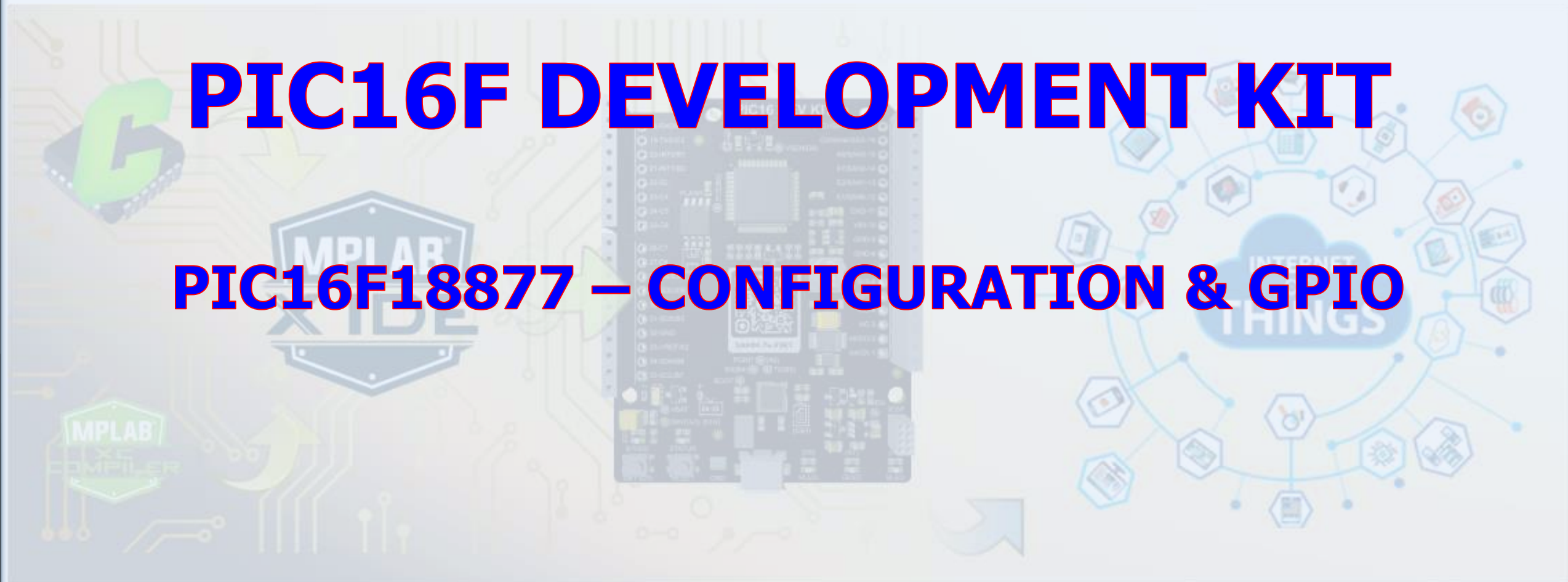


PIC16F DEVELOPMENT KIT

PIC16F18877 – CONFIGURATION & GPIO





NỘI DUNG

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT



GIỚI THIỆU CHUNG

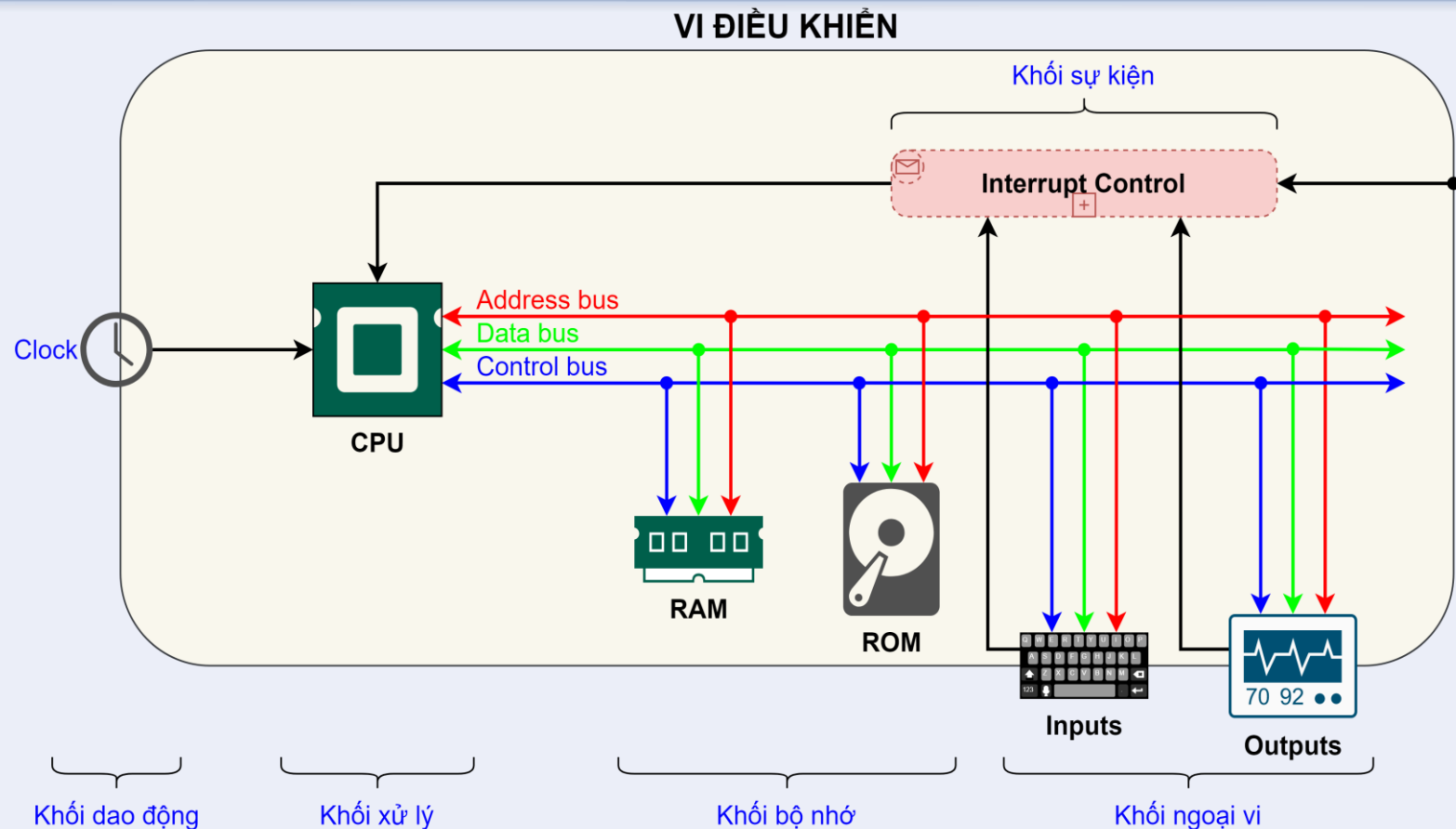
1. HỆ THỐNG VI ĐIỀU KHIỂN

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

1. Khối dao động
2. Khối xử lý
3. Khối bộ nhớ
4. Khối ngoại vi
5. Khối sự kiện
6. Các khối khác:
Quản lý nguồn,
Gỡ lỗi, ...





GIỚI THIỆU CHUNG

2. CÔNG CỤ PHÁT TRIỂN

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

1. MPLAB X IDE/ PIC C IDE

Môi trường soạn thảo, định dạng, gỡ lỗi, ...

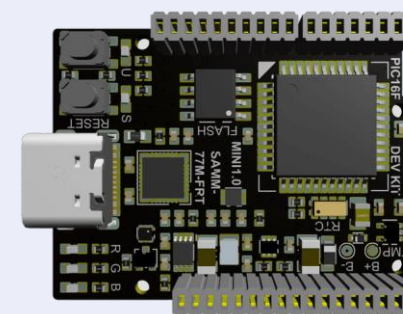


2. MPLAB XC8/ CCS C

Chuyển đổi ngôn ngữ C/Asm sang mã thực thi.



3. Evaluation Kit





GIỚI THIỆU CHUNG

3. TÀI LIỆU KỸ THUẬT

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

1. Datasheet: [DS40001825F](#)
2. Programming Specifications: [DS40001753B](#)
3. Application Notes:
<https://www.microchip.com/en-us/application-notes>
4. Code Examples:
<https://www.microchip.com/doclisting/CodeExamplesByFunc.aspx>



CONFIGURATION

1. CẤU TRÚC BỘ NHỚ

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

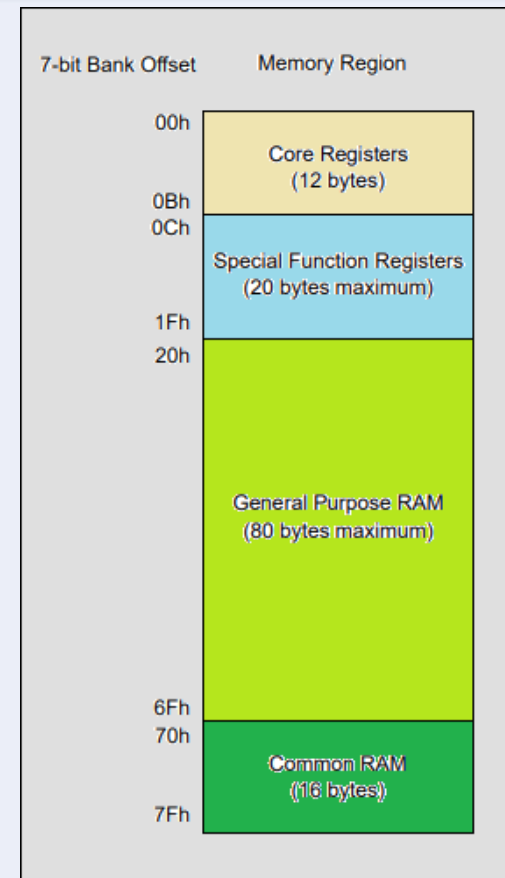
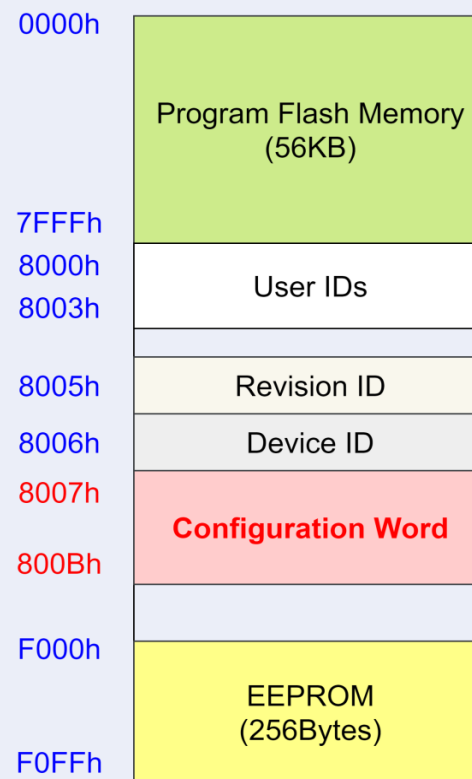
Để làm việc với vi điều khiển cần:

- ☐ Cấu hình được chip.
- ☐ Quản lý được bộ nhớ.
- ☐ Quản lý được ngoại vi.

Xem thêm:

→ [DS40001753B](#) – trang 3.

→ [DS40001825F](#) – trang 39.





CONFIGURATION

2. CẤU HÌNH CHIP

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

❑ Quy định các tính năng cụ thể của chip, bao gồm: dao động, quản lý lỗi, quản lý nguồn, tính năng bảo mật, ...

❑ Các bit cấu hình là cố định, không thay đổi trong quá trình hoạt động.

❑ Vùng nhớ cấu hình chip có độ dài 5 word (5x2byte):

Word 1: Oscillator

Word 2: Supervisors

Word 3: Watchdog

Word 4: Memory

Word 5: Code protection

0000h

Program Flash Memory
(56KB)

7FFFh

8000h

User IDs

8003h

8005h

Revision ID

8006h

Device ID

8007h

Configuration Word

800Bh

F000h

EEPROM
(256Bytes)

F0FFh



CONFIGURATION

2. CẤU HÌNH CHIP

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

```
3 // CONFIG1
4 #pragma config FEXTOSC = OFF // External Oscillator mode selection bits (Oscillator not enabled)
5 #pragma config RSTOSC = HFINT32 // Power-up default value for COSC bits (HFINTOSC with OSCFRQ= 32 MHz and CDIV = 1:1)
6 #pragma config CLKOUTEN = OFF // Clock Out Enable bit (CLKOUT function is disabled; i/o or oscillator function on OSC2)
7 #pragma config CSWEN = ON // Clock Switch Enable bit (Writing to NOSC and NDIV is allowed)
8 #pragma config FCMEN = ON // Fail-Safe Clock Monitor Enable bit (FSCM timer enabled)
9 // CONFIG2
10 #pragma config MCLRE = ON // Master Clear Enable bit (MCLR pin is Master Clear function)
11 #pragma config PWRT = OFF // Power-up Timer Enable bit (PWRT disabled)
12 #pragma config LPBORN = OFF // Low-Power BOR enable bit (ULPBOR disabled)
13 #pragma config BOREN = ON // Brown-out reset enable bits (Brown-out Reset Enabled, SBORN bit is ignored)
14 #pragma config BORV = LO // Brown-out Reset Voltage Selection (Brown-out Reset Voltage (VBOR) set to 1.9V on LF, and 2.45V on F Devices)
15 #pragma config ZCD = OFF // Zero-cross detect disable (Zero-cross detect circuit is disabled at POR.)
16 #pragma config PPS1WAY = ON // Peripheral Pin Select one-way control (The PPSLOCK bit can be cleared and set only once in software)
17 #pragma config STVREN = ON // Stack Overflow/Underflow Reset Enable bit (Stack Overflow or Underflow will cause a reset)
18 // CONFIG3
19 #pragma config WDTCP = WDTCP_31 // WDT Period Select bits (Divider ratio 1:65536; software control of WDTPS)
20 #pragma config WDTE = OFF // WDT operating mode (WDT Disabled, SWDTEN is ignored)
21 #pragma config WDTCS = WDTCS_7 // WDT Window Select bits (window always open (100%); software control; keyed access not required)
22 #pragma config WDTCCS = SC // WDT input clock selector (Software Control)
23 // CONFIG4
24 #pragma config WRT = OFF // UserNVM self-write protection bits (Write protection off)
25 #pragma config SCANE = available // Scanner Enable bit (Scanner module is available for use)
26 #pragma config LVP = ON // Low Voltage Programming Enable bit (Low Voltage programming enabled. MCLR/Vpp pin function is MCLR.)
27 // CONFIG5
28 #pragma config CP = OFF // UserNVM Program memory code protection bit (Program Memory code protection disabled)
29 #pragma config CPD = OFF // DataNVM code protection bit (Data EEPROM code protection disabled)
```




CONFIGURATION

3. CẤU ĐẠO ĐỘNG

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

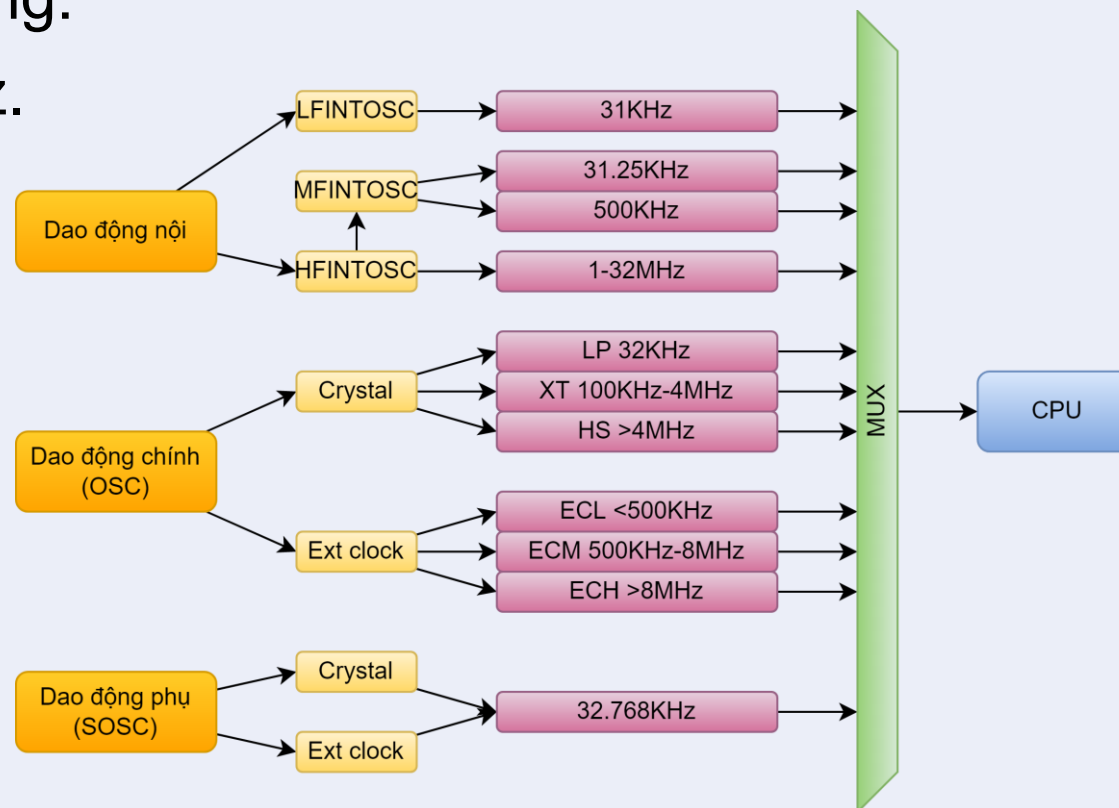
❑ PIC16F18877 có 2 nhóm dao động:

➤ Dao động nội: 31KHz đến 32MHz.

➤ Dao động ngoài:

- Clock chính OSC: lên đến 32MHz.
- Clock phụ SOSC: 32768Hz.

❑ Tốc độ xử lý: $F_{CPU} = F_{OSC} / 4$.





CONFIGURATION

3. CẤU ĐẠO ĐỘNG

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Cấu hình trong
configure bits

Cấu hình trong
software

```
// NOSC HFINTOSC
```

```
// NDIV 1
```

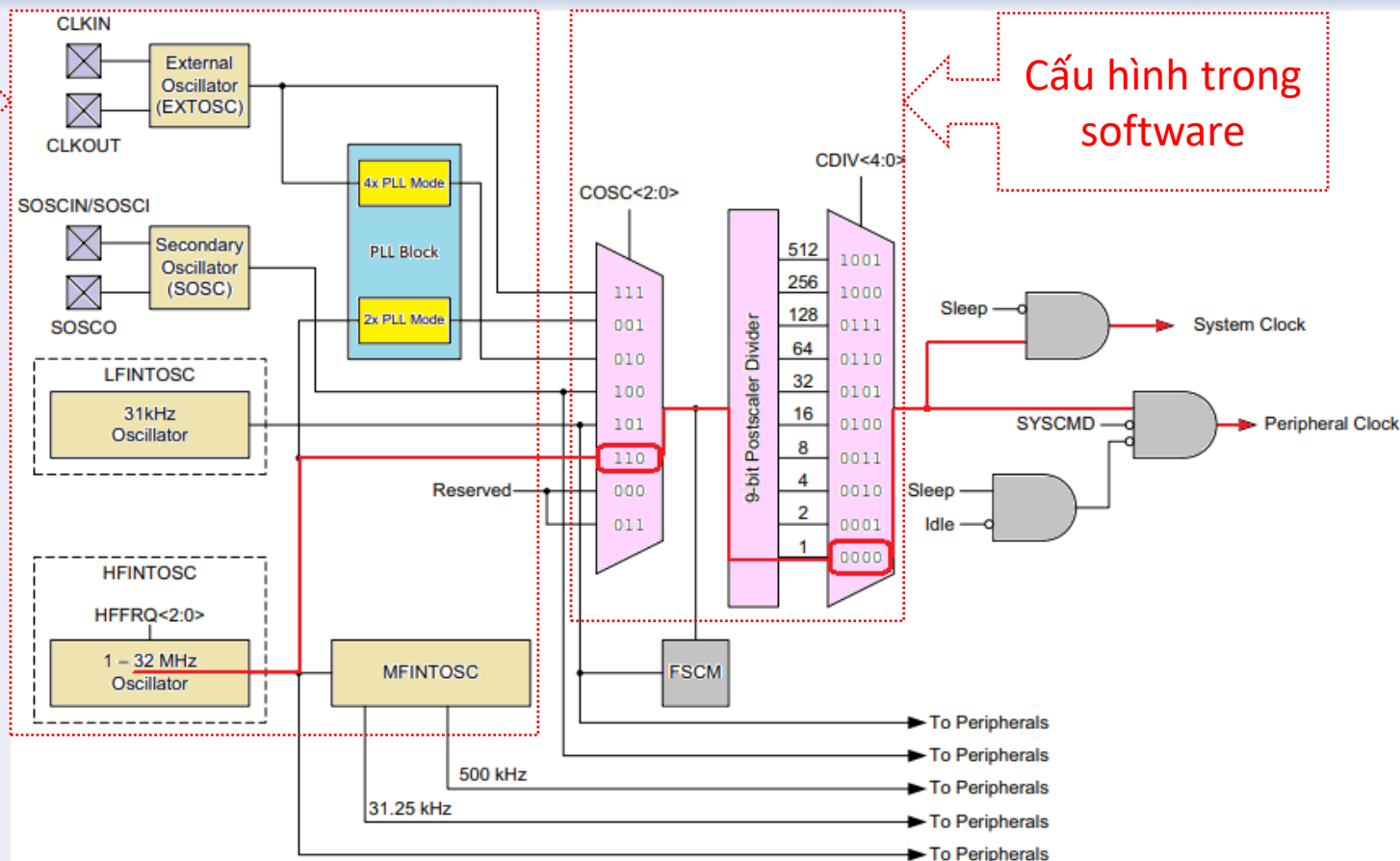
```
OSCCON1=0b01100000;
```

[DS40001825F](#) – trang 121.

```
// HFFRQ 32_MHz
```

```
OSCFRQ=0b00000110;
```

[DS40001825F](#) – trang 125.





GPIO

1. I/O PORTS

GIỚI THIỆU CHUNG

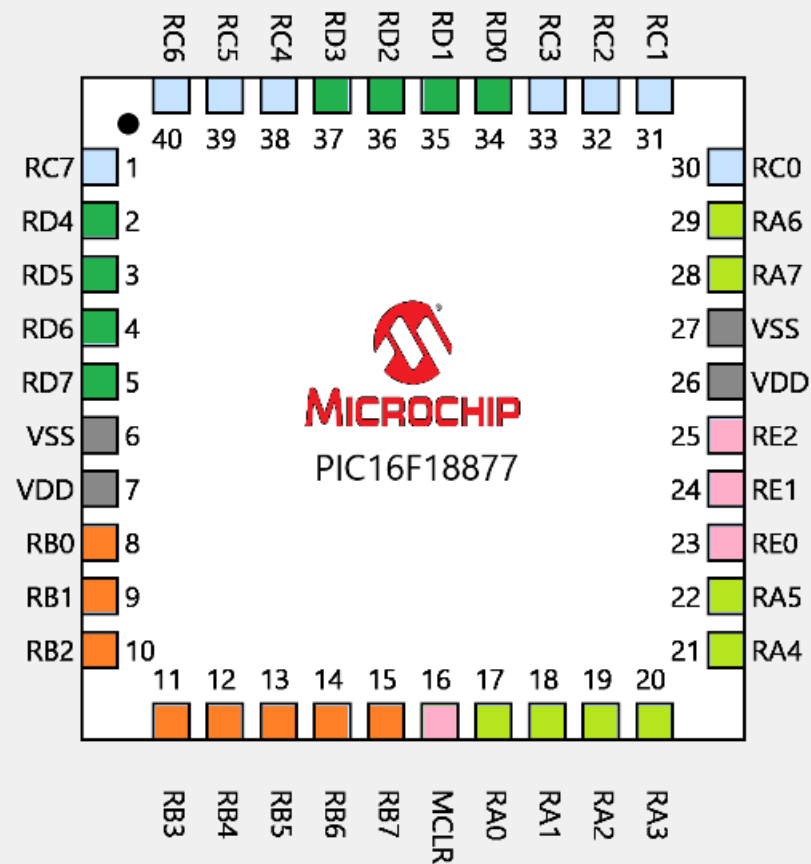
CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

❑ PIC16F18877 có 5 port:

Port A, B, C, D: 8 I/O.

Port E: 4 I/O. Có 1 I/O được mặc định chức năng MCLR.





GPIO

1. I/O PORTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

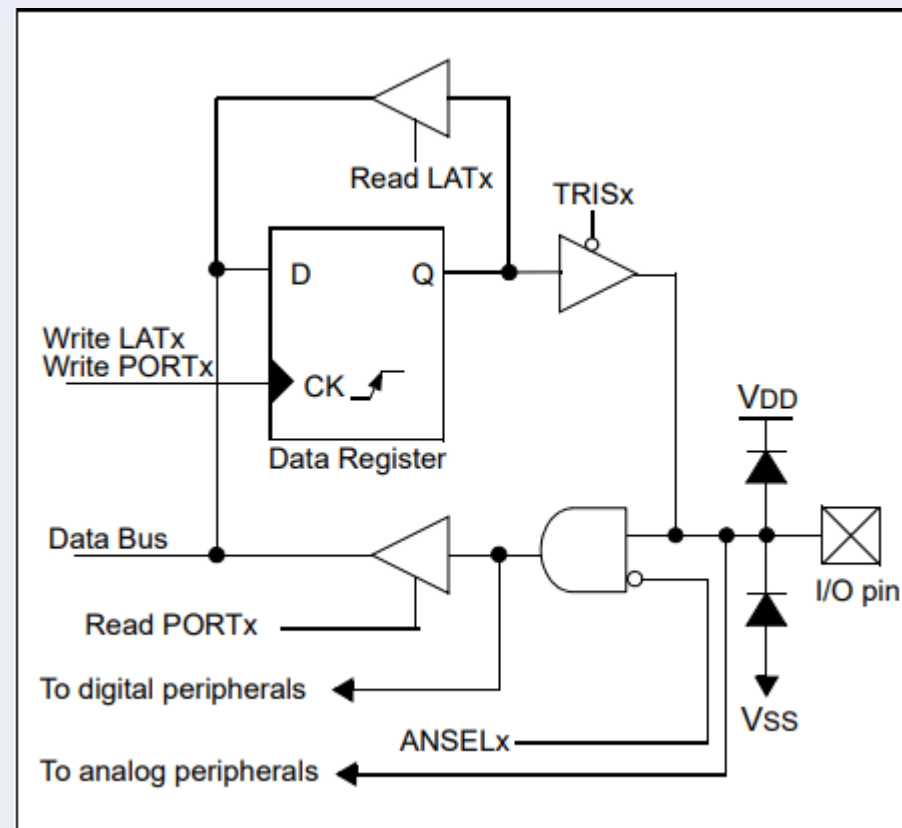
❑ Mỗi port được điều khiển bởi các thanh ghi sau với mỗi bit tương ứng 1 pin:

TRISx: Qui định ngõ vào/ra. Pin là ngõ vào nếu bit tương ứng là 1.

ANSELx: Qui định dạng tín hiệu là số (0) hoặc tương tự (1).

LATx: Qui định trạng thái ngõ ra.

PORTx: Lưu trạng thái ngõ vào.





GPIO

1. I/O PORTS

GIỚI THIỆU CHUNG

CONFIGURATION

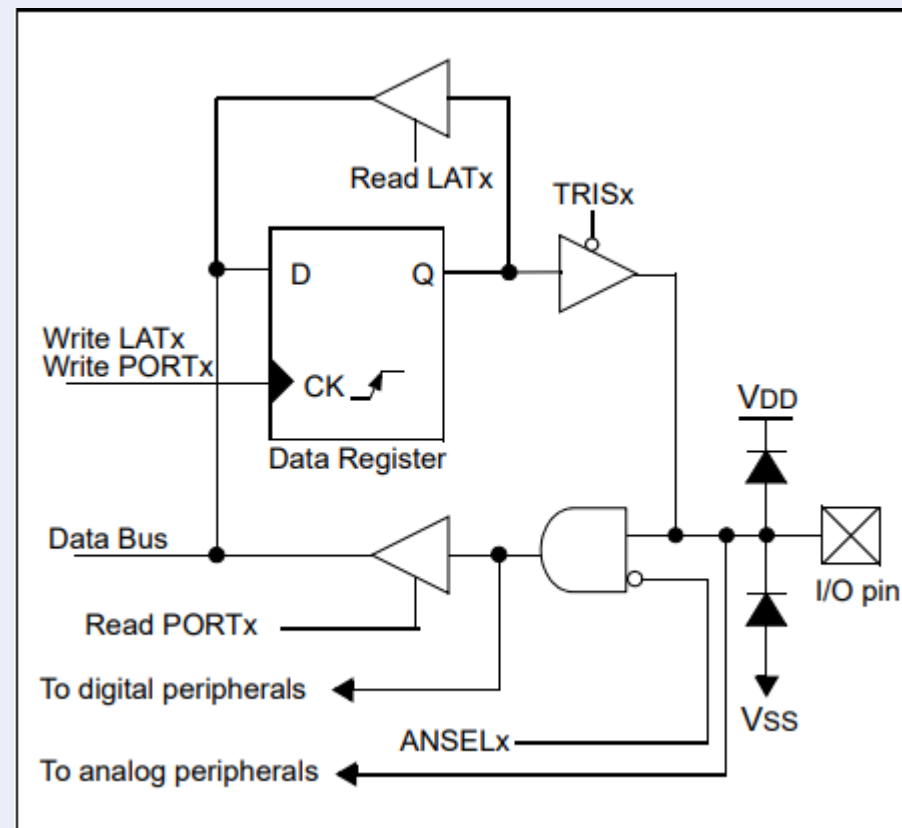
GENERAL-PURPOSE INPUT/OUTPUT

WPUx: Cấu hình điện trở kéo lên.

INLVLx: Qui định chuẩn logic TTL hoặc CMOS. (Mặc định)

SLRCONx: Qui định tốc độ thay đổi trạng thái. (Mặc định)

ODCONx: Cấu hình ngõ ra cực nguồn hở.





GPIO

2. DIGITAL OUTPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Qui trình cấu hình:

- ✓ Cấu hình tín hiệu số: **`ANSELxbits.ANSxy=0;`**
- ✓ Cấu hình ngõ ra: **`TRISxbits.TRISxy=0;`**
- ✓ Cấu hình cực nguồn: **`ODCONxbits.ODCxy=0;`** // Push-Pull
`ODCONxbits.ODCxy=1;` // Open-drain
- ✓ Đặt trạng thái:
`LATxbits.LATxy=0;` // Set low
`LATxbits.LATxy=1;` // Set high



GPIO

2. DIGITAL OUTPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Ví dụ:

```
1  PIC16F18877 Configuration Bit Settings
36
37  #include <xc.h>
38
39  void main(void)
40  {
41      // Cau hình dao dong
42      OSCCON1=0b01100000; // NOSC HFINTOSC; NDIV 1;
43      OSCFRQ=0b00000110; // HFFRQ 32_MHz;
44      // GPIO
45      ANSELbits.ANSA1=0;
46      TRISAbits.TRISA1=0;
47      LATAbits.LATA1=0;
48
49      while(1)
50      {
51          LATAbits.LATA1=~LATAbits.LATA1;
52          __delay_ms(100);
53      }
54  }
```



GPIO

3. DIGITAL INPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Qui trình cấu hình:

- ✓ Cấu hình tín hiệu số: **`ANSELxbits.ANSxy=0;`**
- ✓ Cấu hình ngõ vào: **`TRISxbits.TRISxy=1;`**
- ✓ Cấu hình điện trở kéo: **`WPUAbits.WPUxy=0; // No pull-up`
`WPUAbits.WPUxy=1; // Pull-up`**
- ✓ Đọc trạng thái: **`bool x=PORTxbits.Rxy;`**



GPIO

3. DIGITAL INPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Ví dụ:

```
1  PIC16F18877 Configuration Bit Settings
36
37  #include <xc.h>
38
39  void main(void)
40  {
41      // Cấu hình dao động
42      OSCCON1=0b01100000; // NOSC HFINTOSC; NDIV 1;
43      OSCFRQ=0b00000110; // HFFRQ 32_MHz;
44      // GPIO
45      ANSELbits.ANSA1=0;
46      TRISAbits.TRISA1=0;
47      LATAbits.LATA1=0;
48
49      ANSELDbits.ANSD3=0;
50      TRISDbits.TRISD3=1;
51      WPUDbits.WPUD3=1;
52
53      while(1)
54      {
55          LATAbits.LATA1=~PORTDbits.RD3;
56      }
57  }
```



GPIO

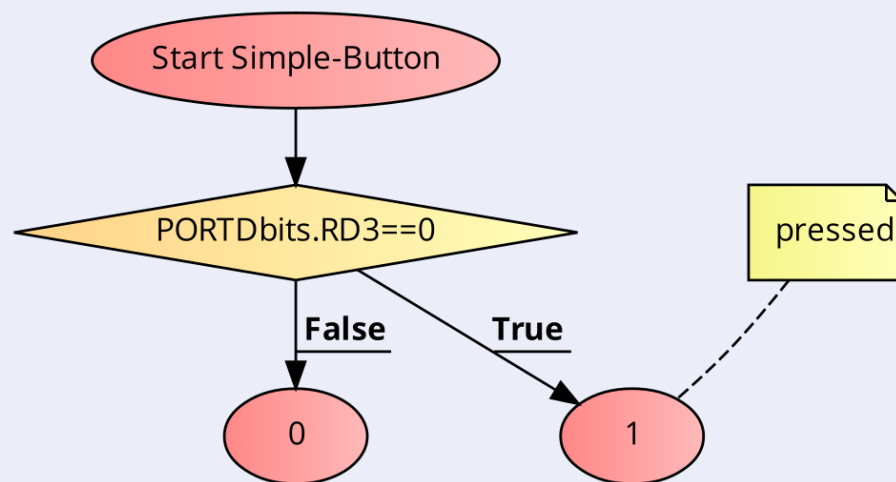
3. DIGITAL INPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Simple button:





GPIO

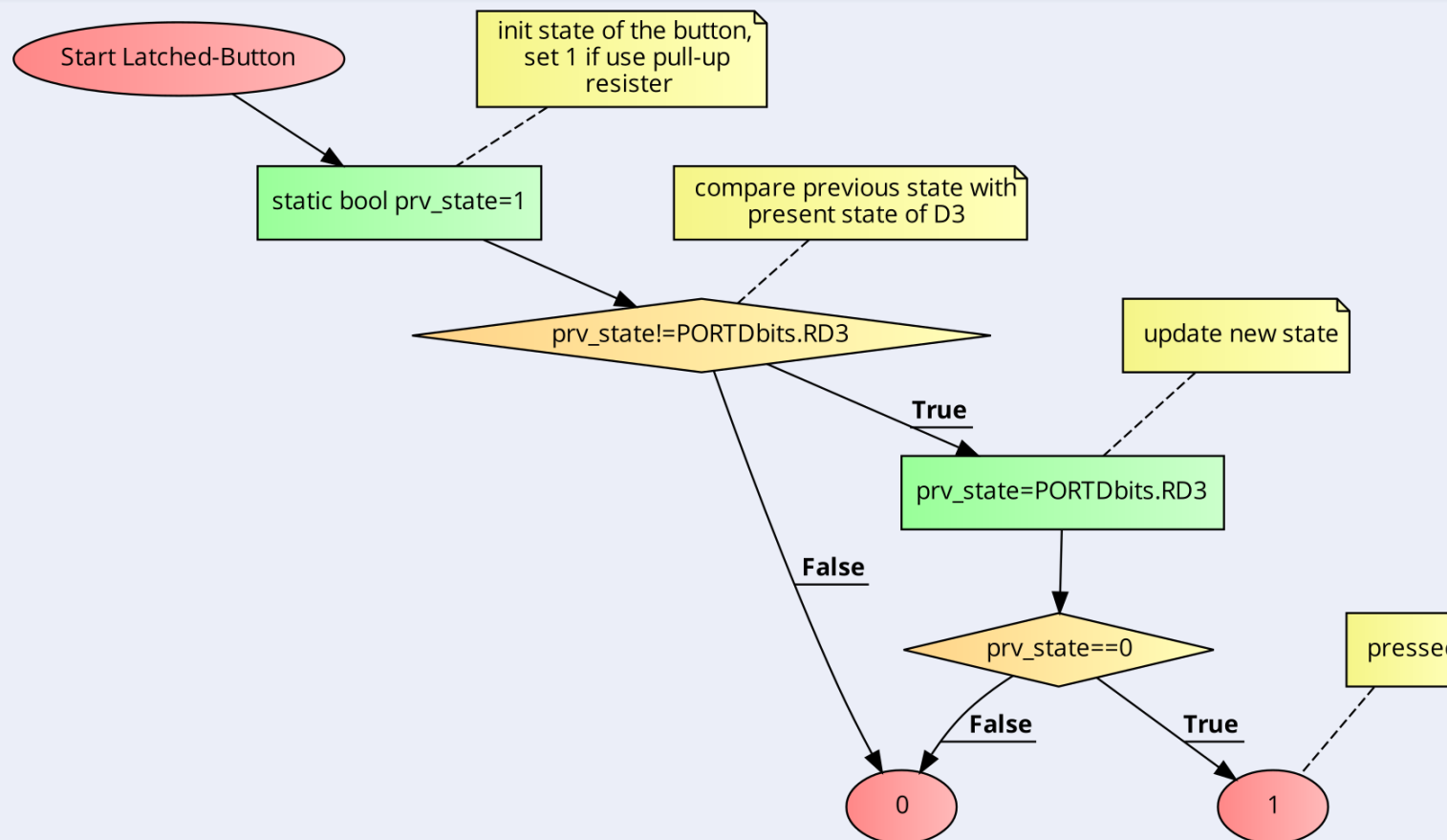
3. DIGITAL INPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Latched button:





GPIO

3. DIGITAL INPUTS

GIỚI THIỆU CHUNG

CONFIGURATION

GENERAL-PURPOSE INPUT/OUTPUT

Debounce button:

