MECE336– Microprocessors I

Lecture 5 – Clock and Delays

Associate Prof. Dr. Klaus Werner Schmidt

Department of Mechatronics Engineering – Çankaya University

Compulsory Course in Mechatronics Engineering
Credits (3/2/4)

Course Webpage: http://MECE336.cankaya.edu.tr

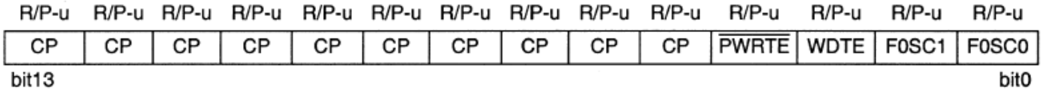
Clock Oscillator: Basics

Facts

- Clock source determines fundamental operating characteristics of a microcontroller
  - Operation speed
  - Power consumption
  - Accurate timing operation
- Fast clock provides fast operating speed and program execution
- Using a fast clock consumes more power and causes more electromagnetic interference
- Simple oscillators can be used if accurate timing is not required
- Stable and accurate clock oscillator must be used if accurate timing is required

Clock Oscillator: Configurations

Flag `FOSC1` and `FOSC0` in the Configuration Word



R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u
CP	CP	CP	CP	CP	CP	CP	CP	CP	CP	$\overline{\text{PWRTE}}$	WDTE	F0SC1	F0SC0
bit13													bit0

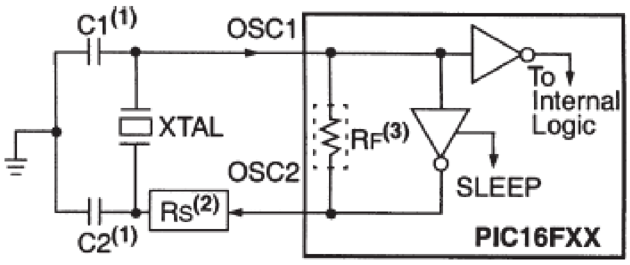
	`FOSC1`	`FOSC0`
RC	1	1
HS	1	0
XT	0	1
LP	0	0

Oscillator Modes

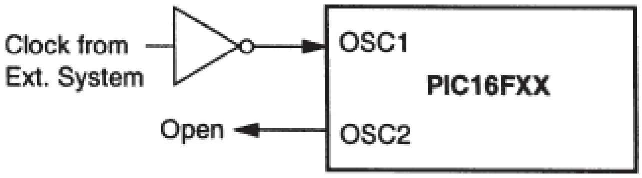
- XT – crystal: intended for crystals and resonators with 1-4 MHz
- HS – high speed: higher drive version of XT for crystals above 4 MHz or ceramic resonators. Such oscillators require more drive current.
- LP – low power: low-frequency crystals below around 200 kHz. Lowest possible power consumption.
- RC – resistor-capacitor: lowest-cost version but with less timing accuracy.

Clock Oscillator: Configurations

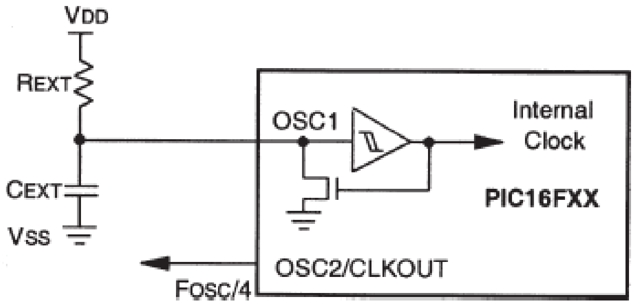
Crystal or Ceramic (HS, XT, LP)



Externally Supplied Clock (HS, XT, LP)

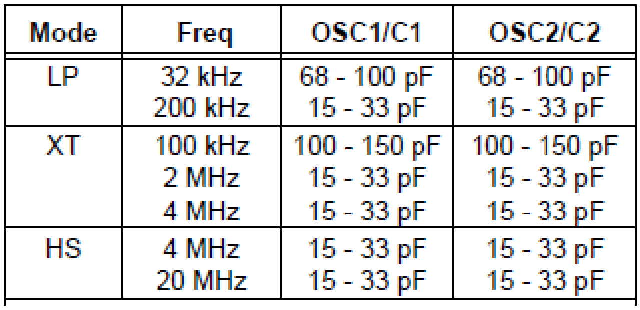


Resistor-capacitor (RC)



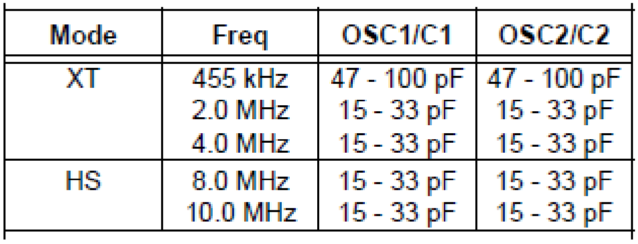
Clock Oscillator: Component Selection Tables

Crystal



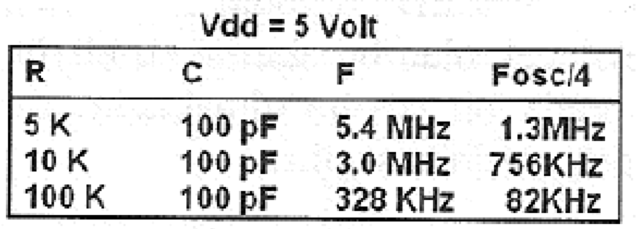
Mode	Freq	OSC1/C1	OSC2/C2
LP	32 kHz 200 kHz	68 - 100 pF 15 - 33 pF	68 - 100 pF 15 - 33 pF
XT	100 kHz 2 MHz 4 MHz	100 - 150 pF 15 - 33 pF 15 - 33 pF	100 - 150 pF 15 - 33 pF 15 - 33 pF
HS	4 MHz 20 MHz	15 - 33 pF 15 - 33 pF	15 - 33 pF 15 - 33 pF

Ceramic



Mode	Freq	OSC1/C1	OSC2/C2
XT	455 kHz 2.0 MHz 4.0 MHz	47 - 100 pF 15 - 33 pF 15 - 33 pF	47 - 100 pF 15 - 33 pF 15 - 33 pF
HS	8.0 MHz 10.0 MHz	15 - 33 pF 15 - 33 pF	15 - 33 pF 15 - 33 pF

Resistor-capacitor



Vdd = 5 Volt

R	C	F	Fosc/4
5 K	100 pF	5.4 MHz	1.3MHz
10 K	100 pF	3.0 MHz	756KHz
100 K	100 pF	328 KHz	82KHz

Power Supply: General

Supply Voltage Levels

- Traditionally 5 V due to TTL (transistor-transistor logic) technology
- Recent devices use 3.3 V and 3.0 V for reduced power consumption

Operating Condition of the PIC 16F84A

- According to data sheet: supply voltage between 4.0 and 5.5 V (suitable for three AA cells)
- At least 4.5 V in HS oscillator mode
- Drop down to 1.5 V without loosing data in RAM in **sleep mode**

Supply Current

- About 1.8 mA when running at 4 MHz with supply voltage 5.5 V
- About 10 mA when running at 20 MHz with supply voltage 5.5 V
- Low power consumptions for low-power device PIC 16LF84A: 15 μA

Reset: Power-up

Reset State at Power-up

- Program counter is zero
- Special function registers are in safe and disabled sate

→ CPU starts running when leaving the Reset state

Reset Input of PIC 16F84A

- Master Clear $\overline{\mathbf{MCLR}}$ (pin 4)
- Microcontroller is in Reset state as long as $\overline{\mathbf{MCLR}}$ is low
- Microcontroller starts program execution when $\overline{\mathbf{MCLR}}$ is high

Stable Power-up

- Program execution should wait until all parts of the circuitry stabilize after start-up
- Different circuits for safe start-up

Reset: Power-up

On-Chip Power-up Timer

- Enable if bit 3 $\overline{\mathbf{PWRTE}}$ of configuration word is cleared
- Holds microcontroller in reset during power-up

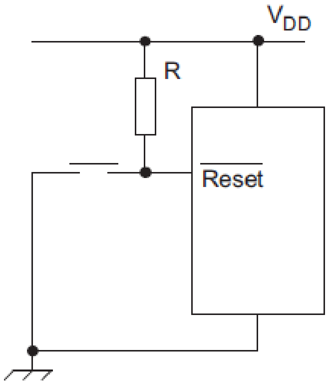
Power-on Reset

- Start-up depends on rise time of resistor/capacitor circuit

Safe Start-up

- Connect $\overline{\mathbf{MCLR}}$ to Vdd
- If external reset is desired, add User Reset Button

User Reset Button



Delay: Basic Information

Instruction Cycle

- Basic operational process of a CPU by which the CPU
  - retrieves a program instruction from its memory
  - determines what actions the instruction dictates
  - carries out those actions
- Instruction cycle frequency is (oscillator frequency)/4
- Time duration of one instruction cycle: 1/(instruction cycle frequency) $\rightarrow$ Oscillator frequency 4 MHz $\Rightarrow$ instruction cycle frequency 1 MHz $\Rightarrow$ instruction cycle 1 μs
- On PIC 16F84A, most instructions take 1 instruction cycle
- Exceptions (see instruction set)
  - GOTO, CALL, RETURN, RETFIE RETLW (2 instruction cycles)
  - DECFSZ, INCFSZ, BTFSC, BTFSS (2 instruction cycles if skip, 1 instruction cycle otherwise)

Delay: Computation

Basic Delay Loop

```
counter  equ     0x0C
         movlw   D'200'
         movwf   counter
loop     nop
         nop
         decfsz  counter,1
         goto    loop
         end
```

Explanation

Result

- 199 iterations with 5 instruction cycles (including GOTO)
- 1 iteration with 4 instruction cycles (no GOTO in last iteration)
- Total: $199 \cdot 5 + 4 = 999$ instruction cycles from `loop` to `end`
- Assume clock frequency 4 MHz $\rightarrow$ delay is $999 \cdot 1\mu s = 999\,\mu s$

Delay: Example

Example

- Turn on/off PORTB with a frequency of 1 kHz

Delay: Example (Ctnd)

Delay: Different Oscillator Frequencies

Same Delay Loop as Above

Notes

```
counter  equ     0x0C
         movlw   D'200'
         movwf   counter
loop     nop
         nop
         decfsz  counter,1
         goto    loop
         end
```

Result

- Total: $199 \cdot 5 + 4 = 999$ instruction cycles from `loop` to `end`
- Assume clock frequency 4 MHz → delay is $999 \cdot 1\mu\text{s} = 999\,\mu\text{s}$
- Assume clock frequency 20 MHz → delay is $999 \cdot 0.2\,\mu\text{s} = 199.8\,\mu\text{s}$
- Assume clock frequency 100 kHz → delay is $999 \cdot 40\,\mu\text{s} = 39.96\,\text{ms}$

→ A different delay loop has to be used for each oscillator frequency

---

Delay: General Formulation of a Single Delay Loop

General Delay Loop

Notes

```
counter  equ     counterAddress
nIt      equ     N
         movlw   nIt
         movwf   counter
loop     nop
          ⋮
         nop
         decfsz  counter,1
         goto    loop
         nop
         end
```

Result

- Assume loop contains k `nop` instructions and oscillator frequency f
  $\Rightarrow (k+3) \cdot (N-1) + k + 2 + 1 = (k+3) \cdot N$ instruction cycles
  $\Rightarrow$ Delay of $(k+3) \cdot N \cdot 4/f$ between `loop` and `end`

Delay: Example Computations

Examples

- Realize a delay of 1 ms for an oscillator frequency of $f = 20\,\mathrm{MHz}$
- Realize a delay of 1 ms for an oscillator frequency of $f = 100\,\mathrm{kHz}$

Delay: Example Computations

Examples

- Realize a delay of 100 ms for an oscillator frequency of $f = 100\,\mathrm{kHz}$
- Realize a delay of 100 ms for an oscillator frequency of $f = 20\,\mathrm{MHz}$

Conclusion

- Large delays cannot be realized in a single loop if f is large
  $\Rightarrow$ Use cascaded delay loops

Cascaded Delay Loops: Overview

Variable Definitions

```
list      p=16f84a
include   "p16f84a.inc"

org       0
counter1  equ     counterAddress1
counter2  equ     counterAddress2
nIt1      equ     N_1
nIt2      equ     N_2
movlw     nIt1
movwf     counter1

loop1     ... (next slide)
```

Explanation

Cascaded Delay Loops: Overview

Loops

```
loop1     nop
          ⋮
          nop
          movlw     nIt2
          movwf     counter2
loop2     nop
          ⋮
          nop
          decfsz  counter2,1
          goto    loop2
          nop
          decfsz  counter1,1
          goto    loop1
          nop
          end
```

Explanation

Cascaded Delay Loops: Evaluation

Computation

- Assume there are k_1 `nop` instructions in loop 1
- Assume there are k_2 `nop` instructions in loop 2
- Assume the oscillator frequency is f
- Number of instruction cycles of inner loop as before using N_2 and k_2
  $C_2 = (k_2 + 3) \cdot N_2$ instruction cycles in loop 2
- Number of instruction cycles of outer loop
  $C_1 = (k_1 + C_2 + 5) \cdot N_1$ instruction cycles in loop 1
- Overall delay:

$$C_1 \cdot \frac{4}{f} = (k_1 + C_2 + 5) \cdot N_1 \cdot \frac{4}{f} = (k_1 + (k_2 + 3) \cdot N_2 + 5) \cdot N_1 \cdot \frac{4}{f}$$

Cascaded Delay Loops: Example

Example

- Let $f = 4\,\text{MHz}$. Determine suitable k_1, k_2, N_1, N_2 for $100\,\text{ms}$ delay
- Option 1

- Option 2

Cascaded Delay Loops: Example

Example

- Let $f = 125\,\text{kHz}$. Determine suitable k_1, k_2, N_1, N_2 for $1.6\,\text{s}$ delay
- Option 1
- Option 2

Cascaded Delay Loops: Program for 1.6 s Delay