SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

COIMBATORE-35

Accredited by NBA-AICTE and Accredited by NAAC – UGC with A++ Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

COURSE NAME: 19EEB303 / Microcontroller and its Applications

III YEAR / VI SEMESTER

Unit II – **PIC MICROCONTROLLER**

Topic Watchdog Timer

Watchdog Timer

- Watchdog Timer (WDT) can be helpful to automatically reset the system whenever a timeout occurs.
- A system reset is required for preventing the failure of the system in a situation of a hardware fault or program error.
- There are countless applications where the system cannot afford to get stuck at a point (not even for a small duration of time). For example, in a radar system, if the system hangs for 5 minutes, it can result in serious repercussions (an enemy plane or missile may go undetected resulting in huge losses).
- The system should be robust enough to automatically detect the failures quickly and reset itself in order to recover from the failures and function normally without errors.
- One can manually reset the system to recover from errors. But it is not always feasible to manually reset the system, especially once it has been deployed.
- To overcome such problems, a watchdog timer is necessary to automatically reset the system without human intervention.

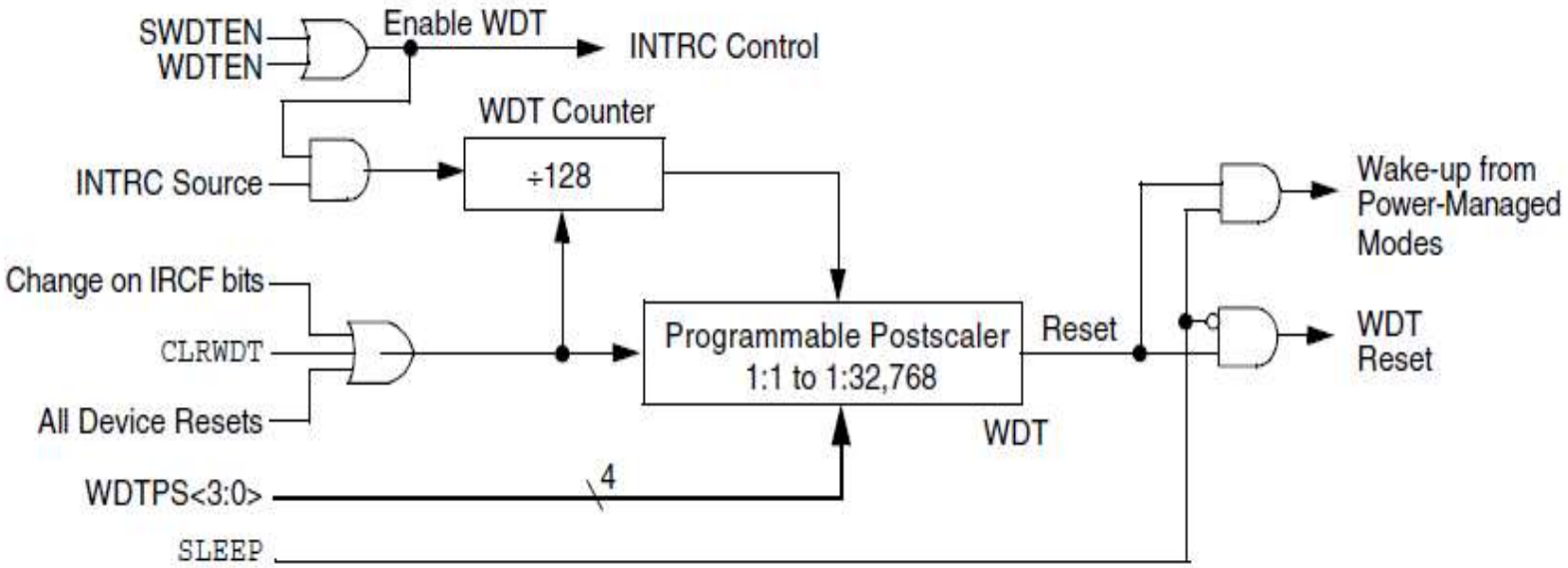
Working of Watchdog Timer

- The watchdog timer is loaded with a timeout period which is dependent on the application.
- The watchdog timer starts its counting independent of a system clock i.e. it has a separate internal oscillator to work independently of a system clock.
- The watchdog timer cleared through software each time before the timeout period occurs.
- Whenever software failed to clear the watchdog timer before its timeout period, the watchdog timer resets the system.
- For this purpose, a watchdog timer is used to overcome software failures in real-time applications.
- The watchdog timer is also used to wake up the microcontroller from sleep mode.

Watchdog Timer

- Inside of PIC18F4550 Watchdog Timer
- In PIC18F4550, the watchdog timer uses a different 31 kHz INTRC clock and it is independent of a system clock.
- Watchdog Timer can be enabled in two ways through Configuration Register (CONFIG2H) and through WDTCON Register.
- CONFIG2H has a WDTEN bit to enable/disable the watchdog timer.
- WDTCON (WDT control register) has the SWDTEN bit which is used to enable/disable the WDT through software.

Block Diagram of Watchdog Timer



Watchdog Timer

- When WDT is enabled, 31 kHz INTRC source gets initialized and provides a clock for the watchdog timer.
- This clock is then divided by 128 (pre-scaler). This pre-scaler gives a nominal time-out period of 4 ms.
- PIC18F4550 also has a programmable Post-scaler which helps to divide down the WDT pre-scaler output and increase the time-out periods. So now we can vary the time-out period in the range of 4ms to 131.072 sec (2.18 min) using Post-scaler.

Watchdog Timer

Enabling and Disabling WDT

There are two ways to enable or disable the WDT which are given as follows:

1. **Through Configuration Register:**

CONFIG2H Register: Configuration Register 2 High

Bit 0 – WDTEN: Watchdog Timer Enable bit

0 = Disables WDT (possible to enable WDT through SWDTEN)

1 = Enables WDT

7	6	5	4	3	2	1	0
			WDTPS3	WDTPS2	WDTPS1	WDTPS0	WDTEN

Watchdog Timer

Through WDTCON Register:

WDTCON Register: Watchdog Timer Control Register

Bit 0 – SWDTEN: Software Controlled Watchdog Timer Enable bit

0 = Disable Watchdog Timer

1 = Enable Watchdog Timer

This software controlled watchdog timer can enable watchdog timer only if configuration bit has disabled the WDT.

Watchdog Timer

calculate the WDT Timeout Period

WDT Timeout Period Calculation

E.g.

If we configure Post-scaler **WDTPS3:WDTPS0 = 1001** i.e. **512** is Post-scaler then,

31 kHz/128 = 242.1875

Now,

Out = 242.1875/512 = 0.473

Time-out period = 1/0.473 = 2.11 sec