U1 — TMS320C54V90

Left side pins:
- 66 INT2
- 64 INT0
- 36 BFSRX2
- 43 BFSR0
- 53 BFSX0
- 65 INT1
- 47 BDR1
- 87 INT3
- 71 RX
- 38 TX
- 48 BCLKX0
- 41 BCLKR0
- 44 BFSR1
- 54 BFSX1
- 49 BCLKX1
- 42 BCLKR1
- 74 C1A5V
- 96 X1
- 97 X2
- 35 BDR2

Right side pins:
- CLKMD3 79
- CLKMD2 78
- CLKMD1 77
- BIO 31
- EMU0 83
- EMU1 84
- NMI 63
- READY 19
- HOLD 30
- C1A 73
- HPI16 80
- MP/MC 32
- RS 98
- DTR 58
- RTS 69
- CTS 81
- DSR 95
- DCD 120
- RI 124

+3.3V

R42 4.7K
R31 4.7K
R40 4.7K
R46 4.7K
R45 4.7K

Y1 4.7456 mhz
C38 22pf
C39 22pf

+5V
C3 .22uf

MODRX
MODTX
MODRI
MODDCD
MODDSR
MODCTS
MODDRTS
MODDTR
MODRES

R27 10

Z4 5.6v
D3
C1 150pf
D4
C6 .1
Z5 5.6v

U2 — PCW308W

Pins:
- C1B 1
- VREG 9
- VREG2 10
- REF 11
- QE2 1
- FILT2 12
- FILT 15
- RX 14
- QB 7
- QE 8
- RNG1 6
- RNG1 5
- DCT 2
- REXT2 12
- REXT 13
- GND 3

R28 10

R7 5.36K
R15 5.36K
R8 5.36K
R24 150
R18 2.2K
R19 2.2K
R16 5.36K
R17 5.36K

R11 9.3K
R13 215
C14 .68uf
R2 402
C16 .1

C13 1uf
C12 1uf
C5 .1uf
C22 .0018
R12 78.7

Q4
Q1
R5 100K
Q2
Z1 43V
Q3
C20 .01
R6 120K
C4 150pf
C9 .1

R10 56K
C8 560pf
C19 .0039
C18 .0039
R9 56K
C7 560pf
R26 10M
R25 10M
D2
D1
C25 1000pf
C24 1000pf
L1 330uh Fbead
L2 330uh Fbead
FB0 Fbead
FB1 Fbead
RV1
MOD6

U50 — TPS7014BPWP

- VIN1 2
- VIN1 3
- VIN2 9
- VIN2 10
- /EN 6
- SEQ 7
- VOUT1 18
- VOUT1 19
- VSENSE1 17
- PG1 16
- /WR2 5
- /RESET 1
- /WR1 4
- VSENSE2 14
- VOUT2 12
- VOUT2 13

+5V
+3.3V
+1.5V

R50 249K
C36 22uf
C34 .01
C35 .01
C51 .1
C50 .01
C40 2.2uf
C33 .01
C41 .01

Set-up Instructions

RS-232 Cable

Phone Jack

ICD-S

PC1

Phone Line

ICD Cable

CPC Cable

A
B

C
0
C
1

D

E

AN0

A A A A A A G
N N N N N N N
1 2 3 4 5 6 7 D

PICNET

PC2

12V Power Supply
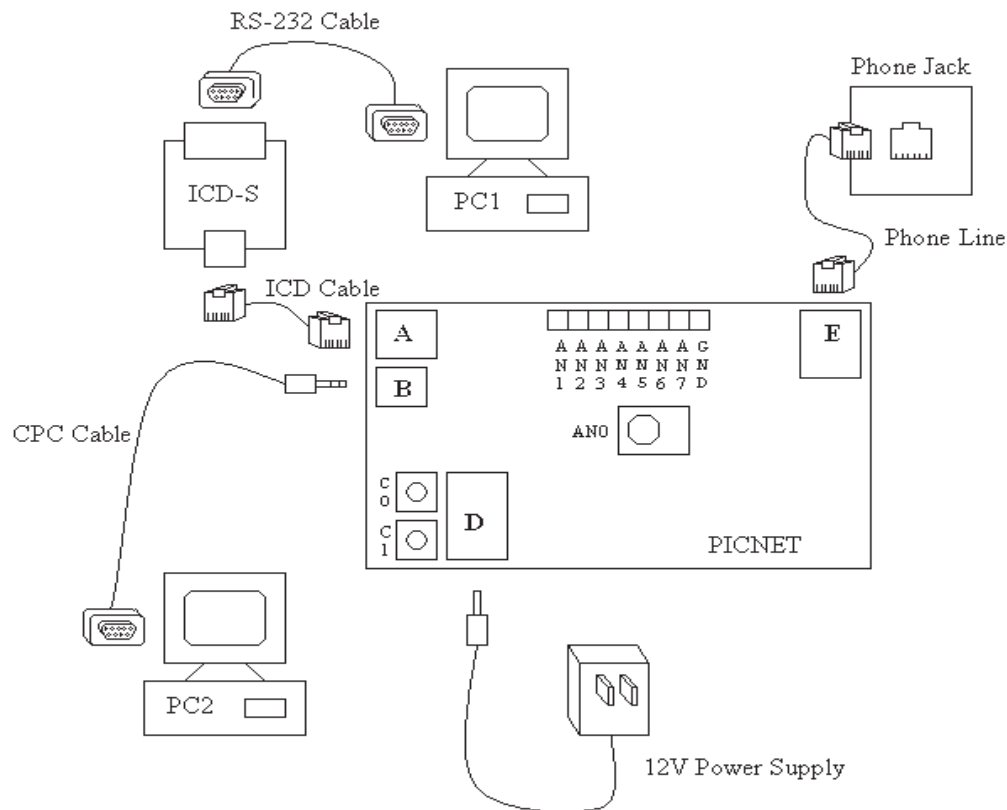
A---ICD Connector
Connects the prototype board to an ICD (In Circuit Debugger) unit for debugging or programming. CCS sells two ICD units, the ICD-S (serial connection) and ICD-U (USB connection). For more documentation on programming or debugging a PIC with an ICD, please refer to your ICD manual. PC1 controls the ICD for debugging and programming.

B---CPC Connector
Connects a CCS CPC cable to the board. The CCS CPC cable is essentially an RS232 serial cable, but with the connecter on the prototype board replaced with a 3 pin connector to save space on the board. Connect the CPC cable to the board and a PC to have an RS232 link between the board and a PC. PC2 communicates to PIC16F877 on the board to receive and transmit data.

PC2 and PC1---can be the same computer if it has two serial ports. PC1 communicates with the ICD-S and PC2 communicates with the internet board.

C0---Switch Button
This button connects to PINC0 of the PIC16F877, and is active low. CCS refers to this as BUTTON2 in their example programs.

C1---Switch Button
This button connects to PINC1 of the PIC16F877, and is active low. CCS refers to this as BUTTON1 in their example programs.

D---12V Supply
A connector for a 12V DC power supply; CCS provides a 300mA 12V DC power supply to connect here.

AN0 Potentiometer connected to AN0
CCS provides a potentiometer connected to AN0 on the PIC16F877's ADC.

AN1-AN7 Header to PIC16F877's ADC
With this header you can connect external signal sources to the PIC16F877's analog to digital converter channels. AN0 is already connected to a potentiometer on the board.

GND Ground
Easy access to the board's common ground.

E Phone Jack
Connects the board to a phone line. The board's modem uses the phone line to dial your ISP and make a PPP connection to the Internet.

Program Examples:

CCS provides the following examples:
 Ex_websv.c -------How to be a webserver
 Ex_dnslookup.c --How to resolve DNS lookups
 EX_email.c -------How to send an E-mail over the internet

These examples are found in the examples directory of your CCS C Compiler installation. The examples share a few common features. First, in order to receive and send data to and from the board using a PC you must connect a CPC cable to connector B. The other side of the CPC cable (the 9pin RS232 connector) connects to the COM port of your PC. On the PC use a serial program at a baud rate of 9600 and no parity.

To change the board's ISP settings (phone number, username, password), press the SPACE bar on the PC running the serial program before the board is connected to the Internet. Since these settings are saved into an EEPROM they will be saved even after a power down.

Pressing BUTTON1 when the board is not connected to the Internet will cause the board to connect to the Internet with the saved settings. After connection the board will display the connected IP address on the PC running the serial program. Pressing BUTTON1 when the board is connected will cause the board to disconnect from the Internet, and hang up the phone.

Ex_email.c
This example shows how to use CCS's provided SMTP driver to send static and dynamic e-mails. Once theboard is connected to the Internet you can send e-mail in two ways. The first method is by pressing BUTTON2, which causes the board to send an e-mail with dynamic information to a pre-determined recepient. The second method is by pressing SPACE bar, which will allow you to send any e-mail you wish to any person.

Ex_dnslookup.c
This is an example of how DNS are done using CCS's provided DNS lookup library. You enter in the hostname you want to lookup (example, www.ccsinfo.com) and the program returns the IP address (in the example of www.ccsinfo.com, 66.36.102.48).
Once connected to the Internet, press the SPACE bar to query a hostname for DNS lookup.

Ex_websv.c
This is an example of how to be a simple webserver, listening to port 80 and sending out a dynamic web page any time a client opens port 80. This webserver is so simple, however, it does not actually see if the client made an HTTP request or see what page the client requested. It does shows the basics of opening port 80 in listen mode and answering requests when they are made.

This example has another added feature in that it listens to port 6666 for requests. Clients can telnet into 6666 of your PIC webserver and change one of the messages that is displayed on the dynamic webpage. Note: You must use the RAW telnet protocol as the s7600 on the PICNET board does not support the telnet protocol.