MinPod™

Controller Board

User Manual

Warranty

New Micros, Inc. warrants its products against defects in materials and workmanship for a period of 90 days. If you discover a defect, New Micros, Inc. will, at its option, repair, replace, or refund the purchase price. Simply call our sales department for an RMA number, write it on the label and return the product with a description of the problem. We will return your product, or its replacement, using the same shipping method used to ship the product to New Micros, Inc. (for instance, if you ship your product via overnight express, we will do the same). This warranty does not apply if the product has been modified or damaged by accident, abuse, or misuse.

Copyrights and Trademarks

Disclaimer of Liability

New Micros, Inc. is not responsible for special, incidental, or consequential damages resulting from any breach of warranty, or under any legal theory, including lost profits, downtime, goodwill, damage to or replacement of equipment or property, and any costs of recovering, reprogramming, or reproducing any data stored in or used with New Micros, Inc. products.

Internet Access

Web site: http://www.newmicros.com

This manual: http://www.newmicros.com/store/product_manual/MinPod™ .zip

Email technical questions: techsupport@newmicros.com

Email sales questions: nmisales@newmicros.com

Also see “Manufacturer” information near the end of this manual.

Internet MinPod™ Discussion List

We maintain the MinPod™ discussion list on our web site. Members can have all questions and answers forwarded to them. It’s a way to discuss MinPod™ issues.

To subscribe to the MinPod™ list, visit the Discussion section of the New Micros, Inc. website.

This manual is valid with the following software and firmware versions
- IsoMax V0.5 or newer.

If you have any questions about what you need to upgrade your product, please contact New Micros, Inc.

Table of Contents

Table Of Figures

Tables

1 GETTING STARTED

Thank you for buying the MinPod™ . We hope you will find the MinPod™ to be the incredibly useful small controller board we intended it to be, and easy to use as possible.

Figure 1 – MinPod™

If you are new to the MinPod™ , we know you will be in a hurry to see it working.

That's okay. We understand.

Let's skip the features and get right to the operation. Once we've got communications, then we can make some lights blink and know for sure that we're in business. Let's make this "Mini board" talk to us!

We'll need PC running a terminal program. Then we'll need a serial cable to connect from the PC to the MinPod™ (which, hopefully, you've already gotten from us). Then we need power, such as from a 6VDC wall transformer (which, hopefully, you've already gotten from us. If we have those connections correct, we will be able to talk to the MinPod™ interactively.

The wall transformer can supply the power to the MinPod™ board, but do not connect it to the board yet. The board can be seen in the figure below, showing the connections of V IN and GND. Connect the serial cable between the J7 connector and the PC. The J7 upper row pinout can also be seen in the figure.

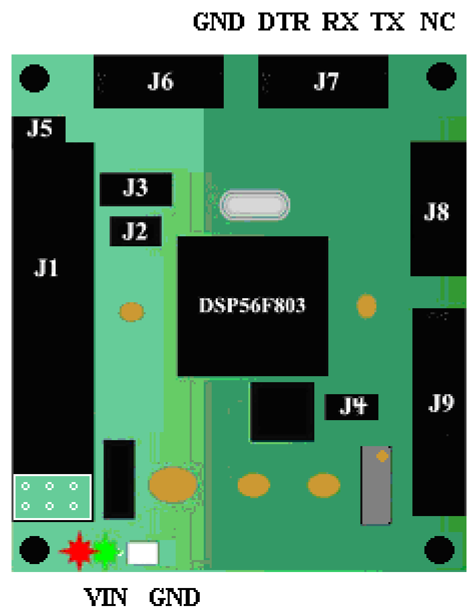


Figure 2 – Power and Serial Connections

Once you have your serial cable and connectors, and wall transformer and connectors ready, follow these steps.

Start with the PC. Install (if not already installed) and run the terminal communications program - NMITerm or HyperTerminal. Set the program for the desired communications channel (COM1, COM2, etc.), and set the communications parameters to 115200, 8N1. Operate the program to get past the opening setups and to the terminal screen, so it is ready to communicate. (If necessary, visit the section on PC Communication if you have trouble understanding how to accomplish any of this.)

Hook the computer end of the serial cable (usually a DB-9 connector, but may be a DB-25, or other, on older PC's) to the PC's communication channel selected in the terminal program.

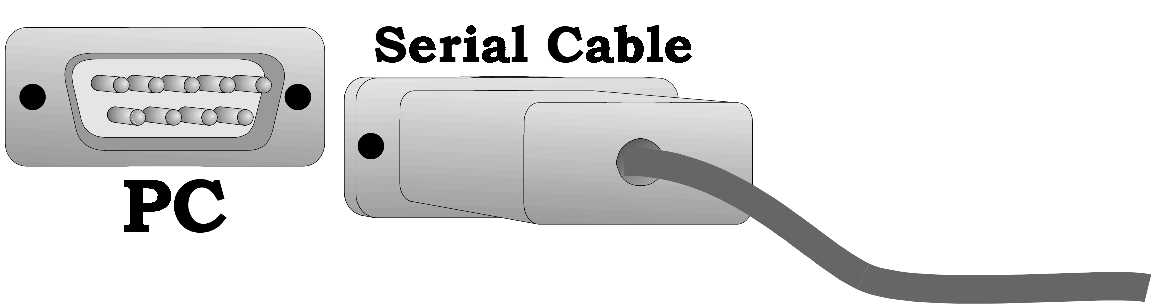


Figure 3 – The Serial Cable

Plug the wall transformer into the wall, but do not plug it into the board yet.

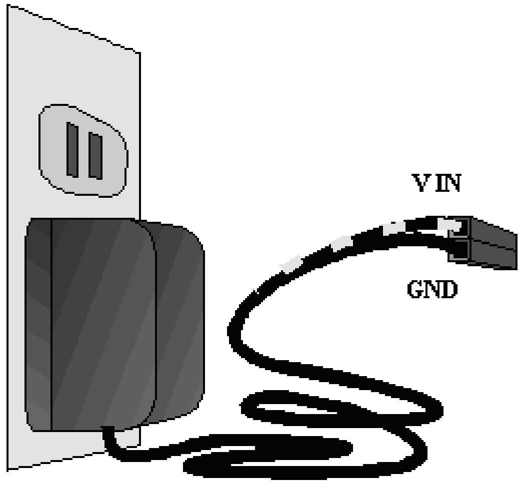


Figure 4 – The Power Cable

1.1 Instructions for IsoMax Users

Now, while watching the LED's plug in the wall transformer connector to the power pins on the MinPod™ board. Be *very* careful not to plug it in reverse polarity here, because there is a chance to kill the board even though the regulator, LM2937 has the reverse voltage protection.

Both the LED's should come on. If the LED's do not light, unplug the power to the MinPod™ quickly.

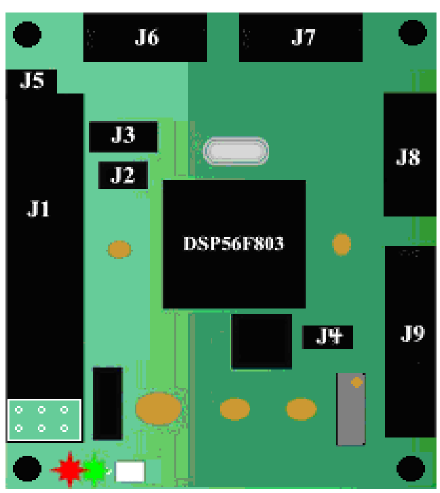


Figure 5 – The LEDs

If the LED glows, then check the screen on the computer. When the power is applied, before any user program installed, the PC terminal program should show "`IsoMax™V0.5`" (or whatever the version currently is, see upgrade policy later at the end of this chapter).

If the LED's don't light, and the screen doesn't show the message, unplug the power to the MinPod™ . Check the power connections, particularly for polarity. (This is the most dangerous error to your board.) If the LED's come on but there is no communication, check the terminal program. Check the serial connections, particularly for a reversal or rotation. Try once more. If you have no success, see the trouble shooting section of this manual and then contact technical support for help, before going further. Do not leave power on the board for more than a few seconds if it does not appear to be operational.

Normally at this point you will see the prompt on the computer screen "`IsoMax™ V0.5`". Odds are you're there. Congratulations! Now let's do something interactive with the MinPod™ .

In the terminal program on the PC, type in, "`WORDS`" (all in "caps" as the language is case sensitive), and then hit "`Enter`". A stream of words in the language should now scroll up the screen. Good, we're making progress. You are now talking interactively with the language in the MinPod™ .

Now let's blink the LED's. Port lines control the LED's. Type:

```
REDLED OFF
```

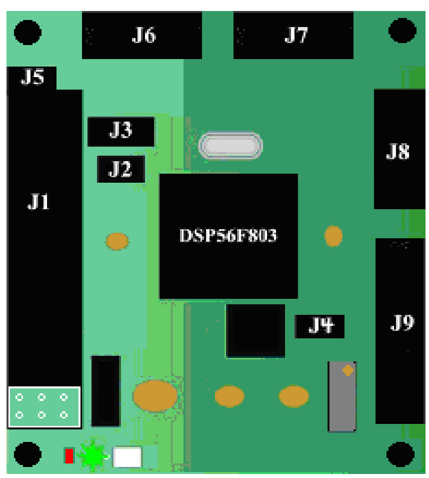


Figure 6 – Red LED off

To turn it back on type:

```
REDLED ON
```

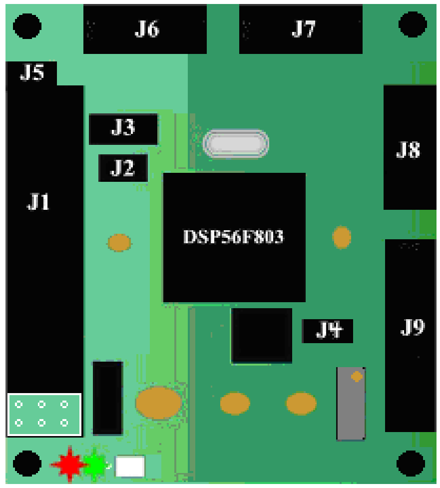


Figure 7 – Red LED on

Now let's use the Green LED. Type:

```
GRNLED OFF
```

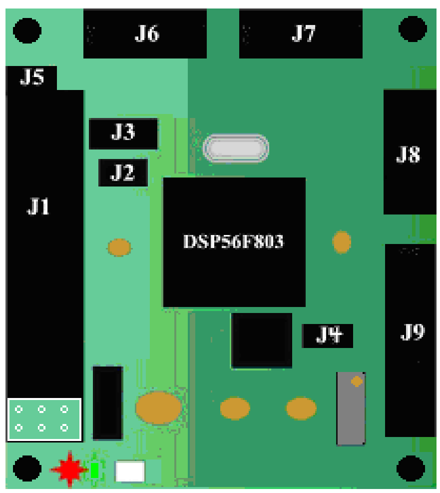


Figure 8 – Green LED off

To turn it back on type:

```
GRNLED ON
```

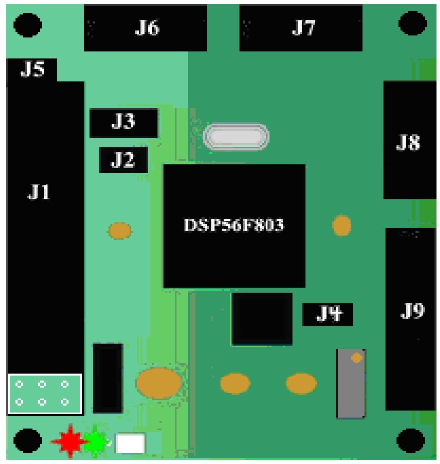


Figure 9 – Green LED on

Now you should have a good feeling because you can tell your MinPod™ is working. It's time for an overview of what your MinPod™ has for features.

First though, a few comments on IsoMax™ revision level. The first port of IsoMax™ to the MinPod™ occurred on May 27, 2002. We called this version V0.1, but it never shipped. While the core language was functional as it then was, we really wanted to add many I/O support words. We added a small number of words to identify the port lines and turn them on and off and shipped the first public release on June 3, 2002. This version was V0.2. Currently we are shipping V0.5 . As we approach a more complete version, eventually we will release V1.0. We want all our original customers to have the benefit of the extensions we add to the language. Any MinPod™ purchased prior to V1.0 release can be returned to the factory (at customer's expense for shipping) and we will upgrade the V0.x release to V1.0 without charge.

1.2 Instructions for Users with Small C and a JTAG Cable

A Small C program will have been flashed to the MinPod™ by NMI prior to shipping. When you connect the power to the board, you should observe the red LED 'on' and the green LED flashing – this proves that the board is working.

1.3 Instructions for Users with Small C without a JTAG Cable

The MinPod™ will contain the Serial Bootloader, and this will have been used by NMI to flash the test program, as detailed in the above section. To ensure correct operation of the board, connect the Serial Cable to your PC (and the MinPod™) and run the desired communications program with the following settings – 115200 baud, 8 bits, no parity, 1 stop bit, Xon/Xoff flow control. When you connect the power to the board, a message will be displayed via the communications program on your PC saying that the Serial Bootloader has started, there will be a 10 second delay, and then the test program will run. You should then observe the red LED 'on' and the green LED flashing – this proves that the board is working

2 QUICK TOUR

Start by comparing your board to the diagram below. Most of the important features on the top board are labeled.

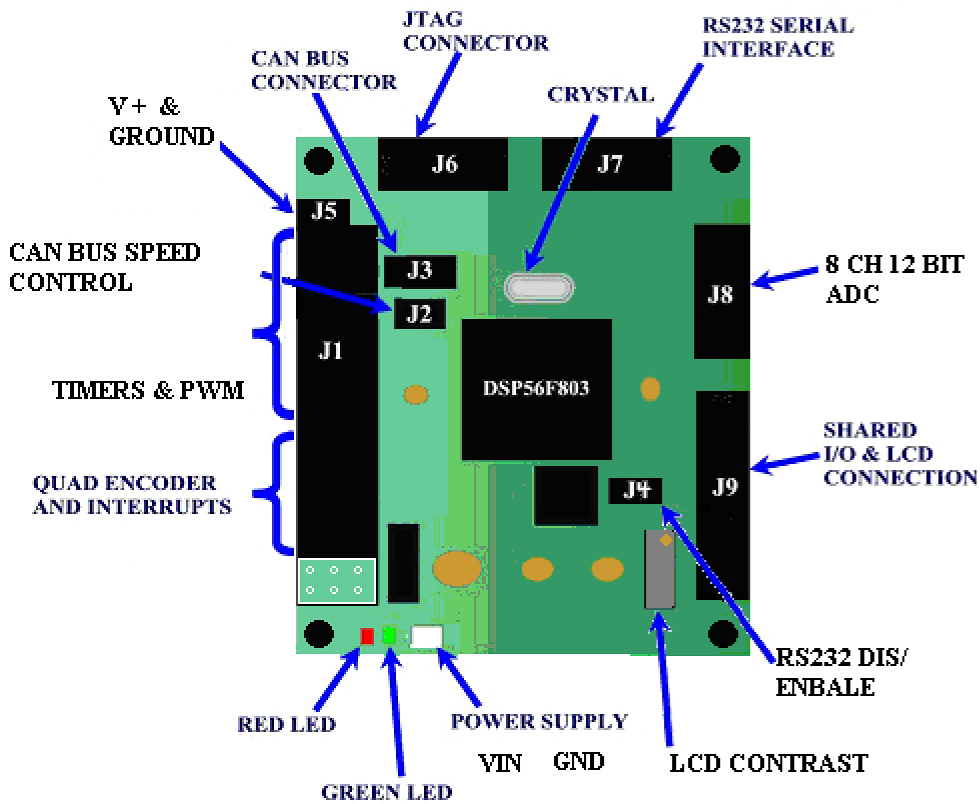


Figure 10 – Board Features

The features most important to you will be the connectors. The following list gives a brief description of each connector and the signals involved.

Table 1 - Connectors

J0	Controller Power Input
J1	PWM, Quad encoder, and Interrupts
J2	CAN BUS Speed
J3	CAN BUS Network Port
J4	RS-232 Dis/Enable
J5	RC-Servo Motor Power Input
J6	JTAG connector
J7	RS-232 Serial Port
J8	A/D Inputs
J9	Shared I/O & LCD & SPI connection & various inputs

The table above explains the type of connection that can be made to each of the connectors. As it can be seen from the table there os no connector assigned for J4. The power reset and ground can be used externally for various applications.

All the connectors are dual or triple row male headers. Connection can be made with female headers with crimped wire inserts, or IDC headers with soldered or cabled wires.

Signals were put on separate connectors where possible, such as with the RS-232, the Can Bus, and PWM connectors. The male headers allow insertion of individually hand-crimped wires in connectors where signals are combined. For instance, R/C Servo motor headers often come in this size connection with a 3x1 header. These can plug directly onto the board side by side on the PWM connector.

The large chip at the center of the board is the CPU (DSP56F803).

The two LED's, Red and Green, are along the bottom of the CPU, and are dedicated to user control.

On the back of the board the components are the voltage regulators. If the total current draw were smaller, we could make a smaller supply, but to be sure every user could get enough power to run at full speed, these larger parts were necessary.

A few smaller chips are also on the board, the RS-232 transceiver and the LED driver, and a handful of resistors and capacitors.

3 CIRCUIT DESCRIPTION

The processor chip contains the vast majority of the circuitry. The remaining support circuitry is described here. The power for the system can be handled several different way, but as the board comes, power will normally be supplied from the VIN pin on board.

3.1 RS-232 Levels Translation

The MAX3221/6/7 converts the 3.3V supply to the voltages necessary to drive the RS-232 interface. Since a typical RS-232 line requires 10 mA of outputs at 10V or more, the MAX3221/6/7 uses about 30 mA from the 3.3V supply. A shutdown is provided, controlled by TD0.

The RS-232 interface allows the processor to be reset by the host computer through manipulation of the ATN line. When the ATN line is low (a logical "1" in RS-232 terms) the processor runs normally. When the ATN line is high (a logical "0" in RS-232 terms) the processor is held in reset.

http://pdfserv.maxim-ic.com/arpdf/MAX3221-MAX3243.pdf

3.2 CAN BUS Levels Translation

A SN65HVD230 buffers the CAN BUS signal.

http://focus.ti.com/lit/ds/symlink/sn65hvd230.pdf

3.3 LED's

A 74HC04 drives the on-board LED's. Each LED has a current limiting resistor to the +3.3V supply.
http://www.fairchildsemi.com/ds/74/74AC05.pdf

3.4 RESET

A S80728HN Low Voltage Detector asserts reset when the voltage is below operating levels. This prevents brown out runaway, and a power-on-reset function.

http://www.seiko-instruments.de/documents/ic_documents/power_e/s807_e.pdf

3.5 POWER SUPPLY

A LM2937 reduces the VIN DC to a regulated 5V. In early versions a 7805C was used. The LM2937 was rated a bit less for current (500 mA Max), but had reverse voltage protection and a low drop out which was more favorable. A drops the 5V to the 3.3V needed for the processor. At full current, 200 mA, these two regulators will get hot. They can provide current to external circuits if care is taken to keep them cool. Each are rated at 1A but will have to have heat sinking added to run there.

http://www.national.com/ds/LM/LM2937.pdf
http://www.national.com/ds/LM/LM3940.pdf

4 TROUBLE SHOOTING

There are no user serviceable parts on the MinPod™ . If connections are made correctly, operation should follow, or there are serious problems on the board. As always, the first thing to check in case of trouble is checking power and ground are present. Measuring these with a voltmeter can save hours of head scratching from overlooking the obvious. After power and ground, signal connections should be checked next. If the serial cable comes loose, on either end, using your PC to debug your program just won't help. Also, if your terminal program has locked up, you can experience some very "quiet" results. Don't overlook these sources of frustrating delays when looking for a problem. They are easy to check, and will make a monkey of you more times than not, if you ignore them.

One of the great advantages of having an interactive language embedded in a processor, is if communications can be established, then program tools can be built to test operations. If the RS-232 channel is not in use in your application, or if it can be optionally assigned to debugging, talking to the board through the language will provide a wealth of debugging information.

The LED's can be wonderful windows to show operation. This takes some planning in design of the program. A clever user will make good use of these little light. Even if the RS-232 channel is in use in your application and not available for debugging, don't overlook the LED's as a way to follow program execution looking for problems.

The MinPod™ is designed so no soldering to the board should be required, and the practice of soldering to the board is not recommended. Instead, all signals are brought to connectors.

So, the best trouble shooting technique would be to unplug the MinPod™ and try to operate it separately with a known good serial cable on power supply.

If the original connections have been tested to assure no out-of-range voltages are present, a second MinPod™ can then be programmed and plugged into the circuit in question. But don't be too anxious to take this step. If the first MinPod™ should be burned out, you really want to be sure you know what caused it, before sacrificing another one in the same circuit.

Finally, for advanced users, the JTAG connection can give trace, single step and memory examination information with the use of special debugging hardware. This level of access is beyond the expected average user of the MinPod™ and will not be addressed in this manual.

5 Online Resources

5.1 *MinPod™ Technical Forum Discussion*

http://www.newmicros.com/discussion/

5.2 *MinPod™ Download Page*

http://www.newmicros.com/store/product_details/download.html

5.3 *Motorola DSP56F803 Users Manual*

http://www.freescale.com/files/dsp/doc/user_guide/DSP56F801-7UM.pdf

5.4 *Motorola DSP56F800 Processor Reference Manual*

http://www.freescale.com/files/dsp/doc/ref_manual/DSP56800FM.pdf

6 CONNECTORS

The MinPod™ has 10 connectors: J0, J1, J2, J3, J4, J5, J6, J7, J8, J9 are below:

J0	Controller Power Input	VIN & GND (6V to 9Vdc @ 200mA or higher)
J1	PWM, Quad Encoder & Interrupts	PWM, Timers, Encoder, IRQx, V+, GND
J2	CAN BUS Speed Jumper	Default Low for Normal operating
J3	CAN BUS Network Port	CANL, GND, CANH
J4	RS-232 Disable/Enable Jumper	I/O controls Ena/Dis Power onboard RS-232
J5	Motor Power Input	GND & V+ (4.8V to 6Vdc)
J6	JTAG connector	JTAG interface connector
J7	RS-232 Serial Port	SIN, SOUT, GND
J8	Analog Inputs	A/D0 – A/D7
J9	Shared LCD & SPI & IO's header And PWM's Fault Inputs	D0-D7/PA0-PA7, PE4-PE6 ISA0-3, FAULT0-3

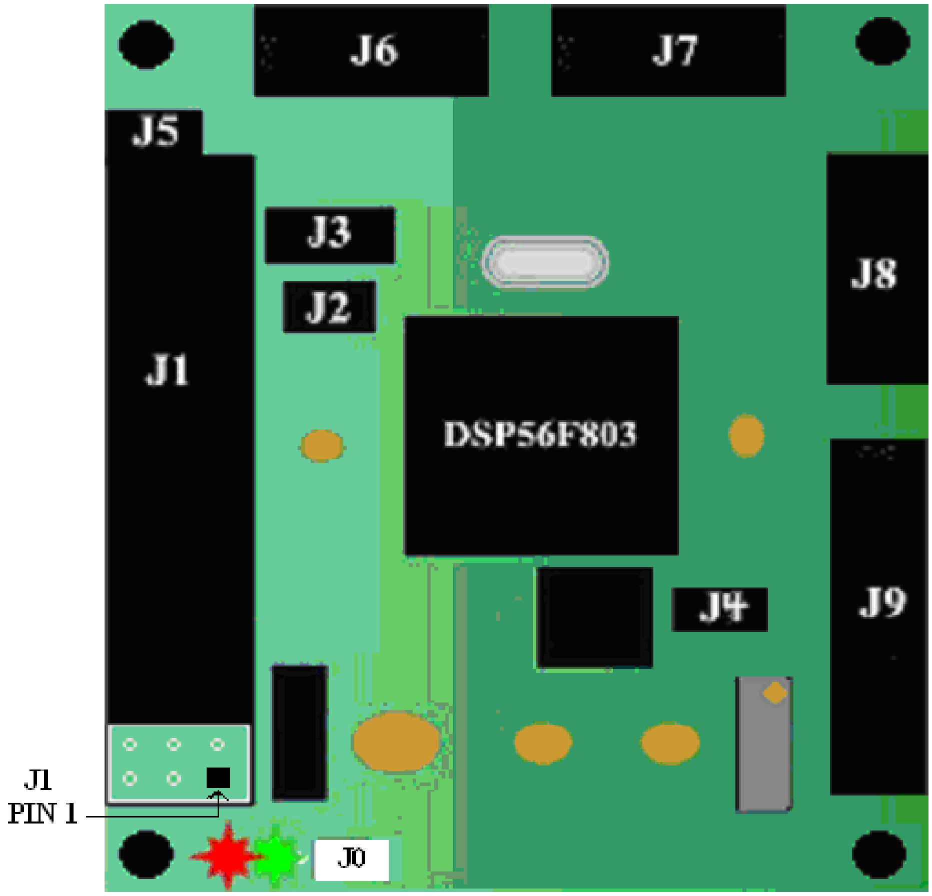


Figure 11 – Connector J1

6.1 J1 PWM, Timers, Quad Encoder & Interrupts

42	GND	41	V+	40	PWMA5
39	GND	38	V+	37	PWMA4
36	GND	35	V+	34	PWMA3
33	GND	32	V+	31	PWMA2
30	GND	29	V+	28	PWMA1
27	GND	26	V+	25	PWMA0
24	GND	23	V+	22	TD2
21	GND	20	V+	19	TD1
18	GND	17	V+	16	TA3/HOME0
15	GND	14	V+	13	TA2/INDEX0
12	GND	11	V+	10	TA1/PHASEB0
9	GND	8	V+	7	TA0/PHASEA0
6	IRQB	5	IRQA	4	GND
3	GND	2	V+	1	+5V

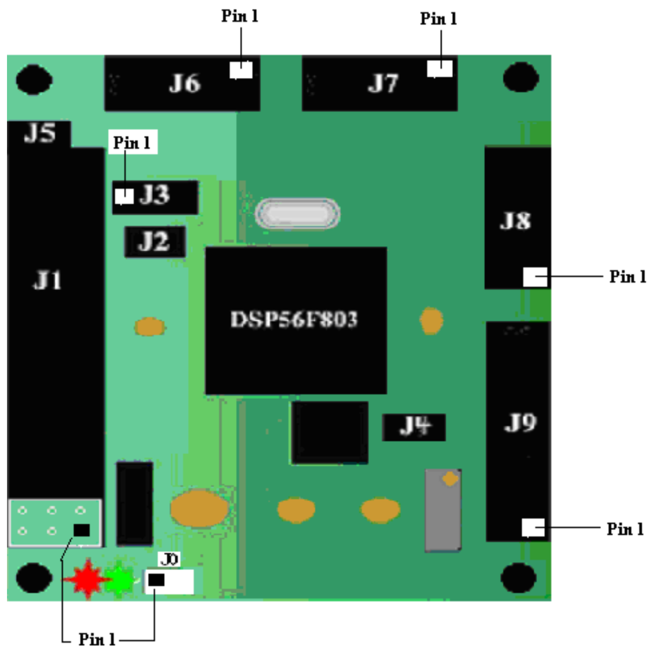


Figure 12 – Connectors J0, J2, J3, J4, J5 and J6

6.2 J0, Controller Power Input

VIN	1	2	GND

6.3 J2, CAN BUS speed control

	1	2	

6.4 J3, CAN BUS

1	CANL	2	GND	3	CANH

6.5 J4, RS232 Disable/Enable

	1	2	

6.6 J5, RC-Servo Power Input

GND	1	2	V+

6.7 J6, JTAG connector

RESET'	TCK	TDO	TDI	+3V
9	7	5	3	1
10	8	6	4	2
TRST'	DE	TMS	GND	GND

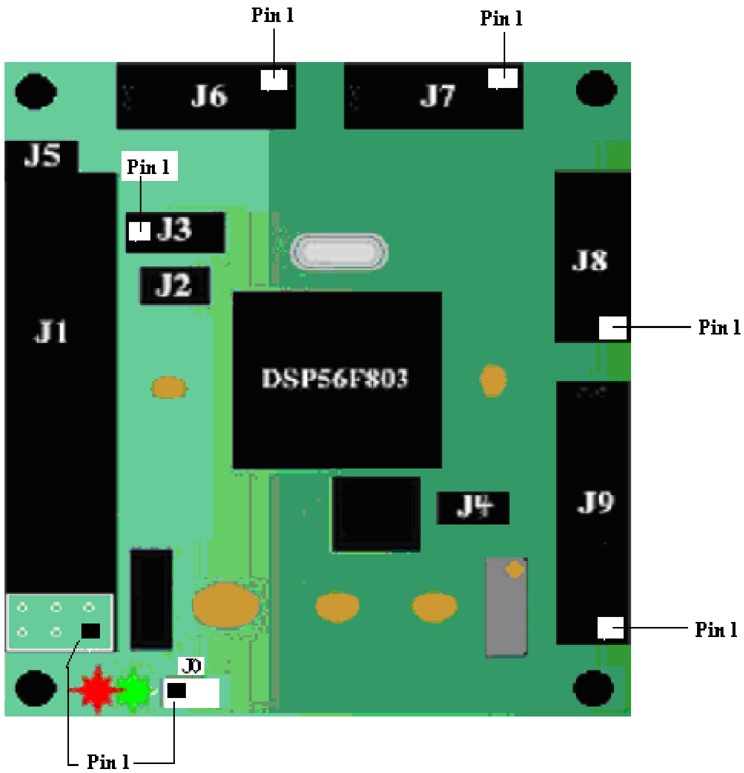


Figure 13 – Connectors J7, J8

6.8 J7, RS232

GND	DTR	RXD	TXD	N.C.
9	7	5	3	1
10	8	6	4	2
N.C.	N.C.	TO 4	TO 6	N.C.

6.9 J8, ADC

ANA7	10	9	ANA6
ANA5	8	7	ANA4
ANA3	6	5	ANA2
ANA1	4	3	ANA0
VSSA	2	1	VREF

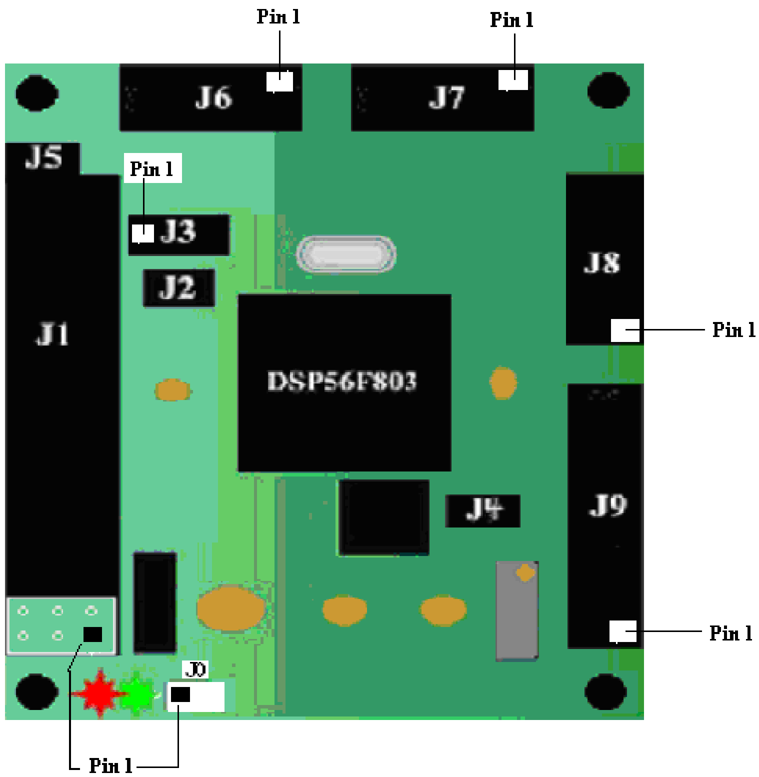


Figure 14 – Connectors J9

6.10 J9, PWM's Current sense & Fault Input connection

FAULT2	22	21	ISA2
FAULT1	20	19	ISA1
FAULT0	18	17	ISA0

6.11 J9, Shared LCD & I/O & SPI connection

E2/PE3	16	15	SS'
D7/PA7	14	13	D6/PA6
D5/PA5	12	11	D4/PA4
D3/PA3	10	9	D2/PA2
D1/PA1	8	7	D0/PA0
E1/PE4/SCK	6	5	RW'/PE5/MOSI
RS/PE6/MISO	4	3	Vo
5V	2	1	GND

7 MANUFACTURER

New Micros, Inc.
1601 Chalk Hill Rd.
Dallas, TX 75212

Tel: (214) 339-2204
Fax: (214) 339-1585

Web site: http://www.newmicros.com

This manual: http://www.newmicros.com/store/product_manual/MinPod™ .pdf

Email technical questions: techsupport@newmicros.com

Email sales questions: nmisales@newmicros.com

8 MECHANICAL

Board size is 1.8" x 2.1"

9 ELECTRICAL

The total draw for the MinPod™ under maximum speed is approximately 150 mA.

Sleeping or slowing the processor can substantially reduce current consumption.

Jumper J4 can be closed to shut down the RS-232 converter, saving about 30 mA, when not used for transmission, if the receiving unit will not sense this as noise.

Jumper J2 can be used to switch between High and Low speed CAN BUS operation based on input from PE3.

Each digital pin is capable of sinking 4 mA and sourcing –4 mA. Each LED draws 1.2 mA when lit.

Table 2 - Absolute Maximum Ratings

Characteristic	Symbol	Min	Max	Unit
Supply voltage	V_{DD}	V_{SS} – 0.3	V_{SS} + 4.0	V
All other input voltages, excluding Analog inputs	V_{IN}	V_{SS} – 0.3	V_{SS} + 5.5V	V
Analog Inputs ANAx, V_{REF}	V_{IN}	V_{SS} – 0.3	V_{DDA} + 0.3V	V
Current drain per pin excluding V_{DD}, V_{SS}, PWM outputs,	I	—	10	mA

TCS, V_{PP}, V_{DDA}, V_{SSA}				
Current drain per pin for PWM outputs	I	—	20	mA
Junction temperature	T_J	—	150	°C
Storage temperature range	T_{STG}	-55	150	°C

Table 3 - Recommended Operating Conditions

Characteristic	Symbol	Min	Max	Unit
Supply voltage	V_{DD}	3.0	3.6	V
Ambient operating temperature	T_A	-40	85	°C

Table 4 - DC Electrical Characteristics

Operating Conditions: $V_{SS} = V_{SSA} = 0$ V, $V_{DD} = V_{DDA} = 3.0–3.6$ V, $T_A = –40°$ to +85°C, $C_L \leq 50$ pF, $f_{op} = 80$ MHz

Characteristic	Symbol	Min	Typ	Max	Unit
Input high voltage	V_{IH}	2.0	—	5.5	V
Input low voltage	V_{IL}	-0.3	—	0.8	V
Input current low (pullups/pulldowns disabled)	I_{IL}	-1	—	1	μA
Input current high (pullups/pulldowns disabled)	I_{IH}	-1	—	1	μA
Typical pullup or pulldown resistance	R_{PU}, R_{PD}	—	30	—	KΩ
Input/output tri-state current	low I_{OZL}	-10	—	10	μA
Input/output tri-state current	low I_{OZH}	-10	—	10	μA
Output High Voltage (at IOH)	V_{OH}	V_{DD} – 0.7	—	—	V
Output Low Voltage (at IOL)	V_{OL}	—	—	0.4	V
Output High Current	I_{OH}	—	—	-4	mA
Output Low Current	I_{OL}	—	—	4	mA
Input capacitance	C_{IN}	—	8	—	pF
Output capacitance	C_{OUT}	—	12	—	pF
PWM pin output source current 1	I_{OHP}	—	—	-10	mA
PWM pin output sink current 2	I_{OLP}	—	—	16	mA
Total supply current	I_{DDT} 3				
Run 4		—	126	162	mA
Wait 5		—	72	98	mA
Stop		—	60	84	mA
Low Voltage Interrupt 6	V_{EI}	2.4	2.7	2.9	V
Power on Reset 7	V_{POR}	—	1.7	2.0	V

1. PWM pin output source current measured with 50% duty cycle.

2. PWM pin output sink current measured with 50% duty cycle.

3. $I_{DDT} = I_{DD} + I_{DDA}$ (Total supply current for $V_{DD} + V_{DDA}$)

4. Run (operating) I_{DD} measured using 8MHz clock source. All inputs 0.2V from rail; outputs unloaded. All ports configured as inputs; measured with all modules enabled.

5. Wait I_{DD} measured using external square wave clock source (f_{osc} = 8 MHz) into XTAL; all inputs 0.2V from rail; no DC loads; less than 50 pF on all outputs. C_L = 20 pF on EXTAL; all ports configured as inputs; EXTAL capacitance linearly affects wait I_{DD}; measured with PLL enabled.

6. Low voltage interrupt monitors the V_{DDA} supply. When V_{DDA} drops below V_{EI} value, an interrupt is generated. For correct operation, set $V_{DDA}=V_{DD}$. Functionality of the device is guaranteed under transient conditions when $V_{DDA}>V_{EI}$.

7. Power-on reset occurs whenever the internally regulated 2.5V digital supply drops below V_{POR}. While power is

ramping up, this signal remains active for as long as the internal 2.5V supply is below 1.5V no matter how long the ramp up rate is. The internally regulated voltage is typically 100 mV less than V_{DD} during ramp up until 2.5V is reached, at which time it self regulates.

10 Embedded Software Development

10.1 Overview

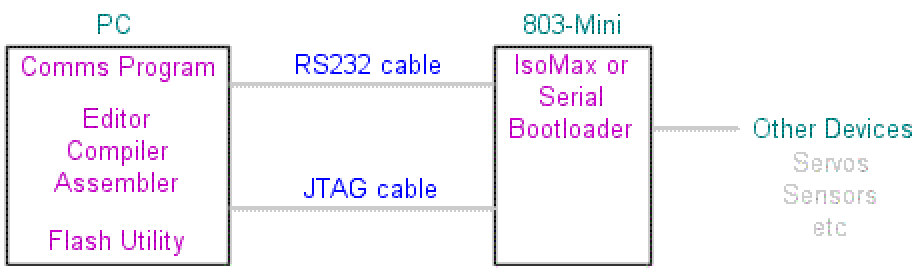


Figure 14 – Embedded Software Development Overview

This diagram shows all of the components available for embedded software development. Note that not all of the components will be used, depending upon the scheme adopted by the developer. These schemes – and the required components – are detailed below.

10.2 IsoMax

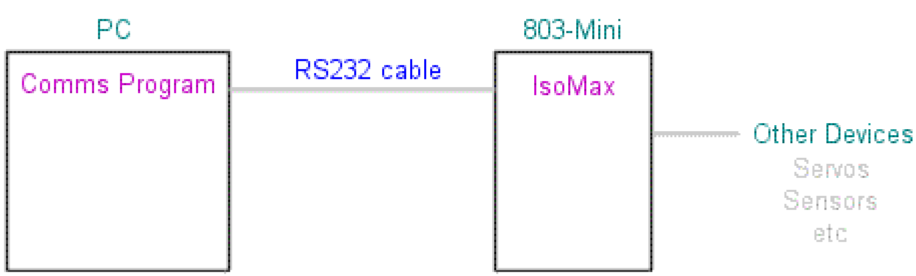


Figure 15 – Software Development using IsoMax

In this configuration, the PC user runs a program which communicates with IsoMax via the RS-232 cable. This allows interactive software development and testing. Details of IsoMax can be found on NMI's website.

10.3 Compiler & Assembler

Using this method, software is developed on the PC and transferred to the MinPod™ using either the RS-232 (serial) cable or a JTAG cable. Note that the diagrams, below, show the components *required* for each scheme. Additional components are optional.

10.3.1 With JTAG

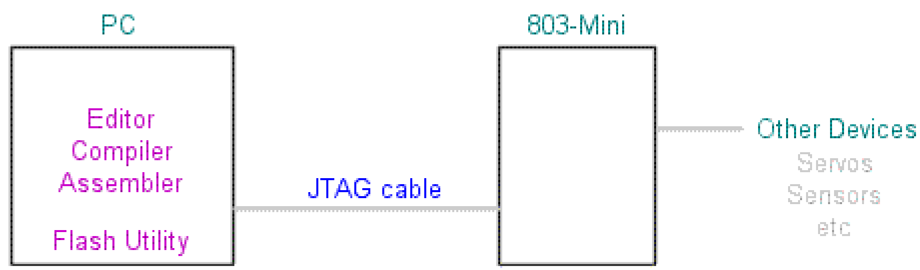


Figure 16 – Software Development using JTAG

In this configuration, an editor is used to write a program on the PC, which is then compiled, assembled, and transferred to the MinPod™ via the JTAG cable using a Flash Utility on the PC. The RS-232 cable and communications program (not shown) are optional.

The JTAG cable plugs into the PC's parallel port, and J6 on the MinPod™ board. However, the Windows Operating System does not allow a 'normal' user program – like the Flash Utility - to directly control the computer's hardware (i.e. the parallel port), so a utility must be used which permits this, such as UserPort.

10.3.2 With Serial Bootloader

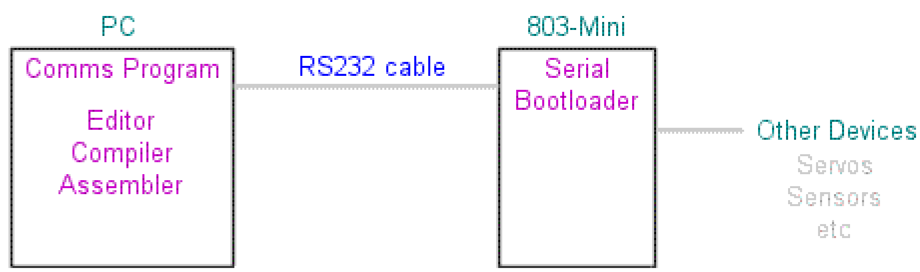


Figure 17 – Software Development using Serial Bootloader

In this configuration, an editor is used to write a program on the PC, which is then compiled, assembled, and transferred to the MinPod™ via the RS-232 cable using the Communications Program on the PC and the Serial Bootloader software on the MinPod™ . The JTAG cable and Flash Utility (not shown) are optional.

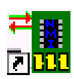
11 PC Communication

A variety of programs are available which allow a PC to communicate directly with the MinPod™ . These programs are sometimes referred to as 'Communications Programs', 'Comms programs' or 'Terminal Programs' (because some emulate Computer Terminals). These programs run on the PC and are used in conjunction with an RS-232 cable, also known as a Serial Cable. One end of this cable plugs into the PC's COM port, and the other end is for the MinPod™ 's RS-232 connector, J7.

Note: The settings given in the following sections are IsoMax and JTAG users only – Serial Bootloader users should refer to the relevant sections in this manual for the required settings.

11.1 NMITerm

Provided Windows terminal program from New Micros, Inc. Usually provided in a ZIP. Un ZIP in a subdirectory, such as C:\NMITerm. To start the program: click, or double click, the program icon.



NMITerm.LNK

NMITerm is a simple Windows-based communications package designed for program development on serial port based embedded controllers. It runs under Windows.

NMITerm provides:

1. Support for COM1 through COM16.
2. Baud rates from 110 through 256000.
3. Control over RTS and DTR lines.
4. Capture files, which record all terminal activity to disk.
5. Scroll-back buffer, editable and savable as a file.
6. On-line Programmer's Editor.
7. File downloader.
8. Programmable function keys.

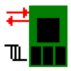
Quick start commands:

1. Baud: default 9600. This needs to be changed to 115200
2. DTR On/Off : ALT+T
3. Download: ALT+D

For further information use the F1 Help screen.
This program can be downloaded from:
http://www.newmicros.com/download/NMITerm.zip

11.2 MaxTerm

Provided DOS terminal program from New Micros, Inc. Usually provided in a ZIP. Un ZIP in a subdirectory, such as C:\MAXTERM. To start the program: click, or double click, the program icon.



Maxterm.ico

MaxTerm is a simple DOS-based communications package designed for program development on serial port based embedded controllers. It can run under stand-alone DOS or in a DOS session under Windows.

MaxTerm provides:

```
1. Support for COM1 & COM2.
2. Baud rates from 300 through 38400.
3. Control over RTS and DTR lines.
4. Capture files, which record all terminal activity to disk.
5. 32K scroll-back buffer, editable and savable as a file.
6. On-line Interactive Programmer's Editor (OPIE).
7. File downloader.
8. Programmable function keys.
9. Received character monitor, which displays all data in HEX.
```

Quick start commands:

```
4. Set comport: ALT+1 or ALT+2 It does not support com3 & 4.
5. Baud: default 9600
6. DTR On/Off : ALT+T
7. Download: ALT+D
8. PACING: ALT+P (IsoMax default decimal 10)
```

For further information use the Help screen (ALT-H) or the program documentation.

```
               MAXTERM Help
alt-B Change baud rate                alt-M Character monitor mode
alt-C Open (or close) capture file    alt-O Toggle sounds
alt-D Download a file (all text)      alt-P Change line pace char
alt-E Edit a file (Split screen)      alt-R Toggle RTS
alt-F Edit function keys              alt-S Unsplit the screen
alt-H Help                            alt-T Toggle DTR
alt-I Program Information             alt-U Change colors
alt-K Toggle redefinition catcher     alt-W Wipe the screen
alt-L Open scrollback log             alt-X Exit
alt-1 (2 3 4) Select Com port         alt-Z Download a file (no fat)
f1-f10 Programmable function keys     f12   Re-enter OPIE
```

```
Status line mode indicators: r = rts, d = dtr, L = log file, S = sounds,
K = redefinition, P = line pacing active
```

11.3 HyperTerminal

Usually provided in Programs/Accessories/Communications/HyperTerminal. If not present, it can be loaded from the Windows installation disk. Use “Add/Remove Software” feature in Settings/Control Panel, choose Windows Setup, choose Communications, click on Hyperterm, then Okay and Okay. Follow any instructions to add additional features to windows.

Hypertrm.exe

C:\Program Files\Accessories\HyperTerminal

Run HyperTerminal, select an icon that pleases you and give the new connection a name, such as Mini803. Now in the “Connect To” dialog box, in the bottom “Connect Using” line, select the communications port you wish to use, with Direct Com1, Direct Com2, Direct Com3, Direct Com4 as appropriate, then Okay. In the COMx Dialog box which follows set up the port as follows: Bits per second: 115,200. Data bits: 8, Parity: None, Flow Control: None, then Okay.

The ATN signal must be unconnected when using this program. There is no option to remotely set and reset the board using the DTR line with this program.

12 REFERENCE

12.1 Decimal / Octal / Hex / ASCII Chart

DEC	OCT	HEX	Character	Control Action
0	0	0	NUL	Null character
1	1	1	SOH	Start of heading, = console interrupt
2	2	2	STX	Start of text
3	3	3	ETX	End of text
4	4	4	EOT	End of transmission, not the same as ETB
5	5	5	ENQ	Enquiry, goes with ACK; old HP flow control
6	6	6	ACK	Acknowledge, clears ENQ logon hand
7	7	7	BEL	Bell, rings the bell...
8	10	8	BS	Backspace, works on HP terminals/computers
9	11	9	HT	Horizontal tab, move to next tab stop
10	12	a	LF	Line Feed
11	13	b	VT	Vertical tab
12	14	c	FF	Form Feed, page eject
13	15	d	CR	Carriage Return
14	16	e	SO	Shift Out, alternate character set
15	17	f	SI	Shift In, resume defaultn character set
16	20	10	DLE	Data link escape
17	21	11	DC1	XON, with XOFF to pause listings; ":okay to send".
18	22	12	DC2	Device control 2, block-mode flow control
19	23	13	DC3	XOFF, with XON is TERM=18 flow control
20	24	14	DC4	Device control 4
21	25	15	NAK	Negative acknowledge
22	26	16	SYN	Synchronous idle
23	27	17	ETB	End transmission block, not the same as EOT
24	30	17	CAN	Cancel line, MPE echoes !!!
25	31	19	EM	End of medium, Control-Y interrupt
26	32	1a	SUB	Substitute
27	33	1b	ESC	Escape, next character is not echoed
28	34	1c	FS	File separator
29	35	1d	GS	Group separator
30	36	1e	RS	Record separator, block-mode terminator
31	37	1f	US	Unit separator

DEC	OCT	HEX	Character	Description
32	40	20	SP	Space
33	41	21	!	Exclamation mark
34	42	22	"	Quotation mark
35	43	23	#	Cross hatch (number sign)
36	44	24	$	Dollar sign
37	45	25	%	Percent sign
38	46	26	&	Ampersand
39	47	27	`	Closing single quote (apostrophe)
40	50	28	(	Opening parentheses
41	51	29	)	Closing parentheses

42	52	2a	*	Asterisk (star, multiply)
43	53	2b	+	Plus
44	54	2c	,	Comma
45	55	2d	-	Hyphen, dash, minus
46	56	2e	.	Period
47	57	2f	/	Slant (forward slash, divide)
48	60	30	0	Zero
49	61	31	1	One
50	62	32	2	Two
51	63	33	3	Three
52	64	34	4	Four
53	65	35	5	Five
54	66	36	6	Six
55	67	37	7	Seven
56	70	38	8	Eight
57	71	39	9	Nine
58	72	3a	:	Colon
59	73	3b	;	Semicolon
60	74	3c	<	Less than sign
61	75	3d	=	Equals sign
62	76	3e	>	Greater than sign
63	77	3f	?	Question mark
64	100	40	@	At-sign
65	101	41	A	Uppercase A
66	102	42	B	Uppercase B
67	103	43	C	Uppercase C
68	104	44	D	Uppercase D
69	105	45	E	Uppercase E
70	106	46	F	Uppercase F
71	107	47	G	Uppercase G
72	110	48	H	Uppercase H
73	111	49	I	Uppercase I
74	112	4a	J	Uppercase J
75	113	4b	K	Uppercase K
76	114	4c	L	Uppercase L
77	115	4d	M	Uppercase M
78	116	4e	N	Uppercase N
79	117	4f	O	Uppercase O
80	120	50	P	Uppercase P
81	121	51	Q	Uppercase Q
82	122	52	R	Uppercase R
83	123	53	S	Uppercase S
84	124	54	T	Uppercase T
85	125	55	U	Uppercase U
86	126	56	V	Uppercase V
87	127	57	W	Uppercase W
88	130	58	X	Uppercase X
89	131	59	Y	Uppercase Y
90	132	5a	Z	Uppercase Z
91	133	5b	[	Opening square bracket
92	134	5c	\	Reverse slant (Backslash)
93	135	5d	]	Closing square bracket
94	136	5e	^	Caret (Circumflex)

95	137	5f	_	Underscore
96	140	60	`	Opening single quote
97	141	61	a	Lowercase a
98	142	62	b	Lowercase b
99	143	63	c	Lowercase c
100	144	64	d	Lowercase d
101	145	65	e	Lowercase e
102	146	66	f	Lowercase f
103	147	67	g	Lowercase g
104	150	68	h	Lowercase h
105	151	69	i	Lowercase i
106	152	6a	j	Lowercase j
107	153	6b	k	Lowercase k
108	154	6c	l	Lowercase l
109	155	6d	m	Lowercase m
110	156	6e	n	Lowercase n
111	157	6f	o	Lowercase o
112	160	70	p	Lowercase p
113	161	71	q	Lowercase q
114	162	72	r	Lowercase r
115	163	73	s	Lowercase s
116	164	74	t	Lowercase t
117	165	75	u	Lowercase u
118	166	76	v	Lowercase v
119	167	77	w	Lowercase w
120	170	78	x	Lowercase x
121	171	79	y	Lowercase y
122	172	7a	z	Lowercase z
123	173	7b	{	Opening curly brace
124	174	7c	|	Vertical line
125	175	7d	}	Closing curly brace
126	176	7e	~	Tilde (approximate)
127	177	7f	DEL	Delete (rubout), cross-hatch box

12.2 Simple ASCII Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_'
6	`	a	b	c	d	e	f	g	h	I	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

13 GLOSSARY

.1" double and triple row connectors
24-pin socket
74AH04
115,200 8N1
A/D
adapter
ASCII
CAN BUS
Caps
carrier board
computing and control function
communications channel
communications settings
COM1
COM2
COM3
COM4
controller
controller interface board
dedicated computer
deeply embedded
double male right angle connector
double sided sticky tape
embedded
embedded tasks
female
hand-crimped wires
headers
high-density connectors
High-Level-Language
HyperTerminal
IDC headers and ribbon cable
interactive
IsoMax™
MinPod™
language
Levels Translation
LED
LM3940
LM78L05
Low Voltage Detector
male

14 Articles and Suggested Reading

Mealy, G. H. State machine pioneer, wrote “A Method for Synthesizing Sequential Circuits,” Bell System Tech. J. vol 34, pp. 1045 –1079, September 1955

Moore, E. F. State machine pioneer, wrote “Gedanken-experiments on Sequential Machines,” pp 129 – 153, Automata Studies, Annals of Mathematical Studies, no. 34, Princeton University Press, Princeton, N. J., 1956