CAN Network
Development/Training Modules

Instruction Manual

CAN-NET Education Board Kit
(PN 905181)
CAN-NET Education Board
(PN 905185)
CAN-NET Node Board
(PN 905186)
CAN-NET I/O Board
(PN 905187)
CAN Manager Program
(PN 905193)

Diversified Engineering

Expanding Your Engineering Options

283 Indian River Road, Orange, CT 06477 USA
Tel 203-799-7875 Fax 203-799-7892
www.diversifiedengineering.net

Table of Contents

CAN-Net Education Board Kit 2
Overview 2
Intended Audience 2

The CAN Education Board 2
Overview 2
Power Sources 3
Microcontroller 3
Oscillator 3
E^2 Memory 4
Push Buttons 4
Potentiometers (RP2 – RP3) 5
LED Indicators (LED0 – LED7) 5
LCD Display (LCD1) 6
Bread-Boarding Area 7
Connecting Terminal Blocks 7
Source Code Utility 7
Serial Port 7
In-Circuit Debugger (ICD) Interface 8

The Node and I/O Boards 8
The Node Board (905186) 8
I/O board (905187) 8
Interconnection and Initial Power-up 9

Software System Overview 10
CAN ID Structure 10
Software 10
CAN-NET Education Board 11
Node Board Software 12
CAN Manager Program PC Interface 14

Appendix
A Layout 15
B LCD Character Set 16
C LCD Instruction Set 17
D Additional CAN Expansion Bds 18

I CAN-Net Education Board Kit

1 Overview

1.1 CAN-NET (Controller Area Network) is a two-wire serial communication protocol that supports distributed real time control.

The ***CAN-NET Education Board Kit*** (PN 905181) provides a means of communication between the CAN-Net Education Board and one remote node, allowing reading and writing through a CAN bus with the provided harness. The CAN Education Board Kit consists of:

- CAN-Net Education Board – PN 905185
- I/O Node Kit – PN 905192,which includes CAN-net node board, CAN-net I/O board and harness

1.2 The CAN Education board can control up to four nodes at one time.
Refer to Appendix D for other options.

2 Intended Audience

2.1 The CAN-NET Education Board Kit is designed for students, engineers and experimenters with a basic knowledge of PIC Microprocessors who want to learn to use CAN with the Microchip MCP2510 series of CAN controllers.

II The CAN Education Board

1 Overview

1.2 The CAN Education Board is designed around the PIC16F874 Microcontroller and MCP2510 CAN Controller manufactured by Microchip Technologies, Inc. Used as a separate instructional tool or installed into the adapter cradle in the Diversified Engineering **Design Center** (P/N 905000), this module provides development systems/components to support experiments and the total lab environment. Its features include:

- On-board 5vdc power supply
- 40 pin Zero Insertion Force (ZIF) socket for the PIC Microcontroller
- 7.3728 MHz crystal for its clock
- All non-dedicated PIC ports brought out to contact pads (J5 and J6)
- SPI memory
- 14 momentary push buttons
- Two potentiometers
- Eight LEDs with limiting resistors
- 2x16 LCD display
- Serial port with RS-232 and RS-485
- Connector for In-Circuit Debugger Module (ICD)

2 Power Sources

2.3 The CAN Education Board operates on an optional AC Power Pak adapter or on 7 to 12 VDC/200 ma DC power. Input power is regulated down to +5VDC for all supplies to the on-board circuits.

2.3.1 **AC Power Pak Adapter** - A Power Pak adapter (DE option #905530) is available for connection into J1 to provide continuous power to the module. The adapter output is 9–12 VDC at 200 ma. The center pin of J1 is positive (+) and the outer shell is negative (-).

2.3.2 **Battery Power Source** - A 9V Panasonic #006PNX battery (or equivalent) installed into the BAT1 connector can be used to power the board. DC Power Source – Between 7 and 12 VDC can be supplied to the board by connecting an external power supply to J1. The center pin of the connector is positive (+) and the outer shell is negative (-).

3 Microcontroller

3.1 The PIC 16F874 is an 8-bit CMOS Flash microcontroller manufactured by Microchip. For information about writing software for, loading programs into and erasing this device please refer to the *Microchip PIC16F874 Data Sheet* available from www.microchip.com. This information may also be obtained directly from Microchip.

******WARNING******

Power Switch S17 must be in the OFF position before you raise the lever on the microcontroller socket (U3) or you will damage the microcontroller.

3.2 To remove the microcontroller (U3), turn off Power Switch S17 and lift the lever on the microcontroller socket (located near pin 1 of the microcontroller) to the vertical position. Lift the microcontroller out of the socket, keeping it level to the surface of the board until all pins have cleared the socket. The microcontroller should lift out with no resistance. If resistance occurs, reset the lever to horizontal and then back to the vertical position. You should never need a tool to lift out the microcontroller!

3.3 To install the microcontroller, turn off Power Switch S17 and lift the lever on the microcontroller socket to the vertical position. Insert the microcontroller into the socket (U3). Verify that Pin 1 is located in the correct hole as designated on the silkscreen and that all pins are placed into all the appropriate holes. Lower the lever 90 degrees into the horizontal position. You should never have to force the microcontroller into the socket!

4 Oscillator (OSC)

4.1 The OSC signal, as supplied by Diversified Engineering, is developed from an on-board crystal (X1) running at 7.3728 MHz.

5 E^2 Memory

5.1 The 25C040 (U5) is a 16K bit serial Electrically Erasable PROM (EEPROM). The memory is accessed through the Serial Peripheral Interface (SPI) control by the microprocessor.

6 Push Buttons

6.1 The MCLR or Master Clear Reset (S14) button is available on the module to manually reset the microcontroller program. The MCLR button is interconnected to VDD (+5V) via a 47K resistor.

6.2 The **Enter** button (S13) is connected across GND and Pin RE2 of the microcontroller and has a 10K resistor to +5VDC. Pressing the **Enter** button will momentarily bring RE2 to GND. See Appendix D.

6.3 There are 12 momentary N.O. pushbutton switches (S1-S12) available to the designer. These switches are arranged in keyboard fashion and wired in a matrix of three columns and four rows. (See Appendix D for details.) Microcontroller pins RB1, RB2, RB3, and RB4 are the driven multiplex output lines (low is a select) to each of the four rows of switches. RA2, RA3 and RA5 are the input lines (pulled high via internal pull-up resistor) of the three columns. Pressing a switch produces a multiplexed output causing the input to go low on its corresponding output line.

******NOTE******

Since RB1, RB2, RB3 and RB4 are also used to drive the LCD display, the software should disable the keyboard inputs when writing to the LCD.

7 Potentiometers (RP2 – RP3)

7.1 There are two adjustable potentiometers available on the board for control. Each generates an analog 0-5 V signal directly into the microcontroller A/D ports RA0 and RA1. The potentiometers are 5K ohm. Their ends are tied to the +5VDC (VDD) and GND (circuit common). Rotating the potentiometer clockwise increases the voltage at its wiper.

8 LED Indicators (LED0 – LED7)

8.1 There are eight LED indicators available on the board. The cathodes are tied to GND (circuit common) and the anodes are connected through a 1k resistor to Port D pins 0 - 7 respectively. The LEDs are rated at 10 ma (Vf=1.0V) to 20 ma (Vf=2.0V).

9 LCD Display (LCD1)

9.1 There is a single 2-line x 16-character LCD display on the board, with each character a 5x7 dot matrix (FEMA # CG1621-SGR1). The LCD is bus-driven with two 4-bit words outputted on the microcontroller pins RB1, RB2, RB3, & RB4. The alphanumeric display can be used to message a multitude of things. RE0 is used for Data/Command select and RE1 is used for Signal enable.

9.2 This chart gives a brief description of each of the controlling pins. Please refer to the Schematic located in Appendix D for further clarification of the hardware interface.

Signal	I/O	Function
RS	I	L: Instructs register (for Write), Busy flag/Address counter (for Read) H: Data Register (for Read & Write)
E	I	Starts Data for Read or Write
DB4	I/O	Bi Directional tristate data bus pin (lsb)
DB5	I/O	Bi Directional tristate data bus pin.
DB6	I/O	Bi Directional tristate data bus pin.
DB7	I/O	Bi Directional tristate data bus pin (msb).

To assist in programming and displaying information, extracts of the data sheet for the LCD display are available in the Appendix. This display uses a Fujitsu FCS2314AK LCD controller and is comparable to the Hitachi HD44780U LCD controller. See **www.diversifiedengineering.net** for full data sheet.

9.3 FEMA, the manufacturer of the LCD, can be reached by phone at 800-292-3362 or by e-mail at fema@femacorp.com.

10 Bread-Boarding Area

10.1 There is an area near the power supply that has a series of holes providing a space for bread boarding custom circuits. These holes are .040” plated through on a .1” x .1” matrix. Within this area are holes tied to +5 VDC (Vdd) located in the upper right hand corner of the hole matrix. Another set of holes is tied to GND (circuit common) located in the lower right hand corner of the hole matrix. For ease of instrument connection, a ground (GND) terminal post is located near the lower left hand corner of the display.

11 Connecting Terminal Blocks

11. All 40 pins of the microprocessor are brought out to holes. This allows the microcontroller to interface with various other components through interconnection of its pins. J5 and J6 are available for soldering wires, or a standard industry connector (Samtec P.N. SSW-110-01-T-D or equal) can be installed. See Schematic in the Appendix D for details of pin configuration.

12 Source Code Utility

12. A source code utility for the PIC16F874 is available to help support experiments with the CAN Education Board. This can be downloaded from our web site at
www.diversifiedengineering.net.
Routines for interfacing with the E^2, LCD display, 4x4 matrix keypad, A/D and others are included in the utility code. Please feel free to download and use this code for *Qik Start*-ing your project.

13 Serial Port

13.1 The microprocessor Universal Asynchronous Receiver Transmitter (UART) can be configured to interface with either RS232 or RS485 driver/receiver.

13.2 Serial control options:

Position	Option
1	PIC UART to RS-232 Driver (DB9 J3)
2	PIC UART to RS-485 Driver (J4)
3	RS-232 (DB9, J3) to RS-485 (J4)

UART JUMPERS

13.3 RS232 / RS485 Connections:

RS-232 (J3)		RS-485	
Pin	**Function**	**Pin**	**Function**
1	Open	1	GND
2	Transmit	2	A
3	Receive	3	B
4	Connected to 6	4	Z
5	Circuit Common	5	Y
6	Connected to 4	6	Connection
7	Connected to 8		
8	Connected to 7		

14 In-Circuit Debugger (ICD) Interface

14.1 Each ***CAN Education Board*** has a connector J2 that permits interfacing the ICD Module for emulation capability to a PC. The configuration of J2 is:

Pin (ICD)	Connection Point
6	No Connection
5	RB6
4	RB7
3	Gnd
2	+Vdd
1	Vpp

Programming, reprogramming and in-circuit debugging require the use of an interface harness (with built in logic) to your PC. This harness assembly is included in the MPLAB- ICD Kit which can be purchased from an authorized Microchip Distributor. Free MP-LAB software is available from the Microchip web site at www.microchip.com.

III The Node and I/O Boards

1 The Node Board (905186)

1.1 The Node Board is designed around the Microchip PIC16F873 microcontroller and MCP2510 CAN controller.

1.2 The Node Board provides an interface between the ***CAN Education Board*** and the I/O board. Its features include:

- 4 Mhz crystal for controlling the PIC16F873
- 16 Mhz crystal for controlling the MCP2510
- PIC16F873 reset switch (S1 RST)
- In circuit debugger module connector (ICD)
- 15-pin connector for interfacing with the I/O (J3)
- 4-position DIP switch for selecting input and output IDs (S4)
- Two 4-pin MTA headers (J1, J2) for CAN bus interfacing with a terminating jumper (J4) for the last node in the system.

2 I/O Board (905187)

2.1 The I/O board provides digital and analog inputs and outputs to the Node Board.

2.2 The form of the inputs and outputs to the Node Board is:

- two analog inputs -- 5k potentiometer (RP1) and photocell (CDS)
- one analog output (LAMP)
- one digital input (S1)
- one digital output (LED1)

3. Interconnection and Initial Power-up

3.1 The CAN-NET Education Board Kit is supplied as four separate components:

- CAN-NET Education Board
- Node Wiring Harness
- CAN-NET Node Board
- CAN-NET I/O Board

3.2 To interconnect the node and the I/O boards:

- Ensure that switch S17 is in the off position
- Connect one end of the harness to J2 of the CAN-Net Education Board.
- Connect the other end to J1 of the CAN-Net Node Board.
- Insure the CAN-Net I/O Board is connected to the Node board through the 15 pin connector

****WARNING****

To prevent circuit pins from shorting out, avoid placing boards on metal or conductive surfaces.

IV Software System Overview

1 CAN ID Structure

1.1 CAN messages are transmitted with an ID to identify them. They are received based on a match between the Mask/Filter settings in the 2510 chip and the message ID. In this implementation, the extended ID is not used and all IDs are assumed to be 11 bits. The code that is shipped with the CAN ED system further assumes that all IDs have the form 0xnm0 where nm is between 0x00 and 0x32 Note that the form used to specify IDs is a 12-bit representation with the ID shifted left into the msb of the 12-bit word. In other words, the lsb is always 0 and is not part of the ID.

2 Software

2.1 Programming Style

2.1.1 The code for the CAN Ed and Node boards is written in the PIC instruction set to be assembled using MicroChip's MPLAB environment. There is significant use of macros to make the code more readable and less error prone. The macros are defined in three places:

near the top of the main file
in the CANLIB.ASM file for CAN macros
in the MACROS16.INC file.

Any unrecognized instructions encountered in the code are probably macros. The macros in MACROS16.INC have been used extensively in writing code for PICs to increase readability and reduce programming errors.

2.2 Common Code

2.2.1 The CAN ED board and the Node board share several common software files. The routines that provide communication with the 2510 CAN chip are in the file CANLIB.ASM and definitions of 2510 registers are in MCP2510.INC. Common macros are in MACROS16.INC.

2.2.2 Communications with the MCP2510 chip is over the SPI bus. The PICs used on the Node board and the CAN Ed board fully support SPI in the master mode. Command strings are sent and received using a single software buffer. To send a string, the software buffer, called pSPIBufBase, is loaded with the bytes to send and the SPI interrupt is turned on. The interrupt handler exchanges bytes with the 2510. The bytes received from the 2510 replace the bytes that were sent from the software buffer so that after the string has been sent the buffer will contain the bytes received from the 2510. All communication with the 2510 is handled in this manner and is encapsulated in the routines in CANLIB.ASM.

3 CAN-NET Education Board

3.1 Software Default ID Structure

3.1.1 The CAN Ed Board code uses these default settings:

RxMask0 0xFFF	RxFilter0 0xFFF	RxFilter1 0xFFF
RxMask1 0xFFE	RxFilter2 0x000	RxFilter3 0x100
	RxFilter4 0x200	RxFilter5 0x300

3.1.2 The potentiometer labeled RP3 is permanently assigned an ID of 0x000. It is measured and transmitted with that ID every 300 mS as a two-byte integer in the range 0 to 4095. Note that the A/D measurement is eight bits which are then left-shifted by four bits before transmission. As a result, the actual range is 0x0000 to 0x0FF0.

3.2 General

3.2.1 The CAN Ed board must be connected to at least one operating Node board or it will not function. If pressing a numerical key on the keypad doesn't change the display then the CAN network is not operating. This means that the transmitted message is not being acknowledged; the CAN Ed program is hung resending the message and waiting for an acknowledgment.)

3.3 Keypad

3.3.1 The keypad is used to select the ID to be displayed. An ID is entered with two key strokes. The first entry selects the "n" for the ID 0xnm0 and the second selects the "m." Note that an ID is displayed and entered as a hex number but the keypad does not contain the hex digits A though F. This restricts the range of IDs that can be selected using the keypad, but the IDs assigned to the various inputs and outputs do not require these hex digits.

3.3.2 When a new ID is entered by the keypad it replaces the ID that is in RxFilter2 regardless of its value or the values in the other filters. This is of concern only when the CAN Ed board is connected to the PC. As described below, the PC program will display all the messages accepted by the CAN Ed board's RxFilters. As a result, changing the value in the RxFilter2 can affect which messages are displayed on the PC.

3.4 LCD

3.4.1 The currently selected ID is displayed along with the most recent message received. The message is assumed to be a two-byte integer with the least significant byte sent first. It is displayed as a decimal number. If no message is received with the selected ID within 1.4 seconds, the previously displayed value is replaced by blanks.

3.5 PC Communications

3.5.1 There is a 115,200 kbaud PC interface that uploads to the PC all messages that match the RxMask/RxFilter values. It allows the PC operator to put messages onto the CAN bus with any ID and with any number of data bytes up to the CAN limit of 8. The PC program can also be used to change the RxMask and RxFilter values of the CAN Ed board.

4 Node Board Software

4.1 General ID Structure

4.1.1 The ID structure used by the Node boards is determined by the DIP switch settings on power-up or after a board reset. Changing the DIP settings while running will have no effect.

4.2 Receive ID Structure

4.2.1 The Node board uses these settings for receiving:

RxMask0	0xFFF	RxFilter0	0xFFF	RxFilter1	0xFFF
RxMask1	0xFFE	RxFilter2	0xn00	RxFilter3	0xn10
		RxFilter4	0xFFF	RxFilter5	0xFFF

where 0xn00 is the Base Receive ID for receiving and is derived from the DIP Swt #1 and DIP Swt #2 as follows:

Swt: 1	2	ID
0	0	0x000
0	1	0x100
1	0	0x200
1	1	0x300

4.2.2 A message received for RxFilter2, the Base Receive ID, is assumed to be a two-byte integer that contains a 12-bit value, that is, a value between 0 and 4095. The 12-bit data is used to generate a PWM output where a value of 0 gives a zero duty cycle, and 0xFFF generates a 100% duty cycle. The PWM output drives the lamp whose output is proportional to the duty cycle.

4.2.3 The ID associated with RxFilter3, Base Receive ID + 0x010, is assumed to be a two-byte integer containing 12-bit data. The data is interpreted as a logical value where a value in the range 0 to 0x7FF is a logic 0, and a value in the range 0x800 to 0xFFF is a logic 1. The logic value is place on the PIC output pin RB2 which is not connected to any output device in this model.

4.3 Transmit ID Structure

4.3.1 The Node board transmits a CAN message every 131 mSec. A message contains two data bytes that represent a 12-bit value with the least significant byte sent first. The Node board cycles between three different sources for the data to be transmitted. Each data source has its own unique ID obtained from the Base Transmit ID that comes from DIP Swt #3 and DIP Swt #4 as follows:

Swt: 3	4	ID
0	0	all transmissions are disabled
0	1	0x100
1	0	0x200
1	1	0x300

4.3.2 The sources of data for transmission are:

- **Potentiometer** The pot is assigned the Base Transmit ID and is measured and transmitted with that ID every 393 mS as a two-byte integer in the range 0 to 4095. Note that the A/D measurement is 8 bits which are then left-shifted by four bits before transmission. As a result, the actual range is 0x0000 to 0x0FF0.
- **Push Button Switch** The switch is assigned Base Transmit ID + 0x010 as an ID. It is measured and transmitted with that ID every 393 mS as a two-byte integer with one of two values: 0x0000 if the switch is open, or 0x0FFF if the switch is closed.
- **CdS detector** The CdS light detector is assigned Base Transmit ID + 0x020 as an ID. It is measured and transmitted with that ID every 393 mS as a two-byte integer with a value in the range 0 to 4095 where the value decreases as the light intensity increases.

5 CAN Manager Program (905193) PC Interface

5.1 General

5.1. The CAN Manager program is a simple tool to control some of the operations of the CAN Ed board and to display selected traffic on the CAN bus. The program is written as a Borland C++ Builder version 3 application, using the Windows serial API for communication through a serial port. It requires Windows 95 or later to operate.

5.1.2 The serial port is set up as a 115,200 kbaud, 8N1 full duplex port using a custom frame protocol that contains a one-byte checksum and byte-stuffing.

5.2 Installation

5.2.1 When the PC program is executed it will require that you select the serial COM port to which the CAN Ed board is connected. After the serial port is chosen the PC program and the CAN Ed board will exchange initialization information. The CAN Ed board will upload the current settings of the RxMask and RxFilter values. These values are displayed by the PC program.

5.3 Receiving Messages

5.3.1 Only those messages that meet the RxMask/RxFilter criteria are accepted by the CAN Ed board and sent to the PC program for display. The PC program examines the ID of each message uploaded from the CAN Ed board and displays it in the associated message box along with its ID. Each time a message is received and displayed, the associated message count is incremented. The message data is displayed as hex bytes in the order they appear in the message.

5.3.2 The RxMask and the RxFilter values can be changed by selecting the associated edit box and entering the desired value as a three digit hex number. This new value will be sent to the CAN Ed board to be sent to the 2510 chip for use in selecting future messages.

5.4 Transmitting messages

5.4.1 A message with an ID can be sent to the CAN Ed board to be put on the CAN bus. The ID should be entered as a three-digit hex number and the data as one or more two-digit hex numbers separated by spaces. The “Scan for Errors” button will process the information entered and rewrite it to the PC if there are no detected syntax errors. It does not send it to the CAN Ed board and is used simply as an optional check before sending the message. The message is sent to the CAN Ed board for a single transmission on the CAN bus by pressing the “Send It” button.

5.5 Trace File

5.5.1 All transactions over the serial port are written into a circular memory buffer. They are then written out to a file called TRACE when the CAN Manager program is exited. The last several thousand bytes sent and received will be in the TRACE file in an ASCII character format.

Appendix A

Board Layout
P/N 905185 Standard

Appendix B

LCD Character Set

Upper / Lower	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	*1	CG		0	@	P	`	p	_	金		ー	タ	ミ	α	P
0001	*2	CG	!	1	A	Q	a	q	_	月	。	ア	チ	ム	ä	q
0010	*3	CG	"	2	B	R	b	r	▬	火	「	イ	ツ	メ	ß	θ
0011	*4	CG	#	3	C	S	c	s	▬	水	」	ウ	テ	モ	ε	∞
0100	*5	CG	$	4	D	T	d	t	▬	木	、	エ	ト	ヤ	μ	Ω
0101	*6	CG	%	5	E	U	e	u	▬	土	・	オ	ナ	ユ	σ	Ü
0110	*7	CG	&	6	F	V	f	v	▬	土	ヲ	カ	ニ	ヨ	ρ	Σ
0111	*8	CG	'	7	G	W	g	w	年	日	ァ	キ	ヌ	ラ	τ	π
1000	*1	CG	(	8	H	X	h	x	|||	上	ィ	ク	ネ	ソ	ｸ	×̄
1001	*2	CG	)	9	I	Y	i	y	|||	下	ゥ	ケ	ノ	ル	ｰ	y
1010	*3	CG	*	:	J	Z	j	z	▬	左	ェ	コ	ハ	レ	万	千
1011	*4	CG	+	;	K	[	k	{		右	ォ	サ	ヒ	ロ	°	円
1100	*5	CG	,	<	L	¥	l			大	ャ	シ	フ	ワ	¢	円
1101	*6	CG	−	=	M	]	m	}	▬	中	ュ	ス	ヘ	ン	±	÷
1110	*7		.	>	N	^	n	→	▬	小	ョ	セ	ホ	゛	ñ	
1111	*8		/	?	O	_	o	←	▬		ッ	ソ	マ	゜	ö	▬

Appendix C

LCD Instruction Set

Instruction	Code RS	R/$\overline{W}$	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Clear display	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DDRAM address 0 in address counter.	
Return home	0	0	0	0	0	0	0	0	1	—	Sets DDRAM address 0 in address counter. Also returns display from being shifted to original position. DDRAM contents remain unchanged.	1.52 ms
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Sets cursor move direction and specifies display shift. These operations are performed during data write and read.	37 µs
Display on/off control	0	0	0	0	0	0	1	D	C	B	Sets entire display (D) on/off, cursor on/off (C), and blinking of cursor position character (B).	37 µs
Cursor or display shift	0	0	0	0	0	1	S/C	R/L	—	—	Moves cursor and shifts display without changing DDRAM contents.	37 µs
Function set	0	0	0	0	1	DL	N	F	—	—	Sets interface data length (DL), number of display lines (N), and character font (F).	37 µs
Set CGRAM address	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Sets CGRAM address. CGRAM data is sent and received after this setting.	37 µs
Set DDRAM address	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Sets DDRAM address. DDRAM data is sent and received after this setting.	37 µs
Read busy flag & address	0	1	BF	AC	AC	AC	AC	AC	AC	AC	Reads busy flag (BF) indicating internal operation is being performed and reads address counter contents.	0 µs
Write data to CG or DDRAM	1	0	Write data								Writes data into DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
Read data from CG or DDRAM	1	1	Read data								Reads data from DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
	I/D = 1: Increment I/D = 0: Decrement S = 1: Accompanies display shift S/C = 1: Display shift S/C = 0: Cursor move R/L = 1: Shift to the right R/L = 0: Shift to the left DL = 1: 8 bits, DL = 0: 4 bits N = 1: 2 lines, N = 0: 1 line F = 1: 5 × 10 dots, F = 0: 5 × 8 dots BF = 1: Internally operating BF = 0: Instructions acceptable										DDRAM: Display data RAM CGRAM: Character generator RAM ACG: CGRAM address ADD: DDRAM address (corresponds to cursor address) AC: Address counter used for both DD and CGRAM addresses	Execution time changes when frequency changes Example: When f_{cp} or f_{OSC} is 250 kHz, $37\ \mu s \times \frac{270}{250} = 40\ \mu s$

Note: — indicates no effect.

* After execution of the CGRAM/DDRAM data write or read instruction, the RAM address counter is incremented or decremented by 1. The RAM address counter is updated after the busy flag turns off.

Appendix D

Additional CAN Expansion Node Boards Available from Diversified Engineering

The CAN-NET Education Board Kit is supplied with one node. The kit has the ability to control up to four separate nodes. Additional Node Boards are available from Diversified Engineering:

CAN-NET LCD Node Kit (PN 905188)

Kit includes:

- 1 CAN-NET Node Board
- 1 CAN-NET Wiring Harness
- 1 CAN-NET LCD Display Board with
  - 2x16 LCD display
  - 1 Switch

CAN-NET Analog Output Node Kit (PN 905189)

Kit includes:

- 1 CAN-NET Node Board
- 1 CAN-NET Wiring Harness
- 1 CAN-NET Analog Output Board with:
  - PWM Outputs converted to
    - 0-5v Output
    - Incandescent Lamp
    - PWM Output at 200ma
    - 10 position Bar Graph
  - CDS Photocell Analog Input
  - PWM Feedback Analog Input

CAN-NET Analog Input Node Kit (PN 905190)

Kit includes:

- 1 CAN-NET Node Board
- 1 CAN-NET Wiring Harness
- 1 CAN-NET Analog Input Board including:
  - Microchip MCP3201 12 Bit AD Converter
  - Option for customer installed 14.5 PSI Pressure Transducer
  - 10k Thermistor
  - EEPROM Memory

CAN-NET Node Board (PN 905186)

- Comes complete with a pre-programmed PICmicro microcontroller (PIC 16F876 allowing for easy interconnection of multiple boards into a single network.

CAN-NET Prototype Board (PN 905191)

- Provides developers a bread boarding area for node development. The Prototype Board connects to a separately supplied Node Board.

Limited Warranty

Diversified Engineering and Manufacturing Inc. (DEM) warrants that the Product(s) will be free from defects in material and workmanship for a period of one (1) year from the date of purchase. Diversified Engineering & Manufacturing Inc. will, at its discretion, repair or replace any part(s) found to be defective in the Product(s) resulting from defective workmanship, material or both. All costs for packaging and transporting *to and from* Orange CT are the responsibility of the customer. All units returned for repair should have prepaid shipping charges, include a proof of purchase receipt and a detail of the problem/symptom.

There is no other Warranty, express or implied. This warranty does not apply to any defect, failure or damage caused by improper use or storage of the product. DEM will not be obliged to provide warranty service for units that:

1) have been damaged from improper use or interconnect to external equipment;
2) have been modified or tampered with or
3) have been improperly stored or exposed to the elements.

Customer Service

Technical support and service is available from:
9:00 AM to 4:30 PM , M-F, EST.
TEL (203) 799-7875 FAX (203) 799-7892
All materials should be addressed to:
Diversified Engineering and Manufacturing Inc.
283 Indian River Road
Orange, CT 06477

This publication and the information provided are subject to change, errors and omissions.
PICmicro and PIC are trade names of Microchip Technology Inc.

Copyright 2000 Diversified Engineering and Manufacturing Inc.
All rights reserved.