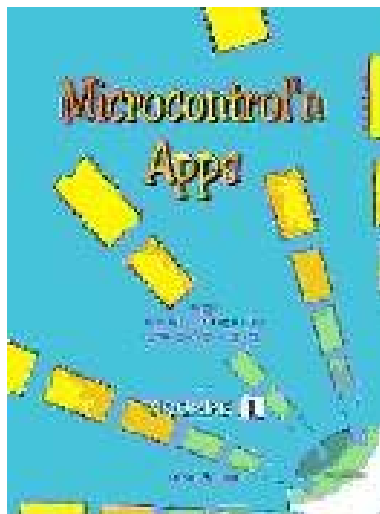


Microcontrol'n Apps

Using PIC ® Microcontrollers

by David Benson



Order Code 439-4331

"Microcontrol'n Apps" © is an intermediate level applications guide covering Microchip Technology's PIC ® Microcontrollers. Serial communication as a means of transferring data between PIC microcontrollers and

peripheral chips and also between two or more PIC microcontrollers is described. Use of the 93C46 serial EEPROM is detailed as an example. Since we live in an analog world, A/D and D/A are discussed with several methods illustrated for each. Conditioning signals from sensors with an analog voltage output is described. Interfacing PIC microcontroller-controlled systems with humans requires some math, binary to decimal conversion and vice versa, alphanumeric LCD interfacing and scanning keypads. Single wire serial communication with a PIC-controlled LCD module which can be built by the reader is included. A digital thermometer project brings these topics together as an example. The book also explains how to establish serial communication between a PIC microcontroller and a PC via a RS-232 conversion circuit and a terminal program. These techniques are used in a digital voltmeter/data logger experiment for uploading data to a PC for display plus graphing using a spreadsheet program. Moving up or down from the now familiar PIC16F84 to other devices is covered in detail so that you will easily be able to work with the new devices being introduced by Microchip. Finally, use of the Microchip in-circuit debugger (ICD) is described. "Microcontrol'n Apps" gives the reader the tools to design, build, and debug intermediate level microcontroller-based instrumentation and systems. By David Benson (8-1/2 x 11 format, 437 pages).

CONTENTS

PREFACE

INTRODUCTION

PRODUCT OVERVIEW

CIRCUIT MODULES FOR EXPERIMENTS

USING OP-AMPS

SERIAL COMMUNICATION

SHIFT REGISTERS

Serial In, Parallel Out Shift Register - 74HC164
Parallel In, Serial Out Shift Register - 74HC165
Serial In, Parallel Out Shift Register - 74HC595

SERIAL EEPROMS

Demo Circuit
Main Program - Initial Test

SERIAL COMMUNICATION BETWEEN TWO PIC MICROCONTROLLERS

LIQUID CRYSTAL DISPLAY INTERFACE

1 X 16 LCD

- Pins And Functions
- Data vs. Instruction
- Display Control
- Character Addresses
- LCD Operation
 - PIC/LCD Circuit
 - Timing And Pulsing
 - Testing The Circuit
- Display RAM
- Initialization
- ASCII
- Example Routines For LCD
 - Fill Display With Blanks
 - Display "HELLO"
 - LCD Initialization

- Character Addresses
- More on Alphanumeric Character Addresses
 - Display 16 Characters
 - Display Hex Byte Subroutine
 - Blanks
- Separate A Hex Byte Into Two ASCII Digits
 - Hex Digit To ASCII Conversion
 - Hex To Bits Subroutine
 - Program Listing
 - To Use/Test Display Hex Byte
- 4-Bit Mode
- LCD Module Serial Interface
- LCD Experiments
- More About ASCII
- LCD Font table

SCANNING KEYPADS

- Software Design
- Scan Decimal Subroutine
- Using Keypad And LCD With PIC Microcontroller
- Debounce
- Function Keys

DIGITAL TO ANALOG CONVERSION

- Do It Yourself D/A Using A Resistor Network
 - 8-Bit Parallel D/A Converter - AD558
 - Do It Yourself D/A Using Pulse Width Modulation
 - PWM Basics
 - Low Pass Filters
- PWM Using A Filter With Unity Gain Follower
 - More PWM Philosophy
 - Analog Output - Increase/Decrease Buttons

- PWM Using Software, TMR0 And Interrupts -
 - Philosophy
 - Hardware PWM
- 8-Bit Serial D/A Converter - MAX522
 - Output A Voltage Level
 - Output A Ramp Voltage
 - Output A Sine Wave

SENSORS - ANALOG VOLTAGE OUTPUT

- LM335 Temperature Sensor
- Offset And Scale
- Three Amplifier Design
- Single Amplifier Design
- Why 1 Op-amp vs. 3 Op-amps?

ANALOG TO DIGITAL CONVERSION

- PIC Pin And RC Time Constant
 - Measuring Resistance
 - Serial 8-Bit A/D Converter - ADC0831
 - ADC0831 Built-In Offset And Scale
- Temperature Measurement Using LM335
 - Temperature Sensor
 - Three Op-amp Offset And Scale
- ADC0831/LM335 Temperature Measurement

- PIC16C711 On-Board A/D (8-bit)
 - A/D Control Registers
 - Configuration Bits
 - Example
- PIC16F870 On-Board A/D (10-bit)
 - A/D Control Registers
 - A/D Conversion Procedures
 - Example

USING COMPARATORS

- Comparator Control Register - CMCON
- Voltage Reference Control Register - VRCON
- Single Comparator Example
- Comparator Interrupts

MATH ROUTINES

Instructions

- Arithmetic
 - Addition
 - Subtraction
 - Multiplication
- Double Precision

- Addition
- Subtraction
- Multiplication
- Multiply A 2-Byte Binary Number By Decimal 10
- 8-Bit X 8-Bit Multiply, 2-Byte Result

DECIMAL INTERFACE

- 3-digit decimal to 8-bit binary**
- Using the 3-digit decimal to 8-bit binary decimal entry program
 - 8-bit to 3-digit BCD**
 - Display result of 8-bit binary to 3-digit BCD
- 16-bit binary to 5-digit BCD -**
 - range 0x0000 to 0x7FFF
- 16-bit binary to 5-digit BCD -**
 - range 0x0000 to 0xFFFF

DIGITAL THERMOMETER

- Building blocks**
- Rounding off**
- Displaying temperature via a LCD**

SIMPLE DIGITAL VOLTMETER EXPERIMENT

TALKING TO A PIC MICROCONTROLLER WITH A PC VIA A WINDOWS TERMINAL PROGRAM

- "U-turn" experiment
- PC-to-PC "2-lane highway" experiment
- Importing a text file into a spreadsheet program
 - Windows 98
- PC/PIC Microcontroller
 - PC baud rates
 - Modify ser_out subroutine
 - Modify ser_in subroutine
 - PIC to PIC at 4800 baud, LSB first
 - RS-232 interface for a PIC microcontroller
 - RS-232 converter circuit using MAX233
 - PC to PIC microcontroller serial communication
 - Display one ASCII character via 8 LEDs
 - PIC microcontroller to PC serial communication
 - Send one ASCII character
 - Code for formatting PIC microcontroller data on a PC screen
- PC to PIC/LCD
 - Control characters
 - Sending a text file (control characters and data)

SIMPLE DATA LOGGER EXPERIMENT

- Main program**
- Data logging**
- Display data sequentially via LCD**
- Uploading data to a PC**
- Code**
- Operating procedure**
- Spreadsheet and graphing data - Windows 98**

MOV'n UP OR DOWN

- Pin function options and how to select them**
- Clock oscillator options**
 - Example - PIC16F627/628
 - Example - PIC16F629/675
- External reset vs. digital I/O option**
 - Example - PIC16F627/628
 - Example - PIC16F629/675
- Special function registers**
- General purpose file registers**
- CBLOCK assembler directive**
- Configuration words**

PIC16F628

- Pins and functions**
- Package**
- Ports**
- Architecture - overview**

- Program memory
- File registers
- Special purpose registers - overview
 - Status register
 - Option register
 - Program counter
 - Control registers
- Comparator control register - CMCON
- Configuration bits
- PIC16F628 programming examples
 - Example - pict1.asm adapted
 - Example - internal 4 MHz clock, no MCLR

PIC12F675 - 8-PIN MICROCONTROLLER

- Pins and functions
- Package
- Ports - GPIO
- Clock oscillator options - covered previously
- External reset vs. digital input pin (GP3) - covered previously
- Architecture - overview
 - Program memory
 - File registers
 - Special purpose registers - overview
 - Status register
 - Option register
 - Program counter
 - Control registers
 - Comparator control (CMCON) register
 - Analog select (ANSEL) register
 - Tristate I/O (TRISIO) register
- Configuration bits
- Calibrating the internal 4 MHz RC clock oscillator
 - Device programming considerations related to internal oscillator calibration
- Example program
 - Led pattern
 - Internal reset
 - Internal clock oscillator

PIC16F870

- Pins and functions
- Package
- Ports
- Architecture - overview
 - Program memory
 - File registers
 - Special purpose registers - overview
 - Status register
 - Option register
 - Program counter
 - Control registers
 - A/D control (ADCON1) register
- Configuration Bits
- F870 vs. F84

CIRCUIT MODULES FOR F870 EXPERIMENTS

- Building Your Own Simple Test Board - 87s Board
- 87s Companion Board

PROGRAMMING THE F870 USING A DEVICE PROGRAMMER

- First F870 Program - To be programmed via a Device Programmer

PORTING YOUR APPLICATION FROM F84 To F870 - F870

- PROGRAMMED VIA A DEVICE PROGRAMMER

PIC16F877

- Disable A/D on port E
- Connect both power and both ground pins

DEBUG'n

GETTING STARTED

- PIC16F87x series
- F870 - my candidate for the debug'n experimenter's part of choice
- F876 - my second choice
- F84 vs. F870 for learning purposes
- ICD vs. ICD2
- Device programmer vs. bootloader vs. ICD
 - Bootloaders
 - Microchip ICD
- What a debugger can do for you
- Debugging methodology
 - Single stepping
 - Breakpoint
 - Watch window
 - Debugging

MICROCHIP ICD

Description

User Board = Target Board

Using The Microchip ICD

- General considerations
- First project
 - First F870 program for use with ICD
- MPLAB Operations
 - Setting up the ICD
 - Toolbar
 - To run a program in real time via the toolbar
 - To reset the F870 via the tool bar
 - Watch window
 - Single stepping
 - Break point
 - Break on address match
 - Clear breakpoint
 - Break on user halt
 - Powering down
 - Operating the 87s board stand alone after debug'n
 - Reconnecting the ICD After 87s board stand alone operation
 - Firing up MPLAB and opening an existing project
- Conclusion

Porting your application from F84 to F870 - F870 PROGRAMMED VIA ICD

USING THE ICD AS A MINI IN-CIRCUIT DEBUGGER FOR F84, F628 ETC.

APPENDICES

- Appendix A - Sources
- Appendix B - Hexadecimal Numbers
- Appendix C - Program Listings vs. Page Numbers