Fire Bird V P89V51RD2

Motion Control using Pulse Width Modulation

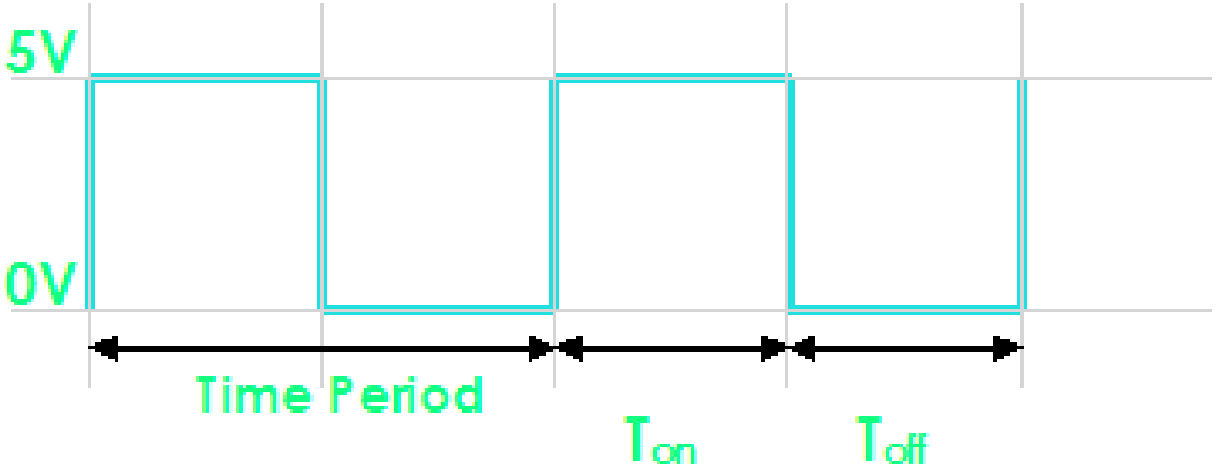
ERTS LAB
CSE IIT BOMBAY

By Pawankumar Suryawanshi

Pulse Width Modulation

- Pulse Width Modulation (PWM), is a method of transmitting information on a series of pulses.
- The data that is being transmitted is encoded on the width of these pulses to control the amount of power being sent to a load.
- Examples: Electric stoves, Lamp dimmers, and Robotic Servos.

Pulse Width Modulation



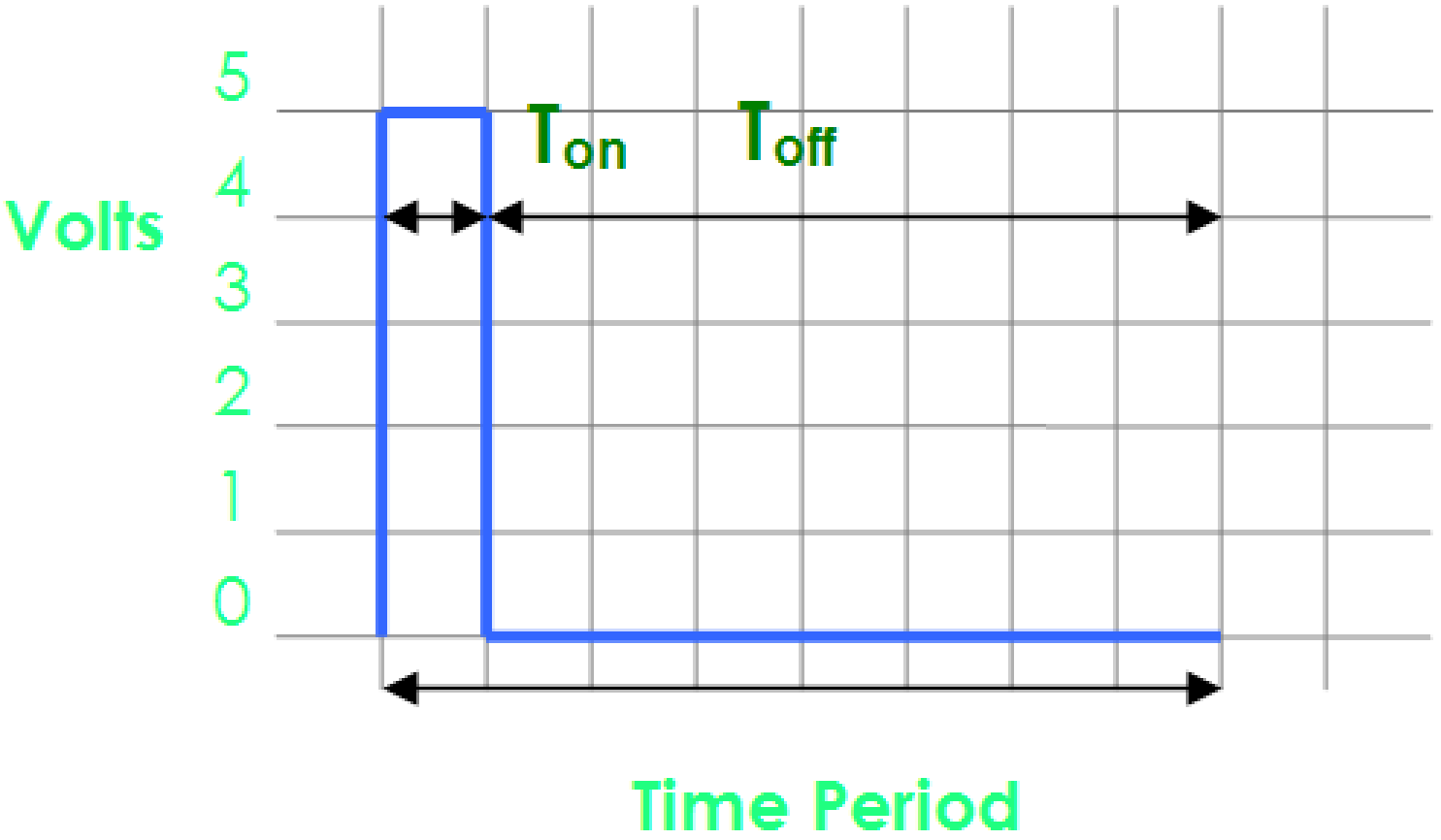
- The signal remains "ON" for some time and "OFF" for some time.
- Ton = Time the output remains high.
- Toff = Time the output remains Low.
- When output is high the voltage is 5v
- When output is low the voltage is 0v

T = Time Period = Ton + Toff

Duty Cycle= $\frac{T_{on}}{\text{Time Period}} \times 100$

Duty Cycle = 50%
Analog Voltage Out = 50% of

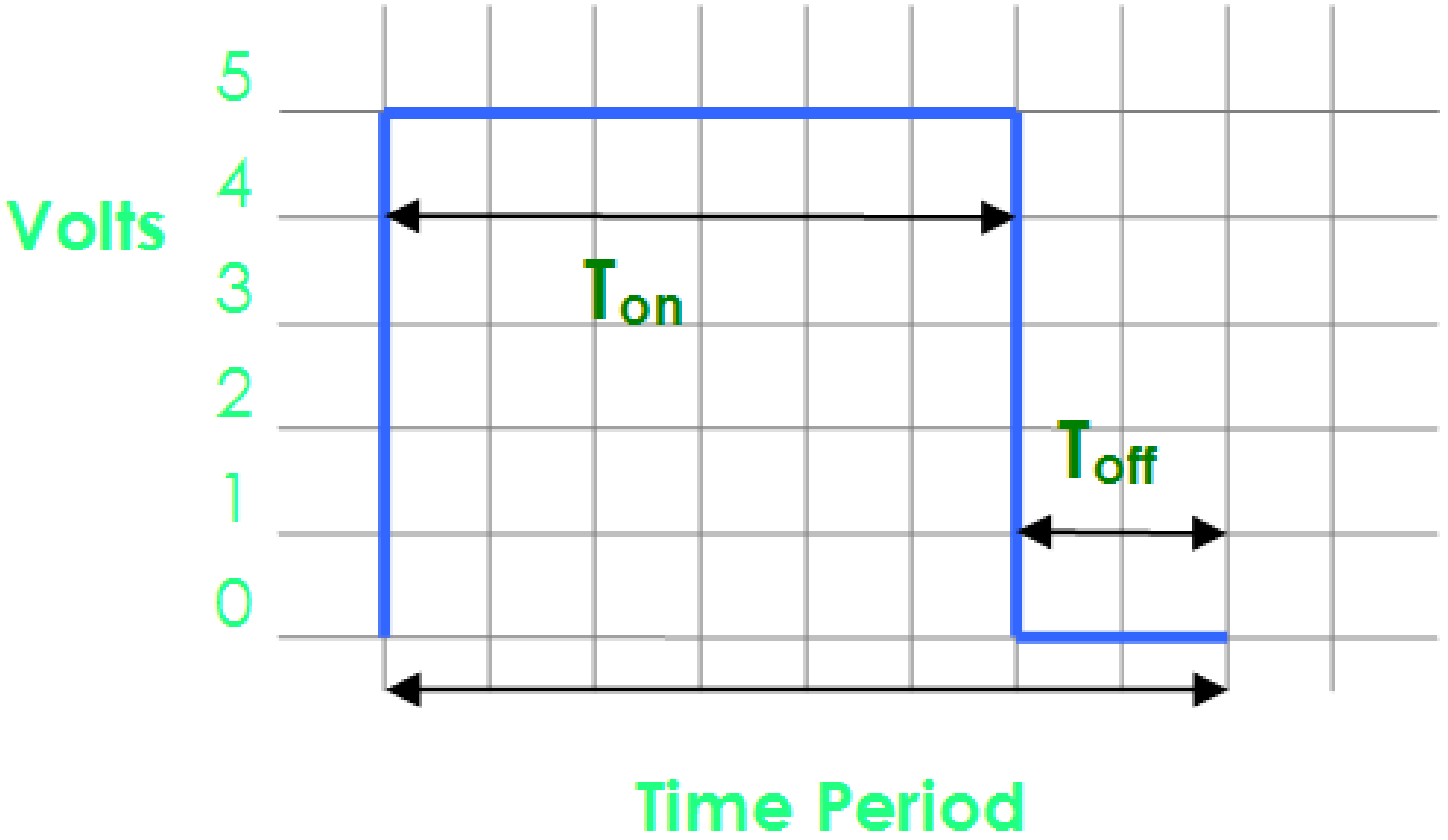
Pulse Width Modulation



Duty Cycle = 12.5%

Analog Voltage Out = 12.5% of

Pulse Width Modulation



Duty Cycle = 75%

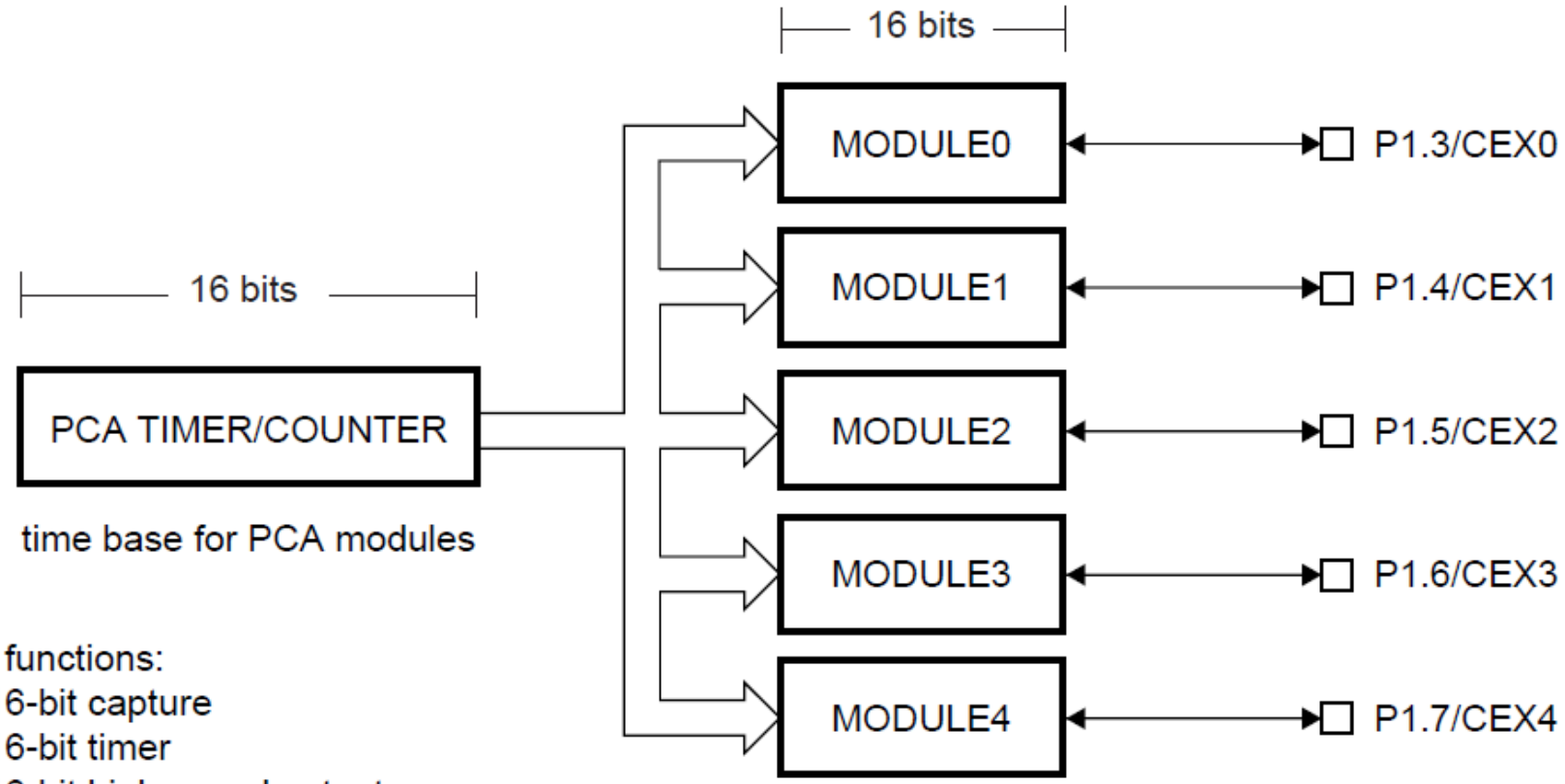
Analog Voltage Out = 75% of

Programmable Counter Array

- Programmable Counter Array (PCA) provides enhanced timer functionality.
- In P89V51RD2 µC has PCA which includes a special 16-bit Timer that has five 16-bit capture/compare modules associated with it.
- Each of the modules can be programmed to operate in one of four modes.

1. Rising and/or falling edge capture
2. Software timer
3. High Speed Output
4. Pulse Width Modulator

Programmable Counter Array



Module functions:

- 16-bit capture
- 16-bit timer
- 16-bit high speed output
- 8-bit PWM

ERTS LAB
CSE IIT BOMBAY

Programmable Counter Array

- To Program PCA, we have to set some register before use it.
- Three registers are:
    1. CMOD - PCA Counter Register Mode
    2. CCON - PCA Counter Control Register
    3. CCAPMn - PCA Modules Compare/Capture Register
- All these three Registers are 8 Bit.

CMOD – PCA Counter Register Mode

Used to select clock frequency source for PCA counter to generate PWM and disable counter overflow interrupt.

Bit	Symbol	Description	Bit Value
7	CIDL	Counter Idle Control : CIDL=1, PCA counter gated off during idle.	1
6	WDTE	Watchdog Timer Enable: WDTE = 0; disable Watchdog timer	0
5	-	Reserved for future use. Should be set to 0	0
4			0
3			0
2	CPS1	PCA Count Pulse Select: CSP1 = 0; CSP0 = 0; Clock source: f_{osc} / 6 (f_{osc} = 11.0592MHz)	0
1	CPS0		0
0	ECF	PCA Enable Counter Overflow Interrupt. ECF = 0 disables that function.	0

CMOD=0X80;

CCON – PCA Counter Control Register

Used to start PCA Counter.

Bit	Symbol	Description	Bit Value
7	CF	PCA Counter Overflow Flag	0
6	CR	PCA Counter Run Control Bit : CR=1, Turn On PCA Counter	1
5	-	Reserved for future use. Should be set to 0	0
4	CCF4	PCA Module 4 Interrupt Flag, (Not Use).	0
3	CCF3	PCA Module 3 Interrupt Flag, (Not Use).	0
2	CCF2	PCA Module 2 Interrupt Flag, (Not Use).	0
1	CCF1	PCA Module 1 Interrupt Flag, (Not Use).	0
0	CCF0	PCA Module 0 Interrupt Flag, (Not Use).	0

```
CCON=0X40;
```

CCAPMn – PCA module Compare/Capture Register

Used to enable PWM generator.

In our case n = 0 for left motor and 1 for right motor.

Bit	Symbol	Description	Bit Value
7	-	Reserved for future use. Should be set to 0	0
6	ECOMn	Enable Comparator : ECOMn = 1 enables the comparator function.	1
5	CAPPn	Capture Positive : CAPPn = 0; Disable positive edge capture.	0
4	CAPNn	Capture Negative : CAPNn = 0; Disable negative edge capture.	0
3	MATn	Match : MATn = 0; Disable flagging of the interrupt in CCFn bit in CCON register.	0
2	TOGn	Toggel, TOGn = 0; Disable toggeling of the CEXn when match of the PCA counter with this module's compare/capture register occours.	0
1	PWMn	Pulse Width Modulation Mode. PWMn = 1; Enables the CEXn pin to be used as a pulse width modulated output.	1
0	ECCFn	Enable CCF Interrupt, ECCFn = 0; Disables compare/capture flag CCFn in the CCON register to disable generation of an interrupt.	0

CCAPM0=0X42; //Left Motor

CCAPM1=0X42; //Right Motor

CCAPnH and CCAPnL

- CCAPnH and CCAPnL registers are use to set duty cycle which are associated with each of PCA module.

- To set the velocity of the left motor:

  CCAP0L=0x00;

  CCAP0H=0xFF;

  Left motor duty cycle. 0 to 0xFF gives 0 to 100% duty cycle i.e. velocity from 0 to maximum speed. Any value in between 0x00 and 0xFF will give intermediate velocity.

- To set the velocity of the right motor:

  CCAP1L=0x00;

  CCAP1H=0xFF;

  Right motor duty cycle. 0 to 0xFF gives 0 to 100% duty cycle i.e. velocity from 0 to maximum speed. Any value in between 0x00 and 0xFF will give intermediate velocity.

To Initialize PCA

```
void pca_init()
    {
        CMOD=0x80;      // Set clock frequency
        CCON=0x40;      // Start PCA Counter

        CCAPM0=0x42; //To enable PWM generator
        CCAPM1=0x42; //To enable PWM generator

        CCAP0L=0x00; //Left Motor Duty Cycle Registers
        CCAP0H=0xFF;

        CCAP1L=0x00; //Right Motor Duty Cycle Registers
        CCAP1H=0xFF;
    }
```