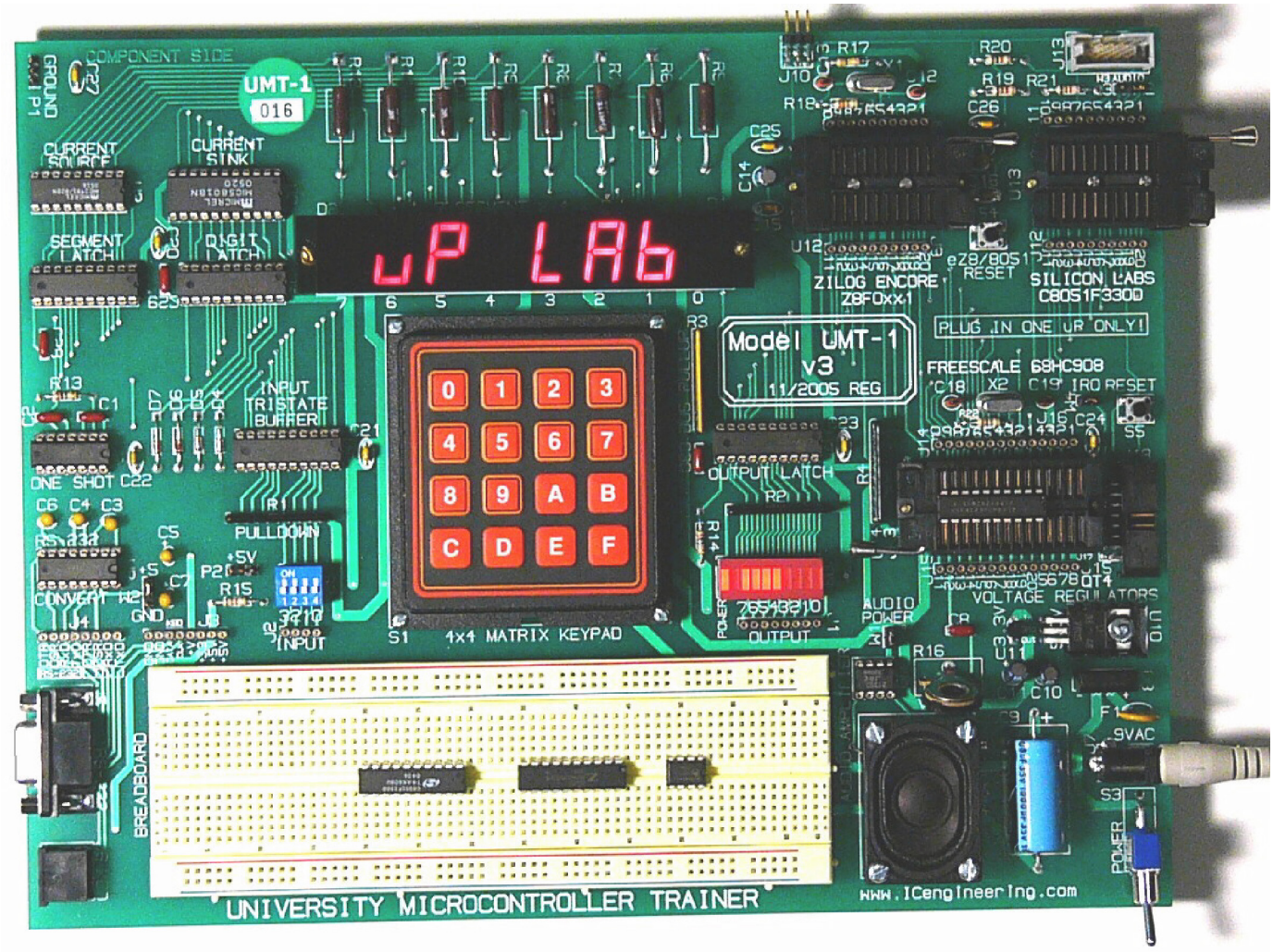
MICROCONTROLLER LAB

Programming the 68HC08, 8051, and eZ8 in Assembly Language

Robert E. Glaser



Microcontroller Lab

Programming the 68HC08, 8051, and eZ8 in Assembly Language

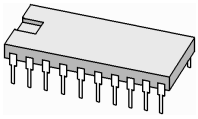
Second Edition

Robert E. Glaser, Ph.D.

IC Engineering, Inc.

The Johns Hopkins University

IC Engineering, Inc.



PO Box 321
Owings Mills, MD
21117-0321 USA

www.ICengineering.com
410-363-8748

Second Edition: September 2006
First Edition: September 2005

ISBN 0-9772246-1-9

The *University Microcontroller Trainer*, Model UMT-1 used in the text is available from IC Engineering, Inc.

This project is dedicated to C. Roger Westgate, who led the way

Table of Contents

Table of Contents 3

68HC908 Related 7
C8051F330D Related 7
Z8F0421 Related 8

Assignments 8

0. Introduction 9
    0.0 Prerequisites 9
    0.1 Microcontrollers 9
    0.2 General Concepts 9
        Number Bases 9
        Signed and Unsigned Numbers 10
        Memory Types 11
        Inside the Microcontroller 12
        Of HLL's and Assembly Language 14
    0.3 To the Instructor 15
        Hardware Requirements 15
        Software Requirements 15
        Support Documentation 15

1. Programming the 68HC08 16
    1.0 Background 16
    1.1 Getting Started 16
    1.2 Sample Programs 16
    1.3 Instructions and Addressing Modes 17
    1.4 Sample A 17
    1.5 Sample B 19
    1.6 Sample C 19
    1.7 Sample D 21
    1.8 Sample E 21
    1.9 Sample F 22
    1.10 Assignment 1a: Test Sample Code 22
    1.11 Assignment 1b: Data List 24
    1.12 Report Write-up 24
    Sample 68HC08 code Listing 25
    Freescale 68HC08 Documentation 27

2. Hardware and I/O 45
    2.0 Microcontroller Packaging and Common Features 45
    2.1 Freescale 68HC908JK3 46
    2.2 Microcontroller Trainer 46
    2.3 Parallel Output and the LED's 47
    2.4 Ports on the 68HC908JK3 48
    2.5 Let There Be Light 48
        In-Circuit Simulator 48
        Flash Programmer 49
        In-Circuit Debugger 49
    2.6 Input Port 50
    2.7 Board Input 50
    2.8 Microcontroller Input/Output 51
    2.9 The Stack and Subroutines 51
    2.10 Assignment 2a: Switch/LED Shift Register 52
    2.11 Display Multiplexing 53
    2.12 Keyboard Scanning 54
    2.13 Assignment 2b: Exercise the Keypad and Display 54

BlankJK3.asm/ioJK3.asm Listings . . . 55
University Microcontroller Trainer Parts List . . . 61
University Microcontroller Trainer Schematic Diagrams . . . 63

3. The Calculator . . . 67
3.0 Software . . . 67
3.1 Loops . . . 67
3.2 Parameter Passing . . . 68
3.3 Lookup Tables . . . 68
Direct Access . . . 68
Search Key . . . 70
3.4 Assignment 3: The Calculator . . . 71
3.5 Data Structures . . . 71
One Decimal Digit per Byte . . . 72
Two Decimal Digits per Byte . . . 74
3.6 Memory Allocation . . . 74
3.7 Development and Testing . . . 75
3.8 Optional Features . . . 75
Backspace Key . . . 75
Subtraction . . . 75
Decimal Fractions . . . 76
3.9 Expectations . . . 76
3.10 Documentation . . . 76

4. Programming the 8051 . . . 78
4.0 Background . . . 78
4.1 Architecture . . . 78
4.2 Silicon Laboratories C8051F330D . . . 79
4.3 Address Spaces . . . 79
4.4 Registers . . . 79
4.5 Flags and the Program Status Word . . . 80
4.6 Addressing Modes . . . 80
4.7 Instruction Set . . . 80
Data Transfer Group . . . 80
Arithmetic Operations . . . 82
Logical Operations . . . 82
Boolean Manipulation . . . 83
Stack Operations . . . 83
Program Branching . . . 84
4.8 Hardware . . . 86
Port Operation . . . 86
4.9 Sample Program . . . 86
4.10 IDE Software . . . 88
4.11 Assignment 4a: Sample51 Modifications . . . 88
4.12 Assignment 4b: 8051 Traffic Signal . . . 89
State Transition Table . . . 89
Sample Code for the C8051F330D Microcontroller . . . 90
io51.asm Listing . . . 91
Silicon Laboratories C8051F330D Documentation . . . 96

5. Programming the eZ8 . . . 107
5.0 Background . . . 107
5.1 The Z8F0421 . . . 107
5.2 Registers and RAM Addressing . . . 107
RAM, the Register File, Registers, and Working Registers . . . 108
5.3 Addressing Modes . . . 110
5.4 Instructions . . . 111
Arithmetic Instructions . . . 111
Bit Manipulation Instructions . . . 111
Block Transfer Instructions . . . 112

Load Instructions 112
Logical Instructions 112
Program Control Instructions 112
Rotate and Shift Instructions 112
Instruction Timing 113
5.5 Hardware 113
5.6 Zilog IDE Software 114
5.7 Assignment 5a: LED Action with the Z8F0421 114
5.8 Assignment 5b: Keypad/Display Action with the Z8F0421 114
Sample Code for the Zilog Z8F0421 Microcontroller 115
DemoZ8.asm Listing 116
ioZ8.asm Listing 117
ZiLOG Z8F0421 Documentation 123

6. Interrupts 143
6.0 Polling 143
6.1 Interrupts 143
6.2 Asynchronous Operation 144
6.3 Level and Edge Sensitive Interrupts 145
6.4 Interrupts on the 68HC908JK3 146
Assignment 6f: IRQ Interrupt on the 68HC908JK3 146
Using Other Interrupts 147
6.5 Interrupts on the C8051F330D 148
Assignment 6s: External Interrupt on the C8051F330D 150
6.6 Interrupts on the Z8F0421 151
Assignment 6z: PC0 Interrupt on the Z8F0421 151

7. Timers 152
7.0 Counters 152
7.1 Timers 152
7.2 Timers in the 68HC908JK3 153
Assignment 7f: Heartbeat on the 68HC908JK3 154
7.3 Timers in the C8051F330D 155
Assignment 7s: Heartbeat on the C8051F330D 155
7.4 Timers in the Z8F0421 156
Assignment 7z: Heartbeat on the Z8F0421 156
7.5 Assignment 7c: The Clock 157
Single Digit per Byte 157
BCD Format 157
Time Set Methods 158
Optional Features 158
7.6 General Usage Timers 158

8. Serial Communication 159
8.0 Parallel Communication 159
8.1 Synchronous Serial Communication 159
8.2 Asynchronous Serial Communication 160
8.3 Hardware and Software UART's 161
8.4 Electrical Interface 162
8.5 UART's on the 68HC908JK3 163
8.6 UART's on the C8051F330D 164
Bit Rate Generator 165
Pinouts 166
Polling Method 166
Interrupt Method 166
8.7 UART's on the Z8F0421 167
Bit Rate Generator 168
Pinouts 168
Polling Method 168
Interrupt Method 168

8.8 Assignment 8: Mini Terminal 170
Optional Features 170
8.9 ASCII Table 171

9. Digital to Analog Conversion 173
9.0 Purpose 173
9.1 Pulse Width Modulation 173
Delay Loops 174
Timer Interrupts 174
Direct Timer PWM Output 174
9.2 PWM on the 68HC908JK3 174
9.3 PWM on the C8051F330D 176
8-Bit Pulse Width Modulator Mode 176
16-Bit Pulse Width Modulator Mode 176
PWM Output Pins 177
9.4 PWM on the Z8F0421 178
PWM Pinouts 178
9.5 Assignment 9a: PWM Lamp Control 178
9.6 The IDAC on the C8051F330D 179
9.7 Assignment 9b: Function Generator 180
Delay Loop Method 181
Timer Interrupt Method 181

10. Analog to Digital Conversion 183
10.0 General Considerations 183
10.1 The ADC on the 68HC908JK3 184
10.2 Assignment 10v: Voltmeter 186
10.3 The ADC on the C8051F330D 187
Temperature Sensor 190
10.4 Assignment 10t: Thermometer 191
10.5 The ADC on the Z8F0421 192
10.6 Assignment 10zs: Data Sampling and Storage 193
The Z8F0421 Task 193
The C8051F330D Task 193
Operation 194
10.7 Assignment 10s: Data Sampling and Storage with one Microcontroller . 195
The Data Acquisition Task 195
The Playback Task 195
Operation 195

11. Keyboard Interface 196
11.0 General 196
11.1 Interface Specifications 196
11.2 Assignment 11: Keyboard Receiver/Transmitter 197
Listen to the Keyboard 197
Send to the Computer 197
11.3 Keyboard Scan Codes 198

12. The 68HC908QT4 199
12.0 Do I Have to Learn an Entirely New Microcontroller? 199
12.1 The 68HC908QT4 versus the 68HC908JK3 199
12.2 Pin Configuration 200
12.3 Oscillator 201
12.4 Using the 68HC908QT4 on the Trainer (and DemoQT4 Code) 202
12.5 Assignment 12: 68HC908QT4 203

13. Flash Memory 204
13.0 Parameter Storage 204
13.1 Microcontroller Flash Memory 204
13.2 Flash in the C8051F330D 205

Erase a 512-byte flash page with this procedure: 206
Program (Write) an erased flash byte with this procedure: 206
13.3 Assignment 13s: Writing to the C8051F330D's Flash Memory 206
13.4 Flash in the Z8F0421 207
Erase a 512-byte flash page with this procedure: 207
Program (Write) an erased flash byte with this procedure: 208
13.5 Assignment 13z: Writing to the Z8F0421's Flash Memory 208
13.6 Flash in the 68HC908JK3 and the 68HC908QT4 209
Erase a 64-byte flash block with this procedure: 209
Program (Write) an erased flash byte with this procedure: 210
Erase and Write from RAM 210
13.7 Assignment 13k: Keyboard Logger 210

14. Wrap It Up 211
14.0 Backtracking 211
14.1 System Design 211
14.2 Defensive Programming 212
Faulty Code 212
Unexpected User Interaction 213
Unanticipated Hardware Response 213
Hardware Upset or Failure 213
14.3 Watchdog Timers 213
The Watchdog Timer in the 68HC908JK3/QT4 214
The Watchdog Timer in the C8051F330D 214
The Watchdog Timer in the Z8F0421 215
14.4 Production Testing 215
14.5 Intellectual Property Protection 216
Code Protection in the 68HC908JK3/QT4 216
Code Protection in the C8051F330D 216
Code Protection in the Z8F0421 216

About the author 217

68HC908 Related

1. Programming the 68HC08 16
2.1 Freescale 68HC908JK3 46
2.4 Ports on the 68HC908JK3 48
6.4 Interrupts on the 68HC908JK3 146
7.2 Timers in the 68HC908JK3 153
8.5 UART's on the 68HC908JK3 163
9.2 PWM on the 68HC908JK3 174
10.1 The ADC on the 68HC908JK3 184
12. The 68HC908QT4 199
13.6 Flash in the 68HC908JK3 and the 68HC908QT4 209
The Watchdog Timer in the 68HC908JK3/QT4 214
Code Protection in the 68HC908JK3/QT4 216

C8051F330D Related

4. Programming the 8051 78
6.5 Interrupts on the C8051F330D 148
7.3 Timers in the C8051F330D 155
8.6 UART's on the C8051F330D 164
9.3 PWM on the C8051F330D 176
9.6 The IDAC on the C8051F330D 179
10.3 The ADC on the C8051F330D 187
13.2 Flash in the C8051F330D 205
The Watchdog Timer in the C8051F330D 214
Code Protection in the C8051F330D 216

Z8F0421 Related

5. Programming the eZ8 107
6.6 Interrupts on the Z8F0421 151
7.4 Timers in the Z8F0421 156
8.7 UART's on the Z8F0421 167
9.4 PWM on the Z8F0421 178
10.5 The ADC on the Z8F0421 192
13.4 Flash in the Z8F0421 207
The Watchdog Timer in the Z8F0421 215
Code Protection in the Z8F0421 216

Assignments

1.10 Assignment 1a: Test Sample Code 22
1.11 Assignment 1b: Data List 24
1.12 Report Write-up 24
2.5 Let There Be Light 48
2.7 Board Input 50
2.8 Microcontroller Input/Output 51
2.10 Assignment 2a: Switch/LED Shift Register 52
2.13 Assignment 2b: Exercise the Keypad and Display 54
3.4 Assignment 3: The Calculator 71
3.10 Documentation 76
4.11 Assignment 4a: Sample51 Modifications 88
4.12 Assignment 4b: 8051 Traffic Signal 89
5.7 Assignment 5a: LED Action with the Z8F0421 114
5.8 Assignment 5b: Keypad/Display Action with the Z8F0421 114
Assignment 6f: IRQ Interrupt on the 68HC908JK3 146
Assignment 6s: External Interrupt on the C8051F330D 150
Assignment 6z: PC0 Interrupt on the Z8F0421 151
Assignment 7f: Heartbeat on the 68HC908JK3 154
Assignment 7s: Heartbeat on the C8051F330D 155
Assignment 7z: Heartbeat on the Z8F0421 156
7.5 Assignment 7c: The Clock 157
8.8 Assignment 8: Mini Terminal 170
9.5 Assignment 9a: PWM Lamp Control 178
9.7 Assignment 9b: Function Generator 180
10.2 Assignment 10v: Voltmeter 186
10.4 Assignment 10t: Thermometer 191
10.6 Assignment 10zs: Data Sampling and Storage 193
10.7 Assignment 10s: Data Sampling and Storage with one Microcontroller 195
11.2 Assignment 11: Keyboard Receiver/Transmitter 197
12.5 Assignment 12: 68HC908QT4 203
13.3 Assignment 13s: Writing to the C8051F330D's Flash Memory 206
13.5 Assignment 13z: Writing to the Z8F0421's Flash Memory 208
13.7 Assignment 13k: Keyboard Logger 210

0.3 To the Instructor

Chapters 0-5 are intended for the first semester, and the remaining chapters for a second semester. I have used a shortened version for the accelerated summer session, in which the students have much less time to complete their assignments: Chapters 0-4 for the first session, followed by Chapters 6-10 for the second session (with the eZ8 left out). I also reduced the assignment requirements for the shortened course.

Hardware Requirements

In addition to the **University Microcontroller Trainer UMT-1**, the following materials are needed:

For the 68HC908JK3/QT4: Either the USB-ML-MON08 (USB version) or the MON08 Multilink (parallel port version) from P&E Microcomputer Systems, `www.PEmicro.com`. The USB version is more convenient and less expensive; the parallel port version includes a wall wart power supply, but you'll also need an IEEE-1284 extension cable for it. The software is more robust with the parallel interface.

For the C8051F330D: Silicon Laboratories part number C8051F330DK, "DEV KIT FOR C8051F330/F331." SiLabs' University Program offers a discount. `www.Digikey.com` 336-1264, `www.Mouser.com` 634-C8051F330DK-G. The development kit includes a USB Adapter and the Standard JTAG Cable EC-SJ; these can be purchased separately from `www.SiLabs.com`, but the development kit includes the software. You get a small surface mount board with the kit which we don't use.

For the Z8F0421: Zilog part number Z8F08200100KIT, "DEV KIT FOR Z8 ENCORE 4K TO 8K" (Serial interface). `www.Digikey.com` 269-3183, `www.Mouser.com` 692-Z8F08200100KIT. You get a small surface mount board with the kit which we don't use.

Small Parts: the assignments use the following:

Section 9.5: 2N7000 and lamp (Mouser part number 606-CM7219, `www.Mouser.com`)

Section 9.6: 1K ohm resistor

Sections 10.2, 10.6, 10.7: 1N4148, 1K ohm resistor

Section 11.2: PS/2 keyboard

Section 13.7: 6-pin mini-DIN plug (Digikey CP-2060, Mouser 171-2606); a keyboard extender cable (Digikey AE9875, Mouser 545-P222-010) also helpful

Laboratory Computers: Workstation PC's with Win2000, WinXP, or Win98 (USB finicky with 98); a serial port; and either a USB or a parallel port, dependent upon which MON08 interface is used.

Software Requirements

For the 68HC908JK3/QT4: `Drivers_install.exe` if the USB MON08 pod is used, from P&E Microcomputer Systems, a free download at `www.PEmicro.com`.

For the 68HC908JK3: `ics08jlz_install.exe`, from P&E Microcomputer Systems, a free download at `www.PEmicro.com`.

For the C8051F330D: Silicon Laboratories IDE V1.9. Comes with the C8051F330DK development kit.

For the Z8F0421: Zilog "ZDS II - Z8 Encore! Family 4.9.5". Comes with the Z8F08200100KIT development kit. Also a free download at `www.Zilog.com`.

For the 68HC908QT4: `ics08qtqyz_install.exe`, from P&E Microcomputer Systems, a free download at `www.PEmicro.com`.

Support Documentation

I provide a folder on a network drive accessible from the student workstations containing supplementary documentation. Also check the `www.ICengineering.com` UMT-1 support section. The following are PDF documents which are free downloads from the manufacturers' websites:

"MC68HC908JL3/H - Rev 1.0", "MC68HC908QY4/D Family Data Sheet Rev. 4", "HCS08 Family Reference Manual HCS08RMv1/D Rev. 1 6/2003": `www.Freescale.com`

"C8051F330/1/2/3/4/5 C8051F330D Rev. 1.3 3/05": `www.SiLabs.com`

"Product Specification Z8 Encore!® 8K Series PS022509-0305", "eZ8 CPU User Manual UM012811-0904": `www.Zilog.com`.

P0SKIP, P1SKIP

We can instruct the CPU to skip port pins when assigning them to peripherals with the P0SKIP (0D4h) and P1SKIP (0D5h) SFR's. Setting a bit in these registers makes the corresponding chip pin unavailable for peripherals. Bit assignments look, in order, at bits 0 to 7 in P0 and bits 0 to 7 in P1. So to assign CEX0 to P0.4, make P0SKIP = 00001111; to assign CEX0 to P0.5, make P0SKIP = 00011111.

9.4 PWM on the Z8F0421

Review section 7.4. The two timers also have a PWM mode of operation: to activate it, set the TMODE2-TMODE1-TMODE0 bits in the Timer Control Register to 011 (instead of 001 for continuous mode we previously employed). The prescaler and timer reload registers operate in the same way as for the continuous mode.

Each timer also has two PWM value registers: TxPWMH and TxPWML (x=0,1) which hold the match value high and low bytes, respectively. These are read/write registers. Their addresses are:

T0PWMH=0F04h T0PWML=0F05h T1PWMH=0F0Ch T1PWML=0F0Dh

If the Timer Control Register TPOL bit = 1, upon timer reload the output pin goes high; when the timer counter matches the value in the PWM registers the output pin goes low (active high PWM output). The duty cycle is (PWM value)/(reload value).

If the Timer Control Register TPOL bit = 0, upon timer reload the output pin goes low; when the timer counter matches the value in the PWM registers the output pin goes high (active low PWM output). The duty cycle is (reload value - PWM value)/(reload value).

The PWM period is (reload value x prescale value)/(system clock frequency).

These timers operate in 16-bit mode only; however, since you define the reload value you can set the maximum timer count to any value less than 65,536 (making it 255 creates 8-bit PWM). The PWM registers should be loaded with values less than or equal to the reload registers.

You can interrupt as for continuous mode, but this is not needed for basic PWM generation.

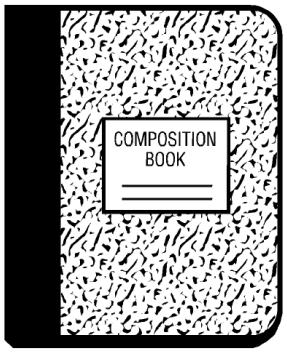
PWM Pinouts

Timer0 only!

PA1 is on the bus ☹

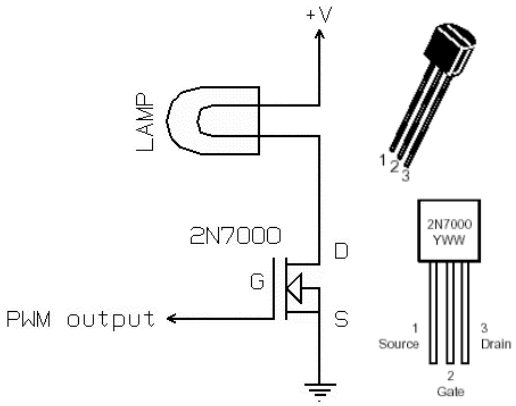
The Alternate Function sub-register described in section 5.5 connects the timer PWM outputs to port pins. Timer0 uses PA1 (J8 pin 9) and Timer1 uses PC1. You'll note that *there is no PC1* on this chip! (That information applies to larger pin count versions of the microcontroller.) So you can only use Timer0 for PWM generation with the Z8F0421. Since PA1 is wired to the trainer's bus, you can't use the I/O subroutines when the PWM alternate function is selected.

9.5 Assignment 9a: PWM Lamp Control



Program a timer on the microcontroller of your choice to generate a PWM output which cyclically ramps the duty cycle from 0% to 100% over a reasonable time period (5-30 seconds). Observe the waveform on an oscilloscope. Construct the following circuit:

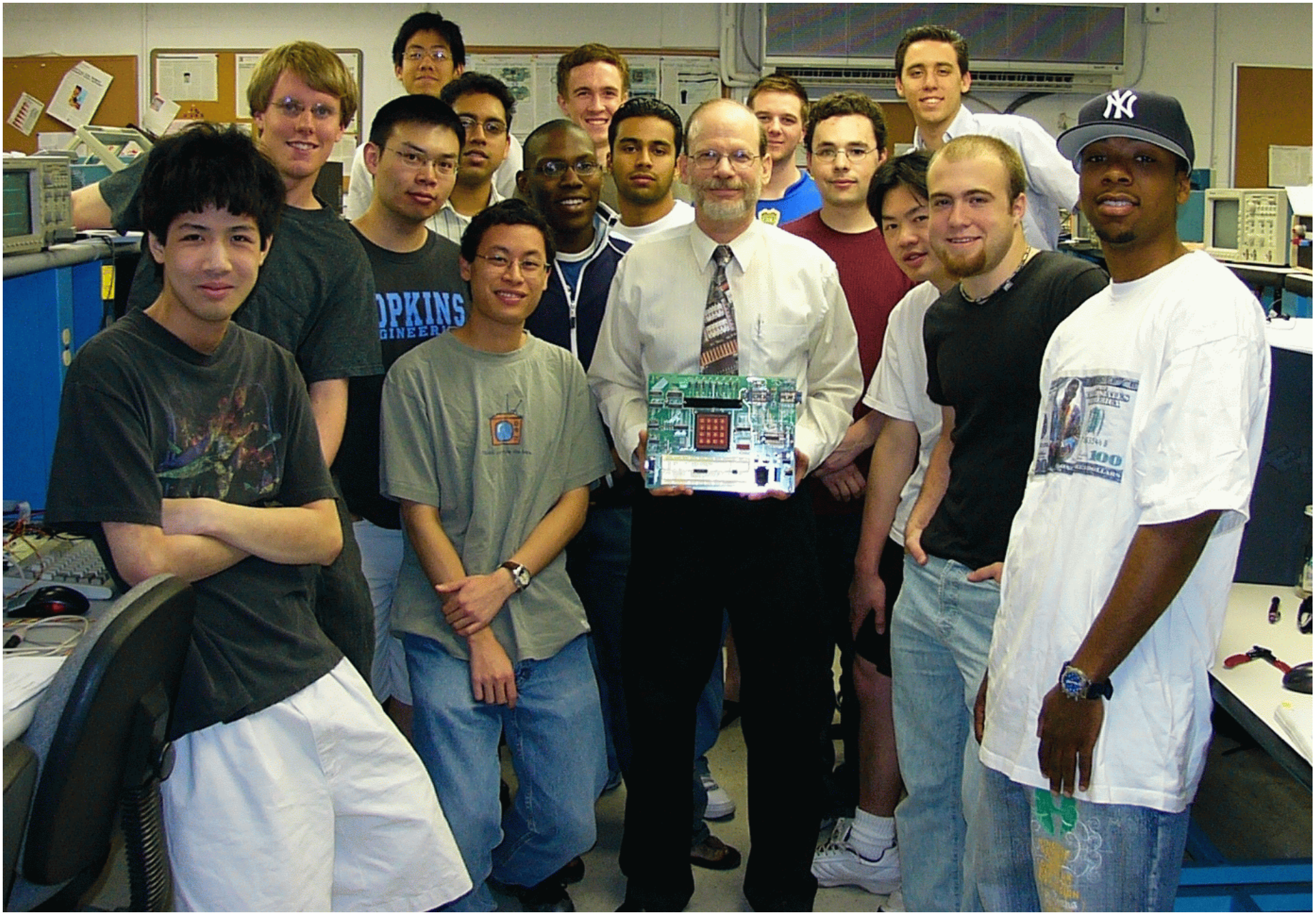
The incandescent lamp is rated at 12 volts, 60 ma.; Mouser part number 606-CM7219 is suitable. The +V on the trainer is approximately 11 volts. First, wire the schematic's PWM output to +5v and measure the drain-source voltage and the current through the lamp. Calculate the power dissipated in the switching transistor (at full load). Calculate the power efficiency of the circuit when driven with a 50% duty cycle.



Wire your microcontroller's PWM output to the transistor gate and test your program.

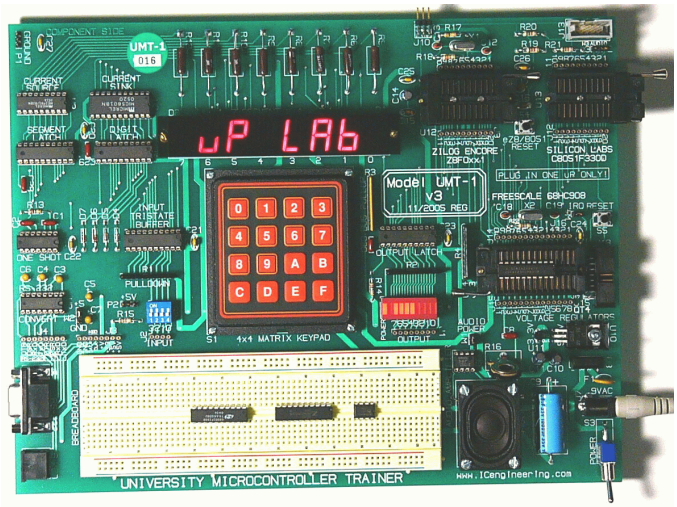
Note that with the 68HC908JK3 and the Z8F0421 you cannot simultaneously generate a PWM signal and control the trainer's I/O ports. However, you can drive the LED's from the former by temporarily discontinuing the PWM function and writing to the LED latch; and on the latter you can control the LED's 7-2 and 0 because they follow the PA port. On the C8051F330D you can generate PWM independently from trainer wiring, and you don't lose I/O capability.

Optional: ramp up/down instead of just up; display activity on LED's and displays; provide for keypad controls (pause, up, down, rate, etc.).



The author has been designing and constructing microprocessor-based systems since the 4004, and for twenty years has taught the Microprocessor Lab courses in the Electrical and Computer Engineering Department at *The Johns Hopkins University*.

Microcontroller Lab



Programming the 68HC08, 8051, and eZ8 in Assembly Language

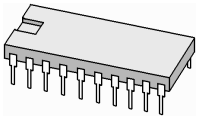
Second Edition

Robert E. Glaser

Assignments are on the:

University Microcontroller Trainer, Model UMT-1

IC Engineering, Inc.



PO Box 321
Owings Mills, MD
21117-0321 USA

www.ICengineering.com
410-363-8748

ISBN 0-9772246-1-9

90000

9 780977 224616