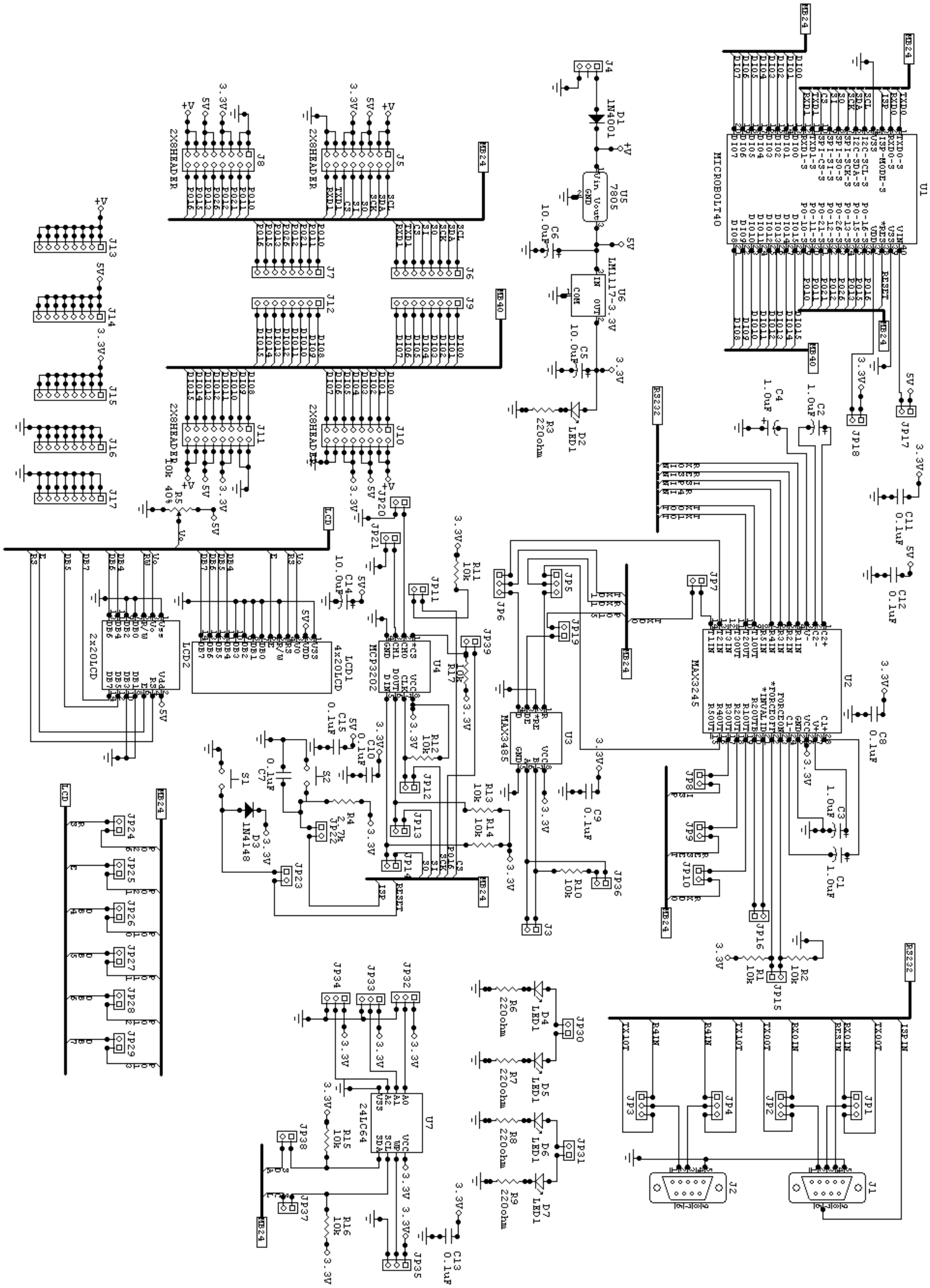




5.17 Development Board Dimensions



5.18 Development Board Schematic



Errata

MicroBolt Development Board

Error on PC board SPI data out (MOSI) and SPI data in (MISO) are swapped.

Please note that there is an error on REV 1.0 of the MicroBolt Development Board. The data in and the data out for the MCP3202, 2-channel 12-bit ADC need to be swapped. The simplest fix for this problem is to solder on wires in place of adding jumpers to connect the data lines to the MicroBolt module. The diagram below shows this more clearly.