Quick guide to change the internal oscillator frequency
Preparado por Ing. Procopio Villarreal
Microcontrollers (TE-2015)
ITESM Campus Monterrey:

Typically the PIC18F45K50 is configured by default to use the internal RC oscillator. If in your application you are not using an external crystal and wants to change the oscillator frequency a value different than the default then you must consider the following.

First, read section 3 from the data sheet for a more detailed explanation. This is just a brief getting started guide but the Oscillator sections is quite complex and also you can take advantage of the features if you know this module in detail
http://ww1.microchip.com/downloads/en/DeviceDoc/30000684B.pdf

Second, the internal high frequency oscillator can have an error of about +/-5% max in the entire temperature ant this is dependent of temperature. So is important that you consider this, if you need more precision, then you will need to use an external crystal.
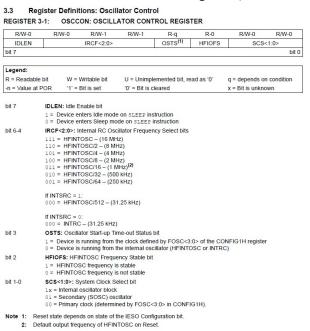
HF-INTOSC Accuracy[1]					
	-2	±1	+2	%	+0°C to +70°C
	-3	—	+2	%	+70°C to +85°C
	-5	—	+5	%	-40°C to 85°C

There is a way to reduce the drift using what is called Active Clock Tuning, however this method is out of the scope of the guide but you can dig further in the data sheet to see how this is performed.

HF-INTOSC Accuracy with Active Clock Tuning (ACT)					
	-0.20	±0.05	+0.20	%	-40°C to +85°C[2], Active Clock Tune is enabled and locked.

Using an external crystal will have much lower drifts in the +/- 10ppm (parts per million) or even better. This is mentioned in Subject 7 presentation.

Ok but let's to our matter, how can I change the frequency of the HF-INTOSC of the clock? The key lies on the first oscillator control register (there are two)

3.3   Register Definitions: Oscillator Control

REGISTER 3-1:   OSCCON: OSCILLATOR CONTROL REGISTER

R/W-0	R/W-0	R/W-1	R/W-1	R-q	R-0	R/W-0	R/W-0
IDLEN	IRCF<2:0>			OSTS[1]	HFIOFS	SCS<1:0>	
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	q = depends on condition
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7   **IDLEN:** Idle Enable bit
1 = Device enters Idle mode on SLEEP instruction
0 = Device enters Sleep mode on SLEEP instruction

bit 6-4   **IRCF<2:0>:** Internal RC Oscillator Frequency Select bits
111 = HFINTOSC – (16 MHz)
110 = HFINTOSC/2 – (8 MHz)
101 = HFINTOSC/4 – (4 MHz)
100 = HFINTOSC/8 – (2 MHz)
011 = HFINTOSC/16 – (1 MHz)[2]
010 = HFINTOSC/32 – (500 kHz)
001 = HFINTOSC/64 – (250 kHz)

If INTSRC = 1:
000 = HFINTOSC/512 – (31.25 kHz)

If INTSRC = 0:
000 = INTRC – (31.25 kHz)

bit 3   **OSTS:** Oscillator Start-up Time-out Status bit
1 = Device is running from the clock defined by FOSC<3:0> of the CONFIG1H register
0 = Device is running from the internal oscillator (HFINTOSC or INTRC)

bit 2   **HFIOFS:** HFINTOSC Frequency Stable bit
1 = HFINTOSC frequency is stable
0 = HFINTOSC frequency is not stable

bit 1-0   **SCS<1:0>:** System Clock Select bit
1x = Internal oscillator block
01 = Secondary (SOSC) oscillator
00 = Primary clock (determined by FOSC<3:0> in CONFIG1H).

Note 1:   Reset state depends on state of the IESO Configuration bit.
     2:   Default output frequency of HFINTOSC on Reset.

What you need to do is to change the IRCF bits to the value you want.

bit 6-4   **IRCF<2:0>:** Internal RC Oscillator Frequency Select bits

```
111 = HFINTOSC – (16 MHz)
110 = HFINTOSC/2 – (8 MHz)
101 = HFINTOSC/4 – (4 MHz)
100 = HFINTOSC/8 – (2 MHz)
011 = HFINTOSC/16 – (1 MHz)(2)
010 = HFINTOSC/32 – (500 kHz)
001 = HFINTOSC/64 – (250 kHz)
```

For example, in the following code I am changing the value of the Fosc to 16 Mhz:

```
//FUNCTION TO INIT THE OSCILLATOR AT 16 MHZ
void init_osc(void){

    OSCCON = 0b01110010;


}
```

The MSB is don care for our application
The Green bits define the frequency
The Blue bits are read only
The Red bit tell the microcontroller to use the internal oscillator, so even if the configuration bits (See Subject 13) set in the PIC flash say to use the crystal, the microcontroller will witch to the internal when the function is executed. So if you want to use an external crystal then the Red bits must be equal to 00.