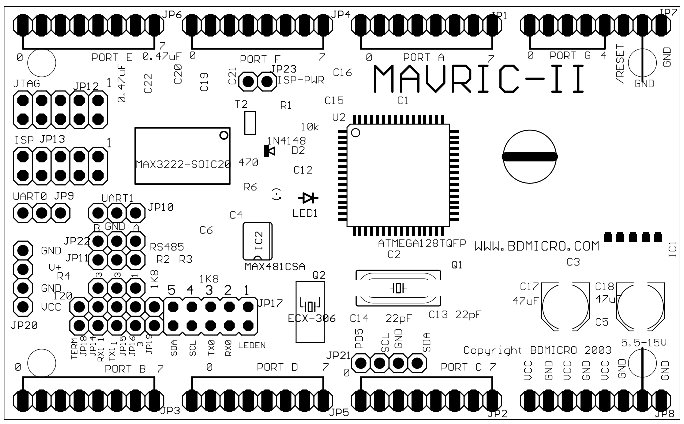
# MAVRIC-II

## Mega AVR Integrated Controller II
## Technical Manual




BDMICRO
http://www.bdmicro.com/

December 30, 2003

**DISCLAIMER**

BDMICRO products are designed and assembled with care and all assembled products must pass functional testing to ensure quality before they are shipped. However, our products are not rated for use in medical, life support, or systems where failure could result in the loss of life or have serious life threatening or economic impact.

BDMICRO does not warrant merchantability for any purpose and shall not be liable for any direct, indirect, incidental, special, exemplary, or consequential damages of any kind, however incurred, through the use of our products.

# Contents

# 1 Introduction

The BDMICRO MAVRIC-II is a powerful microcontroller board based on the ATmega128 MCU. It is fully programmable using familiar languages such as C and BASIC. The GNU GCC C compiler is very popular as it comprises the core of the de-facto standard tool chain of the open source community. Additionally, MCS Electronic's BASCOM-AVR BASIC compiler is popular, economical, and highly regarded by AVR enthusiasts.

The MAVRIC-II is the perfect microcontroller for many applications such as robotics where a fusion of many sensors and digital I/O are needed. Its copious program and data space allows you to implement sophisticated algorithms and logic without having to compromise features. MAVRIC-II's 8 channel, 10-bit A/D converter makes short work of hooking up analog sensors like the SHARP GP2D12 IR distance sensor. It has 2 on-board UARTs both of which are level-shifted to provide true RS232 levels, or they may be used using the unshifted TTL levels if that is more convenient. One of the UARTs can optionally be routed through the on-board RS485 transceiver which incorporates on-board termination and independent transmitter/receiver control. MAVRIC-II is $I^2C$ ready with on-board pull-ups so you can hook up your existing $I^2C$ devices such as your SRF08 Ultrasonic Ranger and your CMPS01 compass module, plus many other $I^2C$ peripherals. In addition to all that, 6 high resolution PWM outputs are available for driving servos and PWM motors with high precision. And with up to 53 digital I/O pins, it can handle even the most complex and demanding control tasks.

The on-board standard programming headers make development easy — 10-pin serial ISP header for hooking up to your AVRISP or STK500, and also the standard 10-pin JTAG header for hooking up to your JTAGICE for programming and single-step, source level debugging.

The MAVRIC-II incorporates an advanced, on-board, low-dropout voltage regulator and can be powered with as little as 5.5 Volts. Four mounting holes are provided for securely mounting your board, and with it's small size of 2.2 x 3.6 inches, you can mount it just about anywhere.

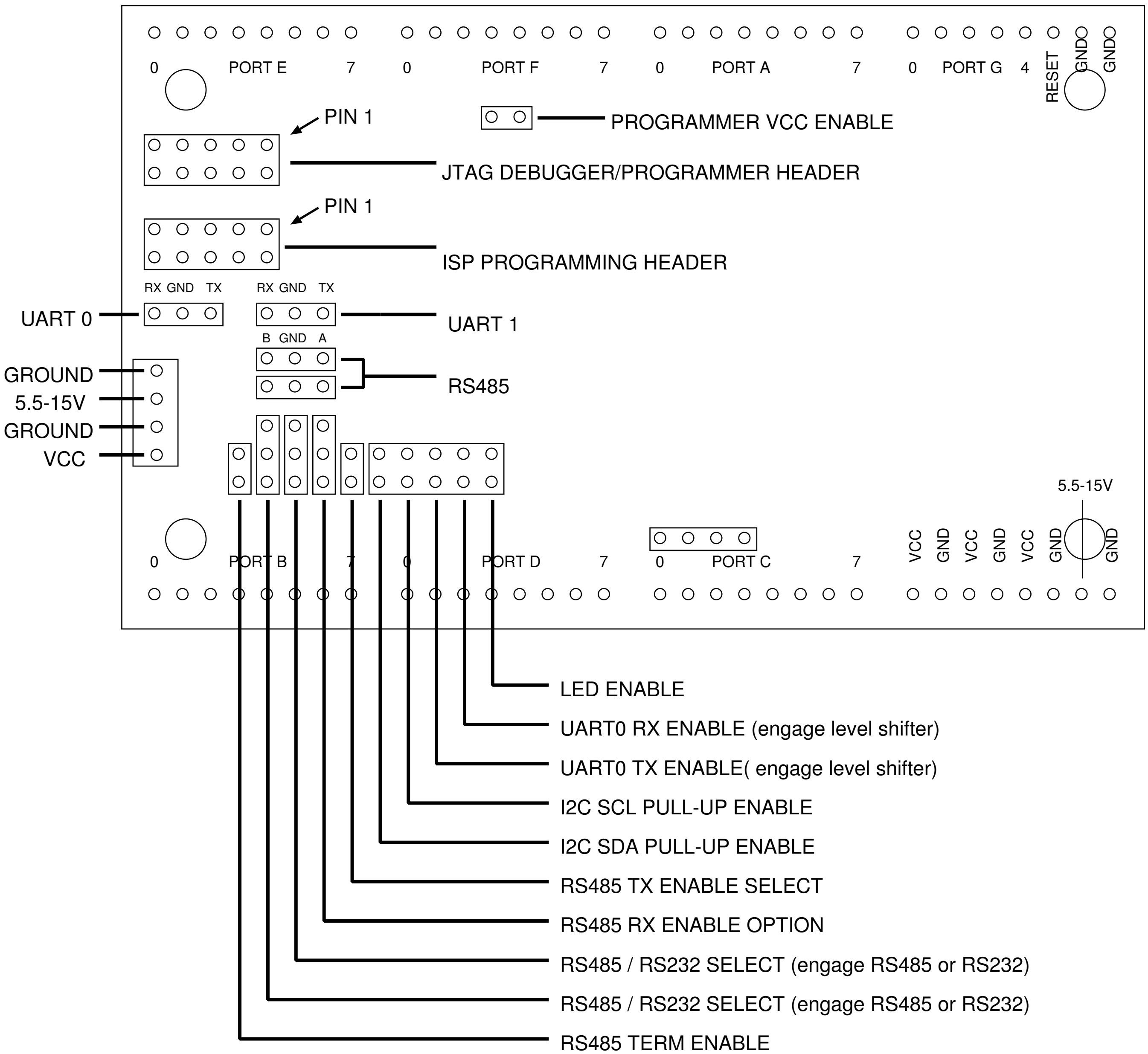
The capabilities of the ATmega128 MCU are numerous. Here are some of the highlights of the chip:

- 128K Flash memory (program space)
- 4K internal static RAM
- 4K EEPROM
- optional 64K external static RAM
- 8 channel analog to digital converter
- dual UARTs
- $I^2C$ interface
- up to 53 digital I/O pins
- JTAG interface for programming and debugging
- 2 8-bit timers, 1 16-bit timer
- 6 PWM channels
- Watchdog timer

For a more detailed look at the ATmega128, the best reference is the datasheet provided by the chip manufacturer, Atmel. The datasheet is available from Atmel in the form of a PDF file here:

`http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf`

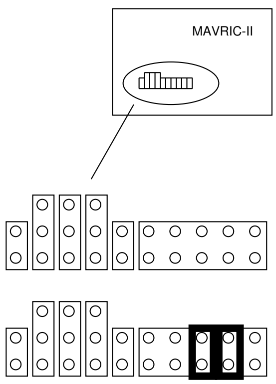
## 2 Headers and Jumpers




Note that all **VCC** connections are regulated 5.0 Volts, input or output, while all connections labeled **5.5-15V** are unregulated input only. If you are using an external regulated supply, connect to one of the **VCC** connections, but be sure not to hook up any power source to the unregulated input at the same time.

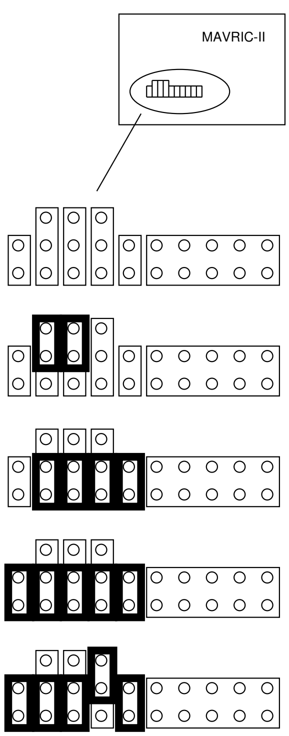
The **PROGRAMMER VCC ENABLE** jumper is used to connect the **VCC** pin present on each of the 10-pin JTAG and ISP headers to the MAVRIC-II **VCC** supply pin. This is useful because some programmers want to power the board, while other programmers want to be powered. The jumper allows the MAVRIC-II to work easily with any combination of programmer supply and external power supply configuration. When using a programmer that wants to be powered by the circuit, install the jumper and run the board from an external power supply. When using a programmer that wants to power the circuit, like the STK500, either install the jumper and disconnect the external power supply, or remove the jumper and run the MAVRIC-II from an external supply.

# 3 UART0 Configuration Options




1. UART0 disconnected from RS232 driver
2. UART0 connected to RS232 driver

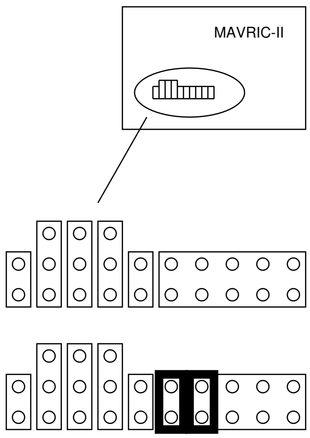
# 4 UART1 Configuration Options




1. RS232 disabled, RS485 disabled
2. RS232 enabled, RS485 disabled
3. RS485 enabled, TE on PD6, /RE follows TE, TERM disabled
4. RS485 enabled, TE on PD6, /RE follows TE, TERM enabled
5. RS485 enabled, TE on PD6, /RE follows PD7, TERM enabled

TE = Transmitter Enable
/RE = (not) Receiver Enable

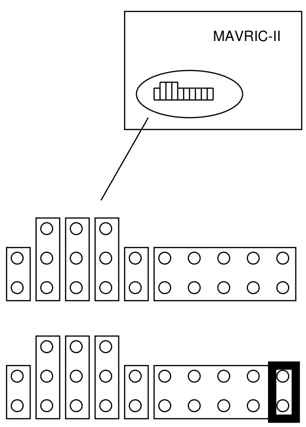
# 5 I2C Configuration Options




1. I2C Pull-ups disabled

2. I2C Pull-ups enabled

# 6 LED Configuration Options




1. LED disabled

2. LED connected to PB0

# 7 Fuse Bit Settings

Atmel AVR processors incorporate *fuse bits* which control various functions of the chip and persist even across a chip erase. By default, Atmel ships the ATmega128 with several fuse bits already programmed by default. Notably, the M103C fuse bit (ATmega103 compatibility mode) is enabled, as well as JTAGEN which enables the JTAG debugging lines of PORT F. Also, by default, the internal clock source is selected to run the processor at 1 MHz.

If you ordered a Kit, your ATmega128 processor's fuse bits are all at their default values and will need to be changed to run on the MAVRIC-II board at 16 MHz. For 16 MHz operation, you will need to program the CKOPT fuse bit (set to '0'). The clock selection lines will need to be modified as well: unprogram CKSEL3, CKSEL2, CKSEL1, and CKSEL0 (set to '1'). Also recommended when using the external crystal is to set SUT1 and SUT0 to '1' (unprogrammed). This setting is for slow rising power which may be necessary depending on your power supply. Additionally, you will need to unprogram the M103C fuse bit (set to '1') if you are compiling your code to run on an ATmega128 (as opposed to an ATmega103).

If you ordered an assembled and tested board, these fuse bits are already set appropriately to run at 16 MHz, and the M103C fuse bit has been unprogrammed.

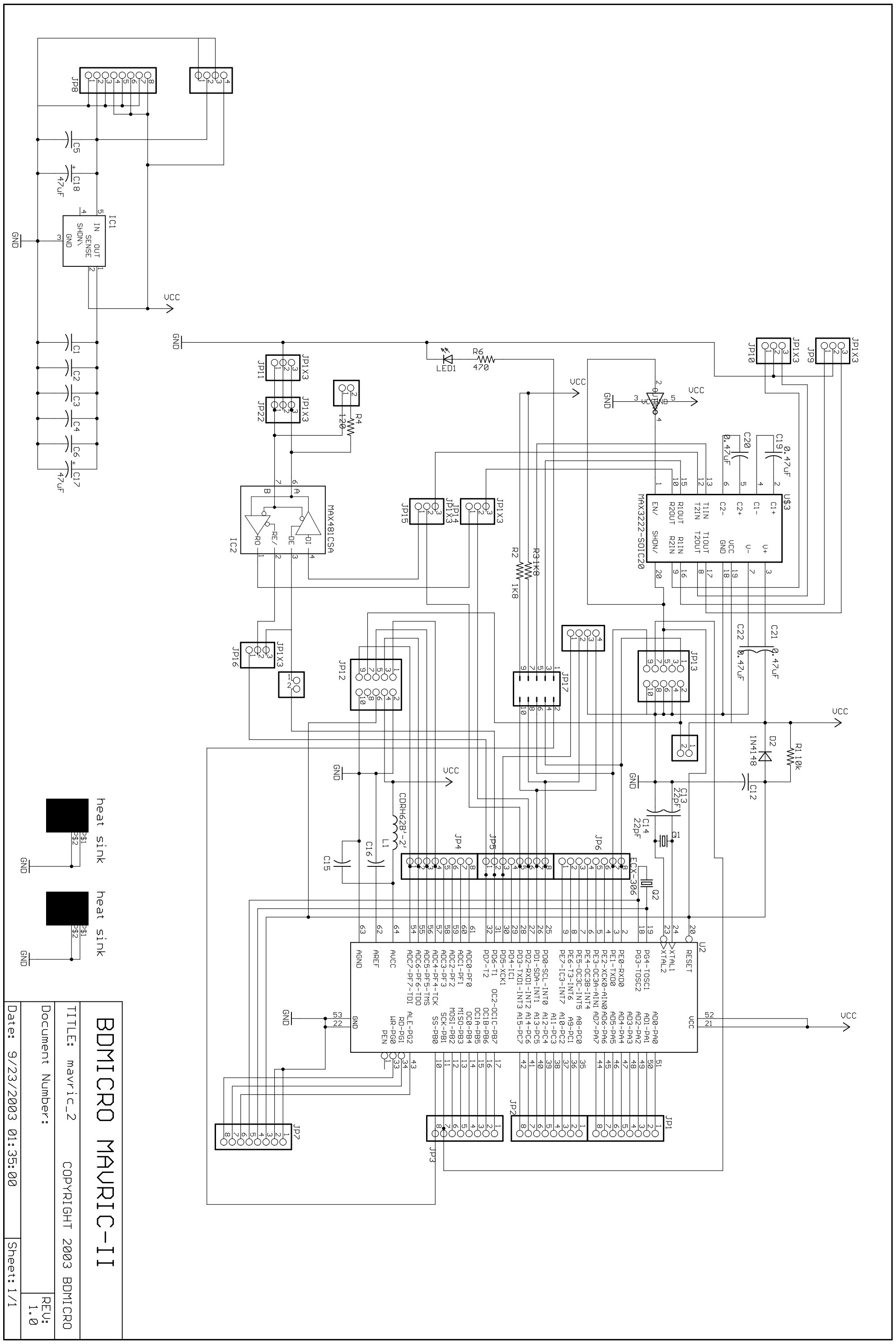
One other note with regard to fuse bits — the JTAG interface uses the upper nibble of PORT F, and the JTAGEN fuse bit is programmed by default. Since the JTAG interface supersedes all other functions of the PORT F lines, their other functions are not available while the JTAGEN fuse bit is programmed. Thus, if you are not using a JTAG programmer/debugger, you may wish to unprogram the JTAGEN fuse bit. We leave the JTAGEN fuse bit in its default programmed state since the recipient of the board may only have a JTAG programmer/debugger. If JTAGEN was unprogrammed, those with JTAG programmers would have to find another type of programmer (serial or parallel) in order to change the state of the JTAGEN bit, just so they could begin working on the board.

For a description of all the fuse bits, see pages 286 and 287 of the ATmega128 data sheet available from Atmel's web site (`http://www.atmel.com/`).

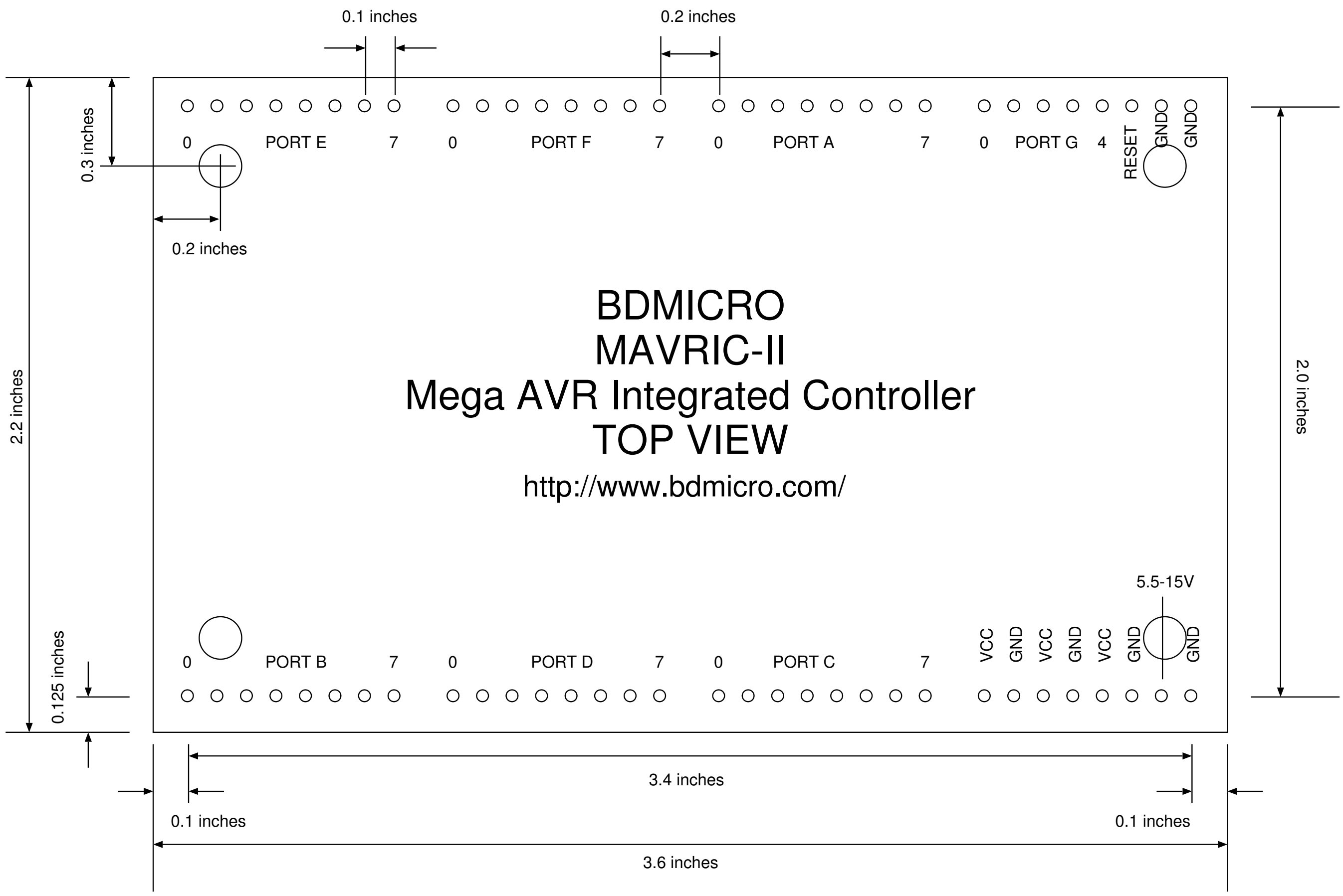
# 8 Memory Expansion

The MAVRIC-II board has 4K of internal static RAM via the ATmega128 processor. While 4K of SRAM is considered large for a microcontroller, some applications are memory intensive and require more. For this purpose, BDMICRO offers the **MAVRIC-II-RE** product, a RAM Expansion card that can be plugged in over the right half of your MAVRIC-II board. The MAVRIC-II-RE expands the memory capacity to 128K of RAM. In order to use the MAVRIC-II-RE, pin headers must be installed in JP1, JP2, JP7, JP8, and JP21. The MAVRIC-II-RE is available on-line from BDMICRO at http://www.bdmicro.com/.

# 9 Schematic Diagram



## 10 Mechanical Drawing



# 11 Soldering the MAVRIC-II Board

If you purchased a bare board or a kit with a bare board, you will be soldering the components on the board. A small diameter solder such as 0.015 inches works well for the small surface mount parts. A temperature controlled iron also makes the job easier. If your vision is not the best (like mine), I've found that a magnifying visor helps a great deal. A short tutorial with some photos is available on the web site at `http://www.bdmicro.com/smt/`. The main points to remember are:

- Use your solder sparingly, it takes only a tiny amount to make a good connection.
- Don't use your soldering iron directly to melt the solder. Doing so causes the solder to bead up due to surface tension on the iron and then release in a big drop across the pins. This results in too much solder and causes bridging, shorting adjacent pins together.

  Instead, use your iron to heat the pin, which then melts the pad underneath the pin. At this point, touch your 0.015 inch solder to the base of the pin. It should melt very quickly, consuming only one or two millimeters of solder, and flow onto the pin and around the pad. Remove the solder quickly.
- Make sure your iron is hot — about 630 degrees F seems to be about right. It needs to be hot enough to melt the solder indirectly, but not so hot that it damages the part.
- While not absolutely required, a good temperature-controlled soldering iron can make the process go much easier than with the standard soldering pencil type irons. If you plan to do much soldering, especially the smaller parts, it would be well worth your investment to acquire one. In fact, good deals can usually be found on on-line auction sites such as E-Bay.

The order in which you solder the parts can make the soldering easier. In general, I solder the parts based on size with the smaller parts first followed by successively larger parts. Thus, I install the surface mount capacitors, resistors, and diodes first, followed by the ICs, the larger surface mount caps and inductor, crystals, and finally the pin headers.

A good technique that I've found for soldering on even the tiniest of surface mount parts is this simple procedure, especially for tiny resistors and capacitors:

- First, prepare one pad by heating it with the tip of your soldering iron and flowing a bit of solder onto the pad. This will make a raised solder surface. Only do one pad at this point, not both.
- Use a pair of fine tweezers to move the part into position.
- Use the tip of your fingernail to hold the part in place, then re-apply heat to the raised solder area. The solder will liquify and the part will sink into position and be held firmly in place by the solder.
- Now move to the unsoldered side and apply a bit of solder to complete the connection.
- Finally, if necessary, return to the first side of the part, and re-apply a tiny bit of solder to ensure a good connection there.

This technique is described on the web site with photos for clarification. See `http://www.bdmicro.com/smt/`.

You can save time by first preparing one pad of each part by applying solder, then attach one side of all the parts, then solder the other side of all the parts, etc, instead of performing all the steps for a part before moving to the next one.

# 12 MAVRIC-II Parts List

| | |
|---|---|
| 1 | ATMEGA128-16AC at U2 (Digikey ATMEGA128-16AC-ND) |
| 1 | MAX3222CDWR at IC3 (Digikey 296-13083-1-ND) |
| 1 | MAX487CSA at IC2 (Digikey MAX487CSA-ND) |
| 1 | LT1129 at IC1 (Digikey LT1129CQ-5-ND) |
| 1 | SN74AHCT1G04DBVR at T2 (Digikey 296-1112-1-ND) |
| 1 | 10K Ohm 0603 SMD at R1 (Digikey P10KGCT-ND) |
| 1 | 120 Ohm 1206 SMD at R4 (Digikey 311-120ECT-ND) |
| 2 | 1.8K Ohm 0603 SMD at R2, R3 (Digikey RR08P1.8KDCT-ND) |
| 1 | 470 Ohm 0805 SMD at R6 (Digikey 311-470ACT-ND) |
| 1 | 10 uH SMD inductor at L1 (Digikey 308-1154-1-ND) |
| 2 | 22 pF ceramic capacitor 0603 SMD at C13, C14 (Digikey PCC220ACVCT-ND) |
| 9 | 0.1 uF ceramic capacitor 0603 SMD at C1, C2, C3, C4, C5, C6, C12, C15, C16 (Digikey PCC1794CT-ND) |
| 4 | 0.56 uF ceramic capacitor 1206 SMD at C19, C20, C21, C22 (Digikey PCC1926CT-ND) |
| 2 | 47 uF electrolytic capacitor FK SMD at C17, C18 (Digikey PCE3417CT-ND) |
| 1 | 32.768 kHz 12.5 pF crystal SMD at Q2 (Digikey XC488CT-ND) |
| 1 | 16 MHz 20 pF crystal SMD at Q1 (Digikey XC694CT-ND) |
| 1 | 14.7456 MHz crystal SMD (optional instead of 16 MHz) at Q1 (Digikey XC692CT-ND) |
| 1 | LED - amber clear 1206 SMD at LED1 (Digikey 160-1166-1-ND) |
| 1 | LL4148 diode 500 mW MINIMELF SMD at D2 (Digikey LL4148DICT-ND) |
| 1 | 4 position .1 inch screw terminal at JP20 (Digikey 277-1275-ND) |

The following part is a 0.1 inch screw terminal connector that can be used for the port headers. They add significant cost, but they make it easy to interface to your board. These are optional.

| | |
|---|---|
| 8 | 8 position .1 inch screw terminal at JP1, JP2, JP3, JP4, JP5, JP6, JP7, JP8 (Digikey 277-1279-ND) |