4
Headers, porting code – which micro?

Arizona Microchip the manufacturers of the PIC Microcontroller make over 100 different types of microcontroller. How do we choose the correct one for the job?

Factors affecting the choice of the microcontroller

When deciding on which Microcontroller to use for your application there are a number of factors you will need to consider.

- How many inputs and outputs do you need. If you are using the program FLASHER.ASM which only flashes 1 LED on and off then any PIC will do this. If you are turning 8 outputs on and off then you will need a microcontroller that has at least 8 I/O (of course). So an 8pin micro i.e. 12F629 will not do because it only has 6 I/O.
- Do you need accurate timing? If so then you will need to add a crystal to your micro to provide the clock. If timing is not that critical then you can use a micro that has an on board oscillator such as the 16F818. You can then omit the crystal and 2 capacitors. The timing accuracy is about 1%. This would do for FLASHER.ASM but not for a 24 hour clock. 1% is about 14 minutes a day.
- Are you making analogue measurements? If so you will need a micro with an AtoD converter on it. The 16F818 has a 5 channel, 10 bit AtoD converter. If you need more that 5 channels then you will need to use a micro with more AtoD channels such as the 16F877 which has 8.
- What operating frequency do you require? The greater the frequency the faster your code will execute. Most newer devices can operate up to 20MHz, some even faster. Some older devices can only achieve 4MHz. The programs in this book only require an operating speed of 4MHz.
- How many instructions are there in your program? The 16F818 has space for 1k i.e. 1024 instructions. The 16F877 has 8k program memory locations. All programs in this book require less than 1k of program memory space.
- How many memory locations are required to store data? The 16F818 has 128 bytes of data memory, the 16F877 has 368.

- Do you need to store data so that it will be saved if the power is removed or lost? If so you need a micro with EEPROM data memory. The 16F818 has 128 bytes of EEPROM memory, the 16F877 has 256.

There may be other requirements that you need from your micro, which are not considered in this book, such as:

- Number of timers
- Comparators
- Pulse width modulation
- In circuit debugging
- USB drivers.

Choosing the microcontroller

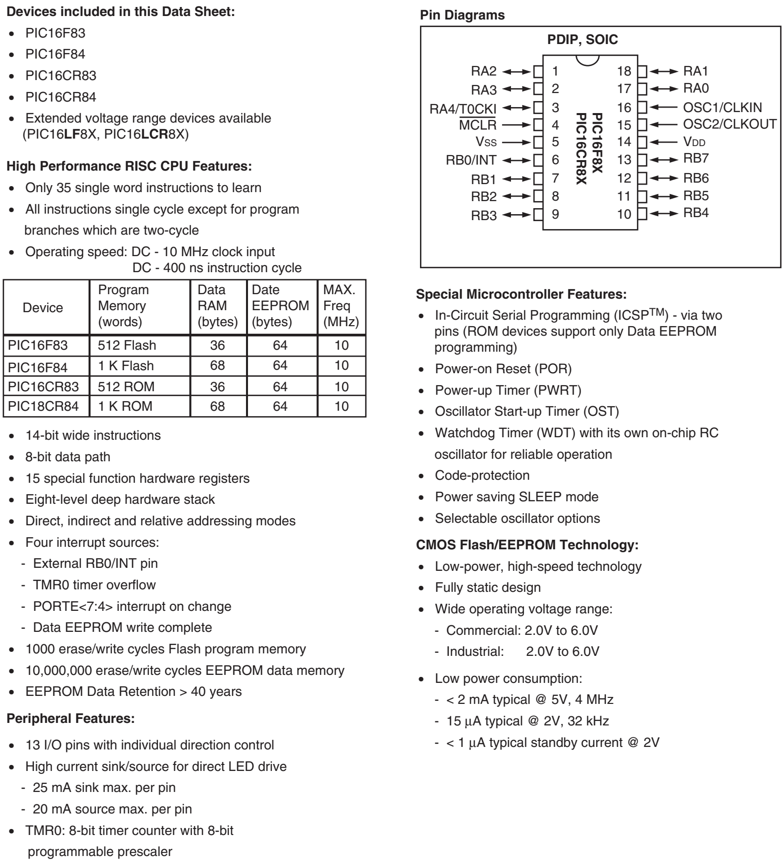
As I mentioned previously the FLASHER.ASM program which flashes 1 LED on and off can be performed by any Micro. Well, that has narrowed the field down! So which microcontroller do we use for that application? If you were mass producing these flasher units the answer would probably be – use the cheapest and smallest – the 12C508 is possibly the device then. But for small scale production or one offs you will probably have (or develop) a favorite. Probably the most common chip used by the beginner is the 16F84; this has been around since about 1998. This micro has built up a very large fan base which is why it is still widely used. People are using this chip because they are used to using it! There is now another micro on the market which will do everything that the 16F84 can do and more. This device is the 16F818.

The data sheets for the 16F84 and 16F818 are shown in Figures 4.1 and 4.2 respectively.

The main differences are that the 16F818 has 16 I/O, an on board oscillator with 8 selectable frequencies, 128 bytes of data RAM, 128 bytes of EEPROM, 3 Timers one of them a 16 bit, 5 channel 10 bit AtoD converter. The 16F84 has 13 I/O, no on board oscillator, 68 bytes of data RAM, 64 bytes of EEPROM, 1 timer, no AtoD. The most surprising difference of all is that the 16F84 is about 3 times the price of the 16F818!!

The programs in this book consist of 2 parts:

- A header section which tells the ‘build’ software which device we are using, configures the device, i.e. defines which pins are inputs and outputs, sets the timer rate and includes some timing delays if you require them in a subroutine section.



Devices included in this Data Sheet:

- PIC16F83
- PIC16F84
- PIC16CR83
- PIC16CR84
- Extended voltage range devices available (PIC16**LF**8X, PIC16**LCR**8X)

High Performance RISC CPU Features:

- Only 35 single word instructions to learn
- All instructions single cycle except for program branches which are two-cycle
- Operating speed: DC - 10 MHz clock input
  DC - 400 ns instruction cycle

Device	Program Memory (words)	Data RAM (bytes)	Date EEPROM (bytes)	MAX. Freq (MHz)
PIC16F83	512 Flash	36	64	10
PIC16F84	1 K Flash	68	64	10
PIC16CR83	512 ROM	36	64	10
PIC18CR84	1 K ROM	68	64	10

- 14-bit wide instructions
- 8-bit data path
- 15 special function hardware registers
- Eight-level deep hardware stack
- Direct, indirect and relative addressing modes
- Four interrupt sources:
  - External RB0/INT pin
  - TMR0 timer overflow
  - PORTE<7:4> interrupt on change
  - Data EEPROM write complete
- 1000 erase/write cycles Flash program memory
- 10,000,000 erase/write cycles EEPROM data memory
- EEPROM Data Retention > 40 years

Peripheral Features:

- 13 I/O pins with individual direction control
- High current sink/source for direct LED drive
  - 25 mA sink max. per pin
  - 20 mA source max. per pin
- TMR0: 8-bit timer counter with 8-bit programmable prescaler

Pin Diagrams

Special Microcontroller Features:

- In-Circuit Serial Programming (ICSP^{TM}) - via two pins (ROM devices support only Data EEPROM programming)
- Power-on Reset (POR)
- Power-up Timer (PWRT)
- Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Code-protection
- Power saving SLEEP mode
- Selectable oscillator options

CMOS Flash/EEPROM Technology:

- Low-power, high-speed technology
- Fully static design
- Wide operating voltage range:
  - Commercial: 2.0V to 6.0V
  - Industrial: 2.0V to 6.0V
- Low power consumption:
  - < 2 mA typical @ 5V, 4 MHz
  - 15 µA typical @ 2V, 32 kHz
  - < 1 µA typical standby current @ 2V

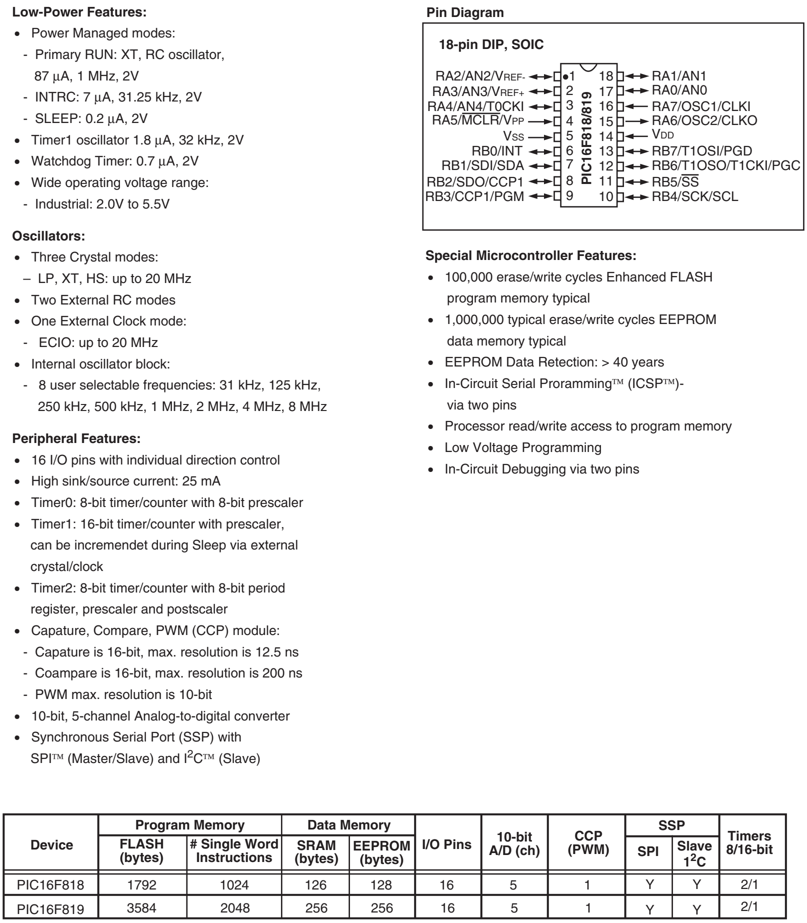
Figure 4.1 The PIC 16F84 data sheet

- The second part of the program, entitled, ‘Program starts now’, is where you write the code to perform your application.

The header program is unique to the particular microcontroller being used, but the ‘application code’ entered after “Program starts now”, is specific to the application not the microcontroller. So any microcontroller that has i.e. the required number of I/O or A/D can be used. As I mentioned before any microcontroller can be used to execute the FLASHER.ASM code.

Headers

Just one point before we look at the headers. The 8 pin micros only have 6 I/O, they do not have PORTA and PORTB pins, they have what is called a General



Low-Power Features:

- Power Managed modes:
  - Primary RUN: XT, RC oscillator, 87 µA, 1 MHz, 2V
  - INTRC: 7 µA, 31.25 kHz, 2V
  - SLEEP: 0.2 µA, 2V
- Timer1 oscillator 1.8 µA, 32 kHz, 2V
- Watchdog Timer: 0.7 µA, 2V
- Wide operating voltage range:
  - Industrial: 2.0V to 5.5V

Oscillators:

- Three Crystal modes:
  – LP, XT, HS: up to 20 MHz
- Two External RC modes
- One External Clock mode:
  - ECIO: up to 20 MHz
- Internal oscillator block:
  - 8 user selectable frequencies: 31 kHz, 125 kHz, 250 kHz, 500 kHz, 1 MHz, 2 MHz, 4 MHz, 8 MHz

Peripheral Features:

- 16 I/O pins with individual direction control
- High sink/source current: 25 mA
- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremendet during Sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Capature, Compare, PWM (CCP) module:
  - Capature is 16-bit, max. resolution is 12.5 ns
  - Coampare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit, 5-channel Analog-to-digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master/Slave) and I^{2}C™ (Slave)

Pin Diagram

Special Microcontroller Features:

- 100,000 erase/write cycles Enhanced FLASH program memory typical
- 1,000,000 typical erase/write cycles EEPROM data memory typical
- EEPROM Data Retection: > 40 years
- In-Circuit Serial Proramming™ (ICSP™)- via two pins
- Processor read/write access to program memory
- Low Voltage Programming
- In-Circuit Debugging via two pins

Device	Program Memory		Data Memory		I/O Pins	10-bit A/D (ch)	CCP (PWM)	SSP		Timers 8/16-bit
	FLASH (bytes)	# Single Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI	Slave 1^2C	
PIC16F818	1792	1024	126	128	16	5	1	Y	Y	2/1
PIC16F819	3584	2048	256	256	16	5	1	Y	Y	2/1

Figure 4.2 The PIC 16F818 and 16F819 data sheet

Purpose I/O or GPIO. So the instruction BSF PORTB,0 would have to be changed to BSF GPIO,0.

The following headers will be used in this book:

HEAD12C508.ASM	; for the 12C508 and 12C509
HEAD12F629.ASM	; for the 12F629
HEAD12F675.ASM	; for the 12F675
HEAD16F627.ASM	; for the 16F627 and 16F628
HEADER84.ASM	; for the 16F84
HEAD16F818.ASM	; for the 16F818 and 16F819
HEAD16F872.ASM	; for the 16F872, 16F874 and 16F877

```
;HEAD12C508.ASM FOR 12C508/9.
;Uses the internal 4MHz clock.

TMR0        EQU    1        ;TMR0 is FILE 1.
OSCCAL      EQU    5
GPIO        EQU    6        ;GPIO is FILE 6.
STATUS      EQU    3        ;STATUS is FILE 3.
ZEROBIT     EQU    2        ;ZEROBIT is Bit 2.
COUNT       EQU    07H      ;USER RAM LOCATION.
TIME        EQU    08H      ;TIME IS 39
;*******************************************************
LIST        P=12C508        ;We are using the 12C508.
ORG         0               ;0 is the start address.
GOTO        START           ;goto start!
;************************************************
;Configuration Bits
__CONFIG H'0FEA'            ;selects Internal RC oscillator, WDT off,
                            ;Code Protection disabled.

;*******************************************************
;SUBROUTINE SECTION.

;1/100 SECOND DELAY
DELAY      CLRF      TMR0                 ;START TMR0
LOOPA      MOVF      TMR0,W               ;READ TMR0 IN W
           SUBWF     TIME,W               ;TIME - W
           BTFSS     STATUS,ZEROBIT       ;CHECK TIME-W=0
           GOTO      LOOPA
           RETLW     0                    ;RETURN AFTER TMR0=39

;P5 SECOND DELAY
DELAYP5     MOVLW      .50
            MOVWF      COUNT
TIMEC       CALL       DELAY
            DECFSZ     COUNT
            GOTO       TIMEC
            RETLW      0

;1 SECOND DELAY
DELAYP5     MOVLW      .100
            MOVWF      COUNT
TIMED       CALL       DELAY
            DECFSZ     COUNT
            GOTO       TIMED
            RETLW      0
```

```
;*******************************************************
;CONFIGURATION SECTION.

START      MOVWF     OSCCAL
           MOVLW     B'00001000'         ;5 bits of GPIO are O/Ps.
           TRIS      GPIO
           MOVLW     B'00000111'
           OPTION                        ;PRESCALER is /256
           CLRF      GPIO                ;Clears GPIO
           MOVLW     .39
           MOVWF     TIME
*******************************************************
;Program starts now.
END
```

HEAD12F629.ASM FOR 12F629 using 4MHz internal RC

```
TMR0          EQU      1          ;TMR0 is FILE 1.
TRISIO        EQU      85H
GPIO          EQU      5          ;GPIO is FILE 6.
STATUS        EQU      3          ;STATUS is FILE 3.
ZEROBIT       EQU      2          ;ZEROBIT is Bit 2.
GO            EQU      1
OPTION_R      EQU      81H
CMCON         EQU      19H
OSCCAL        EQU      90H
COUNT         EQU      20H        ;USER RAM LOCATION.

;*******************************************************
LIST          P = 12F629          ;We are using the 12F629.
ORG           0                   ;0 is the start address.
GOTO          START               ;goto start!
;************************************************
;Configuration Bits
__CONFIG H'3F84'         ;selects Internal RC oscillator, WDT off,
                         ;Code Protection disabled.

;*******************************************************
;SUBROUTINE SECTION.

;1/100 SECOND DELAY
DELAY      CLRF      TMR0                ;START TMR0
LOOPA      MOVF      TMR0,W              ;READ TMR0 IN W
```

```
          SUBLW     .39               ;TIME - W
          BTFSS     STATUS,ZEROBIT    ;CHECK TIME-W=0
          GOTO      LOOPA
          RETLW     0                 ;RETURN AFTER TMR0=39

;P1 SECOND DELAY
DELAYP1   MOVLW     .10
          MOVWF     COUNT
TIMEC     CALL      DELAY
          DECFSZ    COUNT
          GOTO      TIMEC
          RETLW     0

;******************************************************
;CONFIGURATION SECTION.

START     BSF       STATUS,5      ;BANK1
          MOVLW     B'00001001'   ;BITS 0,3 are I/P
          MOVWF     TRISIO

          MOVLW     B'00000111'
          MOVWF     OPTION_R      ;PRESCALER is /256

          CALL      3FFH
          MOVWF     OSCCAL        ;Calibrates 4MHz oscillator

          BCF       STATUS,5      ;BANK0

          MOVLW     7H
          MOVWF     CMCON         ;Turns off comparator
          CLRF      GPIO          ;Clears GPIO
;******************************************************
;Program starts now.

END
```

;HEAD12F675.ASM FOR 12F675 using 4MHz internal RC.

```
TMR0      EQU       1             ;TMR0 is FILE 1.
TRISIO    EQU       85H
GPIO      EQU       5             ;GPIO is FILE 6.
STATUS    EQU       3             ;STATUS is FILE 3.
ZEROBIT   EQU       2             ;ZEROBIT is Bit 2.
```

```
GO          EQU         1
ADSEL       EQU         9EH
ADCON0      EQU         1FH
ADRESH      EQU         1EH
OPTION_R    EQU         81H
CMCON       EQU         19H
OSCCAL      EQU         90H
COUNT       EQU         20H             ;USER RAM LOCATION.

;**********************************************************

LIST        P=12F675    ;We are using the 12F675.
ORG         0           ;0 is the start address.
GOTO        START       ;goto start!

;***************************************************

;Configuration Bits

__CONFIG H'3F84'        ;selects Internal RC oscillator, WDT off,
                        ;Code Protection disabled.

;**********************************************************

;SUBROUTINE SECTION.

;1/100 SECOND DELAY
DELAY       CLRF        TMR0                ;START TMR0
LOOPA       MOVF        TMR0,W              ;READ TMR0 IN W
            SUBLW       .39                 ;TIME - W
            BTFSS       STATUS,ZEROBIT      ;CHECK TIME-W=0
            GOTO        LOOPA
            RETLW       0                   ;RETURN AFTER TMR0=39

;P1 SECOND DELAY
DELAYP1     MOVLW           .10
            MOVWF           COUNT
TIMEC       CALL            DELAY
            DECFSZ          COUNT
            GOTO            TIMEC
            RETLW           0
;**********************************************************
;CONFIGURATION SECTION.

START       BSF         STATUS,5        ;BANK1
            MOVLW       B'00010001'     ;A0 IS ANALOGUE,FOSC/8
            MOVWF       ADSEL
```

```
          MOVLW     B'00001001'    ;BITS 0,3 are I/P
          MOVWF     TRISIO

          MOVLW     B'00000111'
          MOVWF     OPTION_R       ;PRESCALER is /256

          CALL      3FFH
          MOVWF     OSCCAL         ;Calibrates 4MHz oscillator

          BCF       STATUS,5       ;BANK0

          MOVLW     7H
          MOVWF     CMCON          ;Turns off comparator
          CLRF      GPIO           ;Clears GPIO
          BSF       ADCON0,0       ;Turns on A/D converter.

;********************************************************
;Program starts now.
END
```

;HEAD16F627.ASM for the 16F627/8, using the 37kHz internal RC

```
;PortA bits 0 to 7 are inputs
;PortB bits 0 to 7 are outputs
;Prescaler/32

;******************************************
;EQUATES SECTION

TMR0        EQU     1
OPTION_R    EQU     1
PORTA       EQU     5
PORTB       EQU     6
TRISA       EQU     5
TRISB       EQU     6
STATUS      EQU     3
ZEROBIT     EQU     2
CARRY       EQU     0
EEADR       EQU     1BH
EEDATA      EQU     1AH
EECON1      EQU     1CH
EECON2      EQU     1DH
RD          EQU     0
WR          EQU     1
```

```
WREN            EQU           2
PCON            EQU           0EH
COUNT           EQU           20H

;**************************************************

LIST          P=16F627     ;using the 627
ORG           0
GOTO          START

;************************************************
;Configuration Bits

__CONFIG H'3F10'           ;selects Internal RC oscillator, WDT off,
                           ;Code Protection disabled.

;****************************************************
;SUBROUTINE SECTION.

;0.1 SECOND DELAY
DELAYP1    CLRF        TMR0                  ;Start TMR0
LOOPA      MOVF        TMR0,W                ;Read TMR0 into W
           SUBLW       .29                   ;TIME - W
           BTFSS       STATUS,ZEROBIT        ;Check TIME-W=0
           GOTO        LOOPA
           RETLW       0                     ;Return after TMR0=29

;0.5 SECOND DELAY
DELAYP5         MOVLW         5
                MOVWF         COUNT
LOOPB           CALL          DELAYP1       ;0.1s delay
                DECFSZ        COUNT
                GOTO          LOOPB
                RETLW         0             ;Return after 5 DELAYP1

;1 SECOND DELAY
DELAY1          MOVLW         .10
                MOVWF         COUNT
LOOPC           CALL          DELAYP1       ;0.1s delay
                DECFSZ        COUNT
                GOTO          LOOPC
                RETLW         0             ;Return after 10 DELAYP1

;*******************************************************
```

```
;CONFIGURATION SECTION.

START       BSF         STATUS,5     ;Bank1
            MOVLW       B'11111111'
            MOVWF       TRISA        ;PortA is input

            MOVLW       B'00000000'
            MOVWF       TRISB        ;PortB is output

            MOVLW       B'00000100'
            MOVWF       OPTION_R     ;Option Register, TMR0/32
            CLRF        PCON         ;Select 37kHz oscillator.
            BCF         STATUS,5     ;Bank0
            CLRF        PORTA
            CLRF        PORTB
            MOVLW       7
            MOVWF       1FH          ;CMCON turns off comparators.

;*******************************************************
;Program starts now.
END
```

;HEADER84.ASM for the 16F84 using a 32kHz crystal

```
;EQUATES SECTION

TMR0          EQU      1          ;TMR0 is FILE 1.
PORTA         EQU      5          ;PORTA is FILE 5.
PORTB         EQU      6          ;PORTB is FILE 6.
STATUS        EQU      3          ;STATUS is FILE 3.
TRISA         EQU      85H        ;TRISA (the PORTA I/O selection)
TRISB         EQU      86H        ;TRISB (the PORTB I/O selection)
OPTION_R      EQU      81H        ;the OPTION register is file 81H
ZEROBIT       EQU      2          ;ZEROBIT is Bit 2.
COUNT         EQU      0CH        ;USER RAM LOCATION.

;********************************************************

  LIST        P=16F84             ;We are using the 16F84.
  ORG         0                   ;0 is the start address.
  GOTO        START               ;goto start!

;********************************************************
```

```
;Configuration Bits

__CONFIG H'3FF0'          ;selects LP oscillator, WDT off, PUT on,
                          ;Code Protection disabled.

;*************************************************

;SUBROUTINE SECTION.

;1 SECOND DELAY
DELAY1    CLRF      TMR0                ;START TMR0
LOOPA     MOVF      TMR0,W              ;READ TMR0 IN W
          SUBLW     .32                 ;TIME - W
          BTFSS     STATUS,ZEROBIT      ;CHECK TIME-W = 0
          GOTO      LOOPA
          RETLW     0                   ;RETURN AFTER TMR0 = 32

;0.5 SECOND DELAY
DELAYP5   CLRF      TMR0                ;START TMR0
LOOPB     MOVF      TMR0,W              ;READ TMR0 IN W
          SUBLW     .16                 ;TIME - W
          BTFSS      STATUS,ZEROBIT     ;CHECK TIME-W = 0
          GOTO      LOOPB
          RETLW     0                   ;RETURN AFTER TMR0 = 16

;******************************************************
;CONFIGURATION SECTION.

START        BSF          STATUS,5        ;Turn to BANK1
             MOVLW        B'00011111'     ;5 bits of PORTA are I/Ps.
             MOVWF        TRISA
             MOVLW        B'00000000'
             MOVWF        TRISB           ;PORTB IS OUTPUT

             MOVLW        B'00000111'
             MOVWF        OPTION_R        ;PRESCALER is /256

             BCF          STATUS,5        ;Return to BANK0
             CLRF         PORTA           ;Clears PORTA
             CLRF         PORTB           ;Clears PORTB
             CLRF         COUNT
;******************************************************
;Program starts now.

END
```

```
; HEAD818.ASM for 16F818. This sets PORTA as digital INPUT.
;PORTB is an OUTPUT.
;Internal oscillator of 31.25kHz chosen
;The OPTION register is set to /256 giving timing pulses 32.768ms.
;1second and 0.5 second delays are included in the subroutine section.

;*****************************************************
;EQUATES SECTION
TMR0        EQU   1      ;means TMR0 is file 1.
STATUS      EQU   3      ;means STATUS is file 3.
PORTA       EQU   5      ;means PORTA is file 5.
PORTB       EQU   6      ;means PORTB is file 6.
ZEROBIT     EQU   2      ;means ZEROBIT is bit 2.
ADCON0      EQU   1FH    ;A/D Configuration reg.0
ADCON1      EQU   9FH    ;A/D Configuration reg.1
ADRES       EQU   1EH    ;A/D Result register.
CARRY       EQU   0      ;CARRY IS BIT 0.
TRISA       EQU   85H    ;PORTA Configuration Register
TRISB       EQU   86H    ;PORTB Configuration Register
OPTION_R    EQU   81H    ;Option Register
OSCCON      EQU   8FH    ;Oscillator control register.
COUNT       EQU   20H    ;COUNT a register to count events.
;*****************************************************

    LIST      P=16F818    ;we are using the 16F818.
    ORG       0           ;the start address in memory is 0
    GOTO      START       ;goto start!

;*****************************************************
;Configuration Bits

__CONFIG H'3F10'    ;sets INTRC-A6 is port I/O, WDT off, PUT
                    ;on, MCLR tied to VDD A5 is I/O
                    ;BOD off, LVP disabled, EE protect disabled,
                    ;Flash Program Write disabled,
                    ;Background Debugger Mode disabled,
                    ;CCP function on B2,
                    ;Code Protection disabled.

;*************************************************
;SUBROUTINE SECTION.
;0.1 second delay, actually 0.099968s
DELAYP1   CLRF    TMR0            ;START TMR0.
LOOPB     MOVF    TMR0,W          ;READ TMR0 INTO W.
```

```
            SUBLW     .3                  ;TIME-3
            BTFSS     STATUS,ZEROBIT      ;Check TIME-W = 0
            GOTO       LOOPB              ;Time is not = 3.
            NOP                           ;add extra delay
            NOP
            RETLW     0                   ;Time is 3, return.

;0.5 second delay.
DELAYP5      MOVLW         .5
             MOVWF         COUNT
LOOPC        CALL          DELAYP1
             DECFSZ        COUNT
             GOTO          LOOPC
             RETLW         0

;1 second delay.
DELAY1       MOVLW         .10
             MOVWF         COUNT
LOOPA        CALL          DELAYP1
             DECFSZ        COUNT
             GOTO          LOOPA
             RETLW         0
;*********************************************************

;Configuration Section

START        BSF           STATUS,5        ;Turns to Bank1.

             MOVLW         B'11111111'     ;8 bits of PORTA are I/P
             MOVWF         TRISA

             MOVLW         B'00000110'     ;PORTA IS DIGITAL
             MOVWF         ADCON1

             MOVLW         B'00000000'
             MOVWF         TRISB           ;PORTB is OUTPUT

             MOVLW         B'00000000'
             MOVWF         OSCCON          ;oscillator 31.25kHz

             MOVLW         B'00000111'     ;Prescaler is /256
             MOVWF         OPTION_R        ;TIMER is 1/32 secs.
            BCF         STATUS,5     ;Return to Bank0.
            CLRF        PORTA        ;Clears PortA.
            CLRF        PORTB        ;Clears PortB.

;*****************************************************
;Program starts now.
END
```

;HEAD872.ASM Header for 16F872 using 32kHz oscillator

```
;EQUATES SECTION

        TMR0        EQU        1
        OPTION_R    EQU        1
        PORTA       EQU        5
        PORTB       EQU        6
        PORTC       EQU        7
        TRISA       EQU        5
        TRISB       EQU        6
        TRISC       EQU        7
        STATUS      EQU        3
        ZEROBIT     EQU        2
        CARRY       EQU        0
        EEADR       EQU        0DH
        EEDATA      EQU        0CH
        EECON1      EQU        0CH
        EECON2      EQU        0DH
        RD          EQU        0
        WR          EQU        1
        WREN        EQU        2
        ADCON0      EQU        1FH
        ADCON1      EQU        1FH
        ADRES       EQU        1EH
        CHS0        EQU        3
        GODONE      EQU        2
        COUNT       EQU        20H
;**************************************************
        LIST        P=16F872
        ORG         0
        GOTO        START
;****************************************************
;Configuration Bits
__CONFIG H'3F30'                  ;selects LP oscillator, WDT off, PUT on,
                                  ;Code Protection disabled.
```

```
;**************************************************

;SUBROUTINE SECTION.

;1 SECOND DELAY
DELAY1      CLRF        TMR0                  ;Start TMR0
LOOPA       MOVF        TMR0,W                ;Read TMR0 into W
            SUBLW       .32                   ;TIME - W
            BTFSS        STATUS,ZEROBIT       ;Check TIME-W = 0
            GOTO        LOOPA
            RETLW       0                     ;Return after TMR0 = 32

;0.5 SECOND DELAY
DELAYP5     CLRF        TMR0                  ;Start TMR0
LOOPB       MOVF        TMR0,W                ;Read TMR0 into W
            SUBLW       .16                   ;TIME - W
            BTFSS       STATUS,ZEROBIT        ;Check TIME-W = 0
            GOTO        LOOPB
            RETLW       0                     ;Return after TMR0 = 16

;*******************************************************
;CONFIGURATION SECTION.

START      BSF              STATUS,5        ;Bank1
           MOVLW            B’11111111’
           MOVWF            TRISA           ;PortA is input

           MOVLW            B’00000000’
           MOVWF            TRISB           ;PortB is output

           MOVLW            B’11111111’
           MOVWF            TRISC           ;PortC is input

           MOVLW            B’00000111’
           MOVWF            OPTION_R        ;Option Register, TMR0/256

           MOVLW            B’00000000’
           MOVWF            ADCON1          ;PortA bits 0,1,2,3,5 are analogue
           BSF              STATUS,6        ;BANK3
           BCF              EECON1,7        ;Data memory on.
           BCF              STATUS,5
           BCF              STATUS,6        ;BANK0 return
           BSF              ADCON0,0        ;turn on A/D
           CLRF             PORTA
```

```
            CLRF            PORTB
            CLRF            PORTC

;*********************************************************
;Program starts now.
END
```

These headers can be used for applications that use the corresponding microcontrollers. E.g. Any one of them can be used with FLASHER.ASM. Other applications may require functions that are not in all of the devices i.e. AtoD.

The explanation of the operation of the headers will be dealt with later when the individual micros are examined.