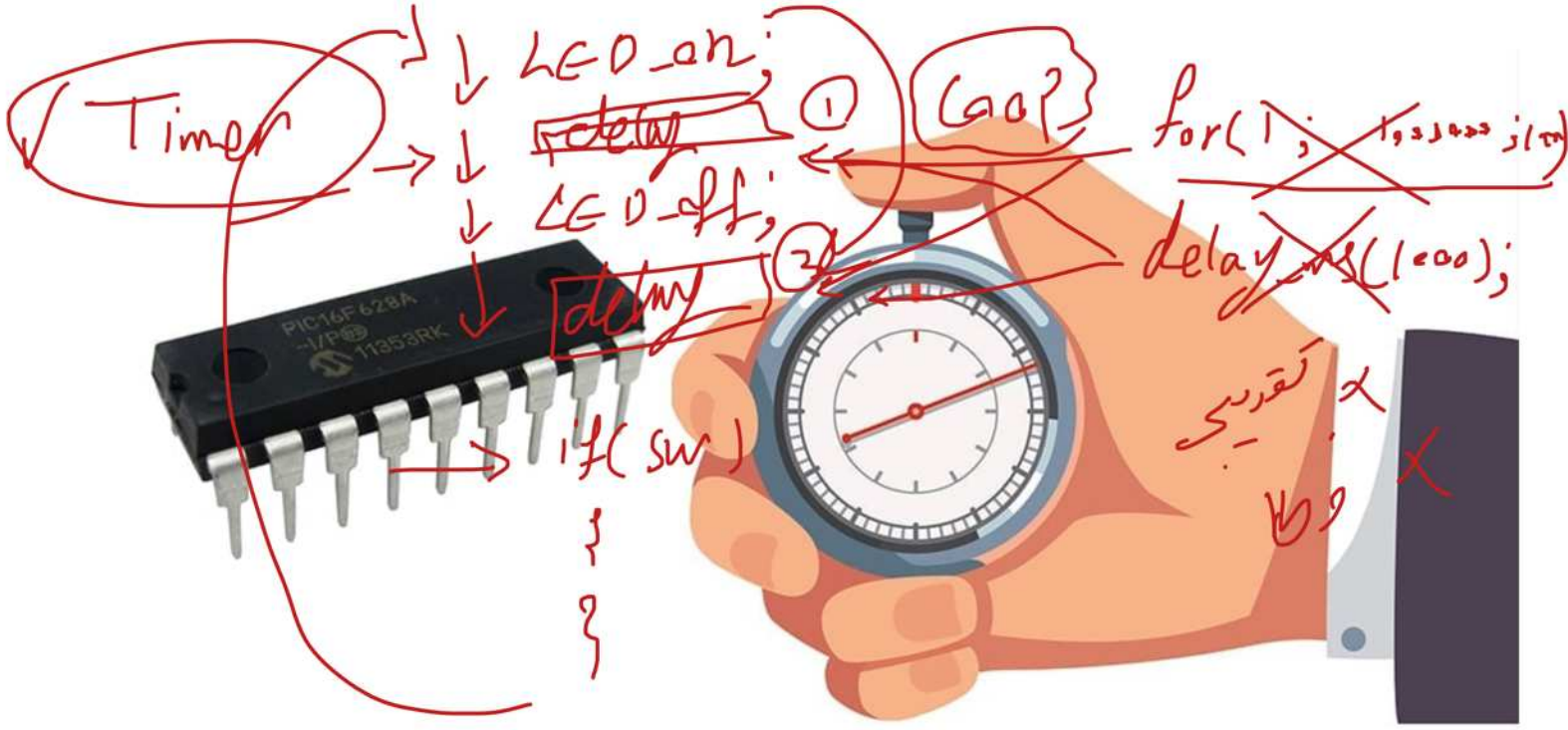
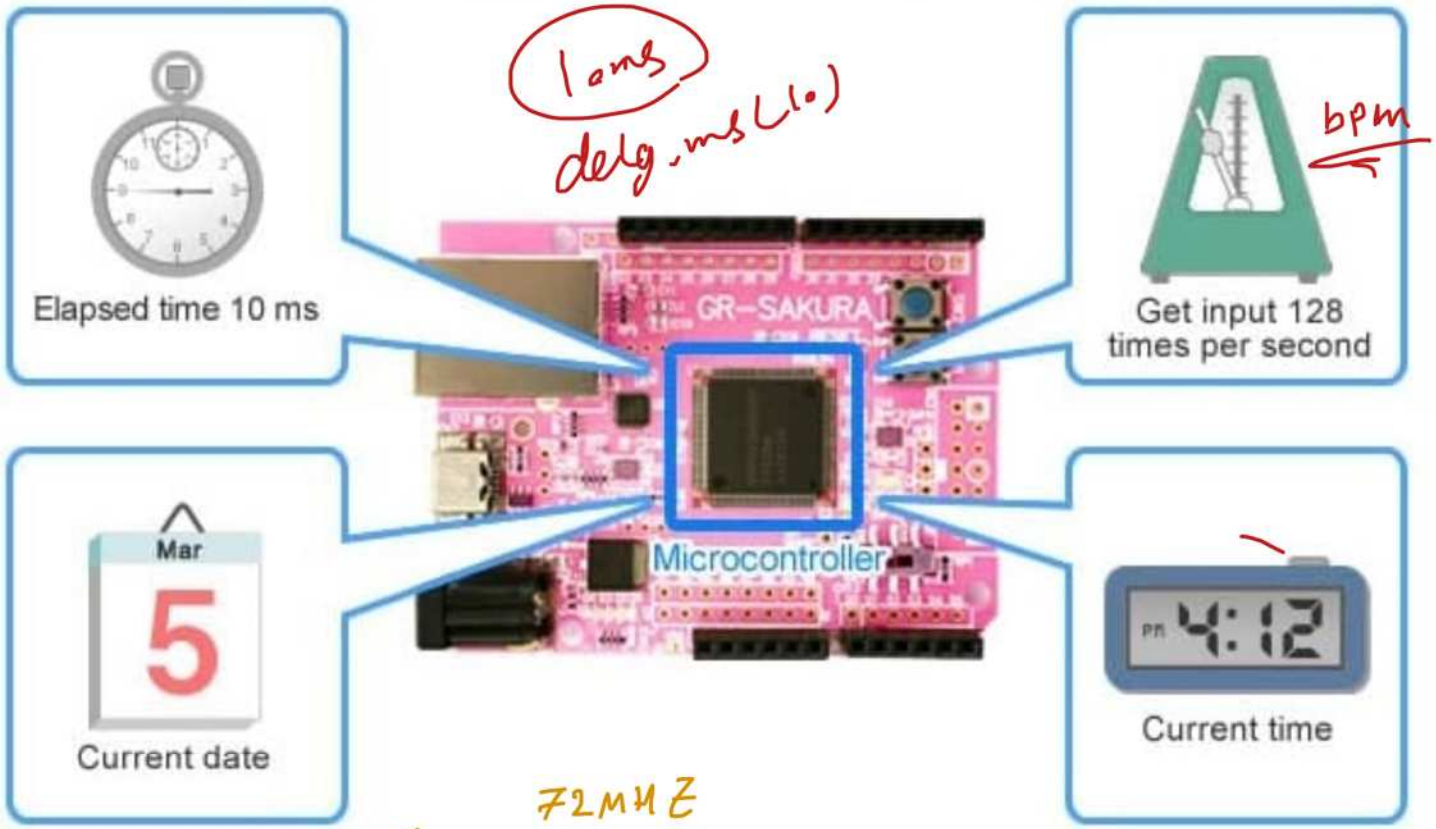
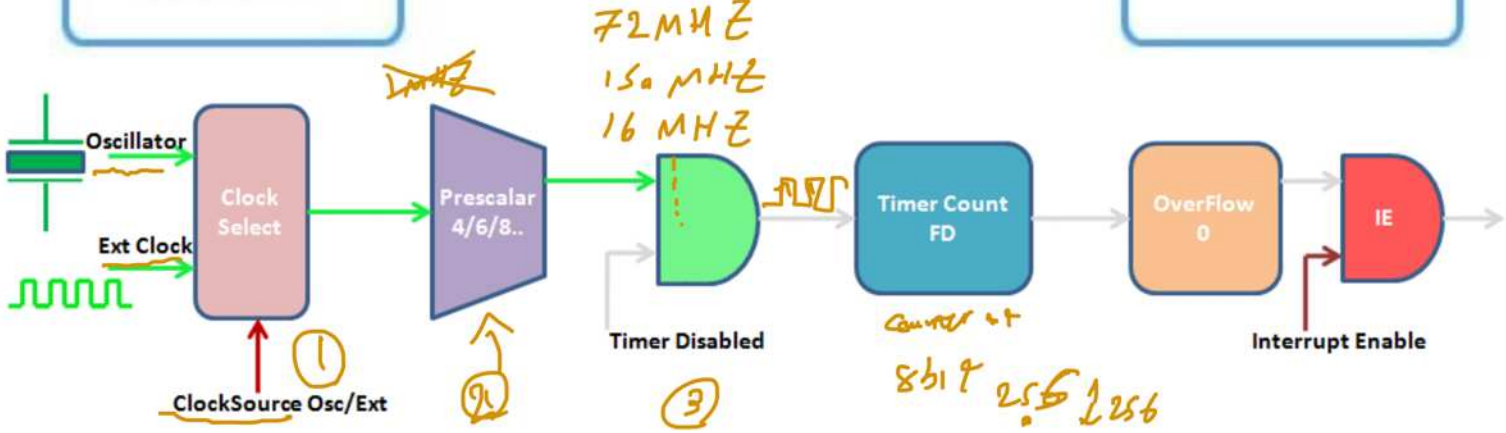
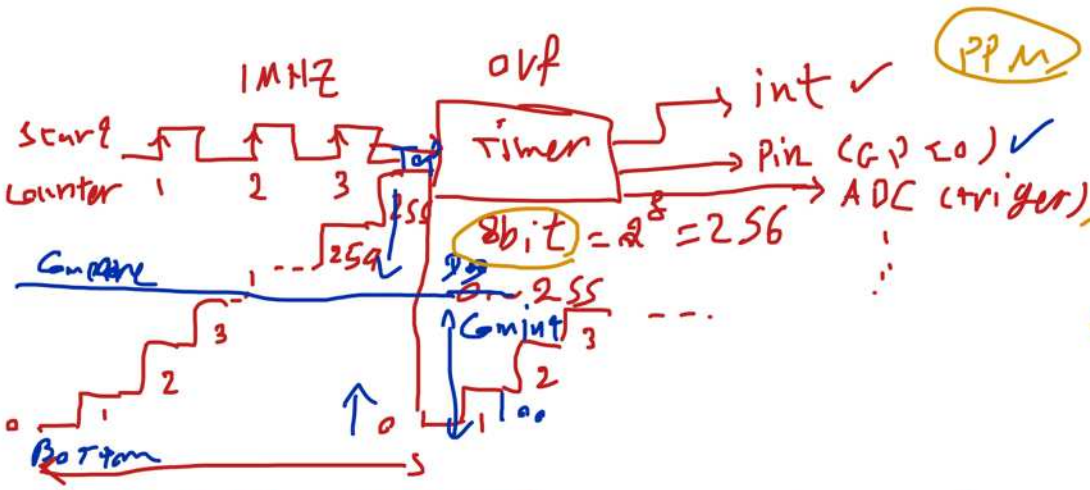

Timer
LED_on;
delay
LED_off;
delay
if(sw)
Goo?
for(1;
delay_ms(1000);
PIC16F628A
11353RK


10ms
delay_ms(10)
bpm
Elapsed time 10 ms
Get input 128 times per second
GR-SAKURA
Microcontroller
Mar
5
Current date
4:12
Current time


72MHZ
150 MHZ
16 MHZ
Oscillator
Ext Clock
Clock Select
Prescalar 4/6/8..
Timer Disabled
Timer Count FD
OverFlow 0
IE
Interrupt Enable
ClockSource Osc/Ext
1
2
3
8bit 256 256



8bit = 256

16bit = 65536

32bit = 4,294...

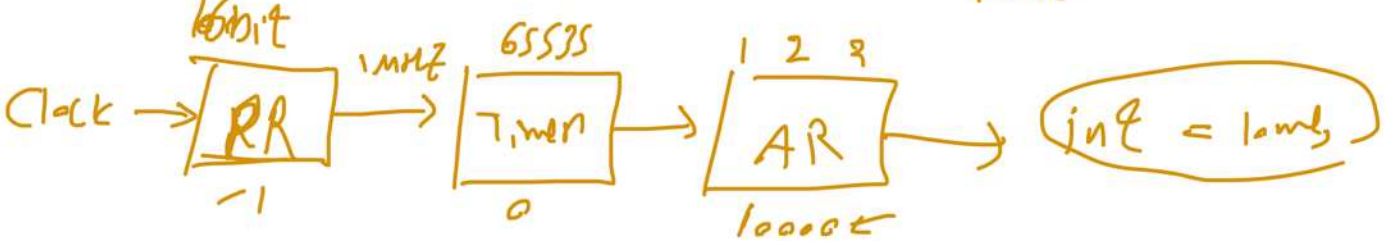
$T_{int} = \frac{256}{1\times10^6} = \boxed{256\,\mu s} \longrightarrow$ int_ovf

ISR(ovf)
{
counter++
مجموع کدها
$\boxed{256\,\mu s} \times 4 =$
if(counter==4)
{
≃ 1ms = 1.024ms
}
}

ISR(Com Int) $T_{int} = \frac{100}{1\times10^6} = \boxed{100\,\mu s}$

{
counter++ ← 100μs
if(counter==10)
{ ← 1ms
}
}

$T_{int} = 1ms \Rightarrow 1\times10^{-3} = \frac{x}{1\times10^6} \Rightarrow \boxed{x = 1000}$

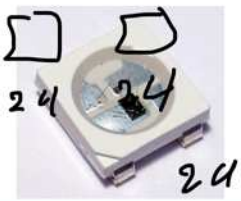


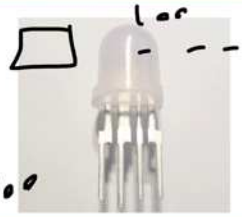
-1
-2
-4
⋮
-1024

1 8 64
128 256 1024
0~65535

AR ?

$\frac{72MHZ}{72} = \frac{1MHZ}{10000} = 100HZ$





	Timing WS2812(S)	Timing WS2812B	WS2812D (8mm)	PD9823 (5/8mm)	Cycles
T_{HI_IN} "0"	62.5 ns - 500 ns	62.5 ns - 563 ns	62.5 ns - 600 ns	62.5 ns - 600 ns	<3
T_{HI_IN} "1"	≥ 563 ns	≥ 625 ns	≥ 625 ns	≥ 625 ns	>3
T_{PERIOD_IN}	≥ 875 ns	≥ 1063 ns	≥ 1063 ns	≥ 1000 ns	>5
$T_{DELAY\_IN\_OUT}$	~ 166 ns	~ 208 ns	~ 208 ns	~ 190 ns	1
T_{HI_OUT} "0"	~ 333 ns	~ 416 ns	~ 408 ns	~ 380 ns	2
T_{HI_OUT} "1"	~ 666 ns	~ 832 ns	~ 812 ns	~ 760 ns	4
T_{RESET}	> 10.8 µs	> 9 µs	tbd	> 14 µs	-

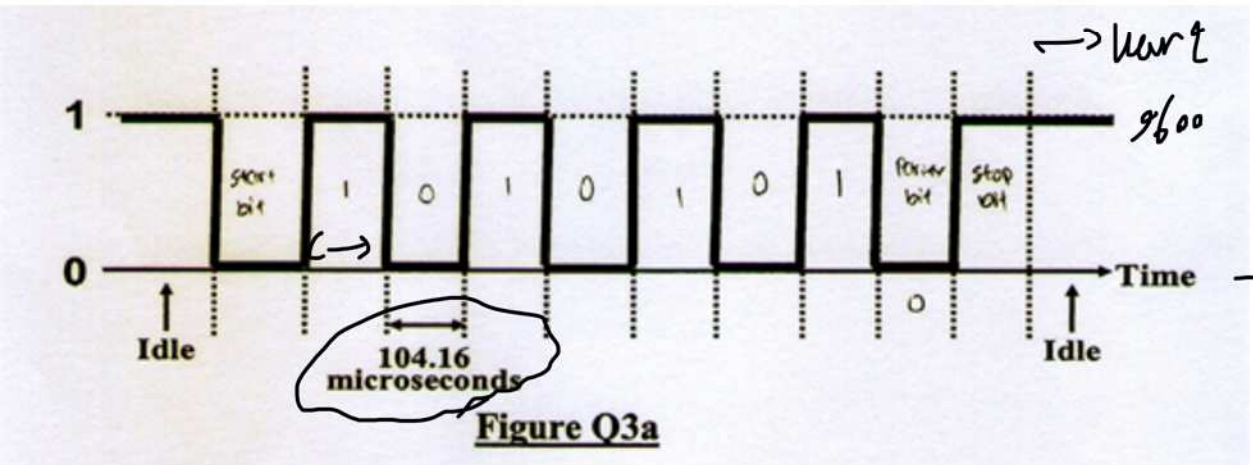
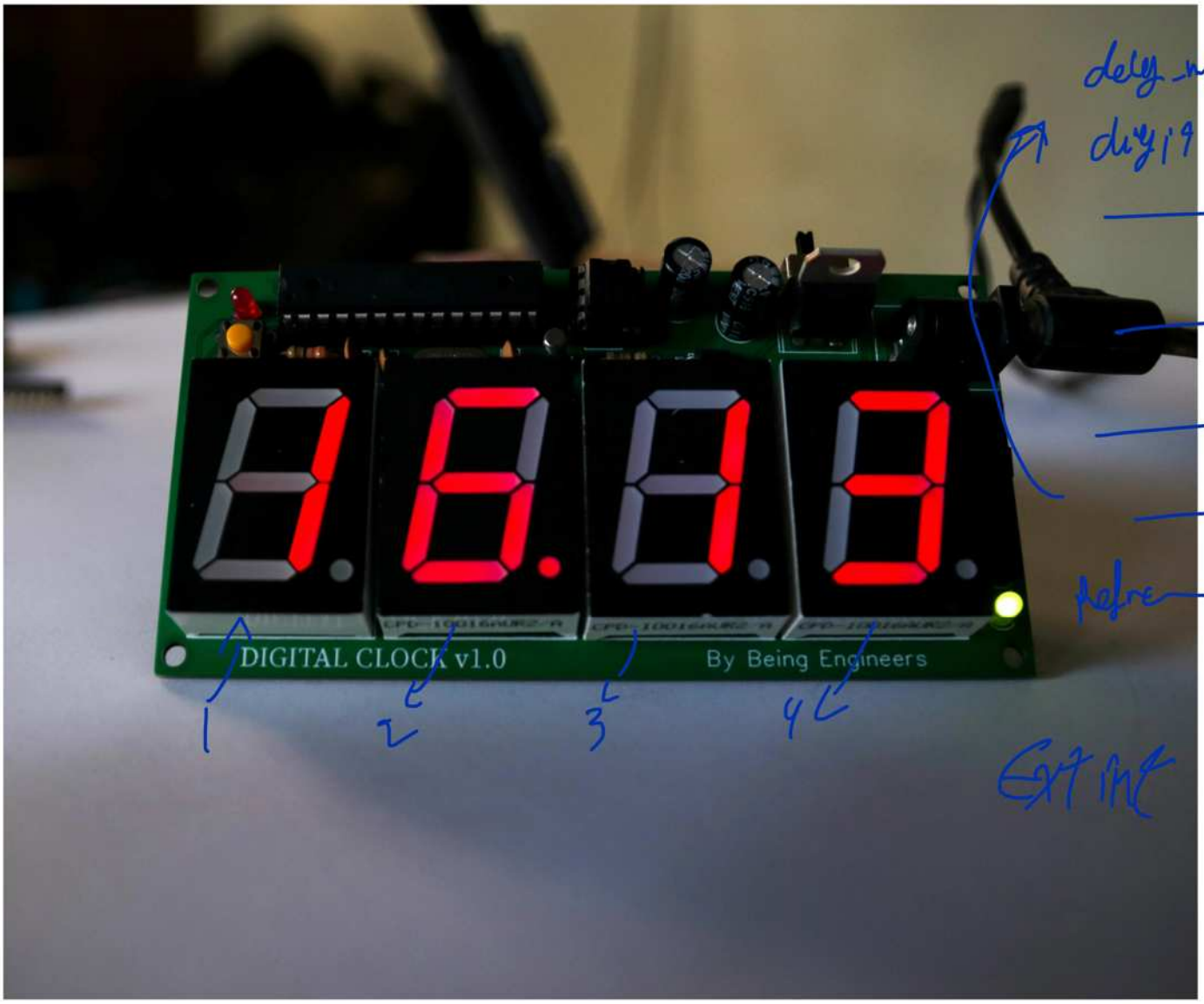
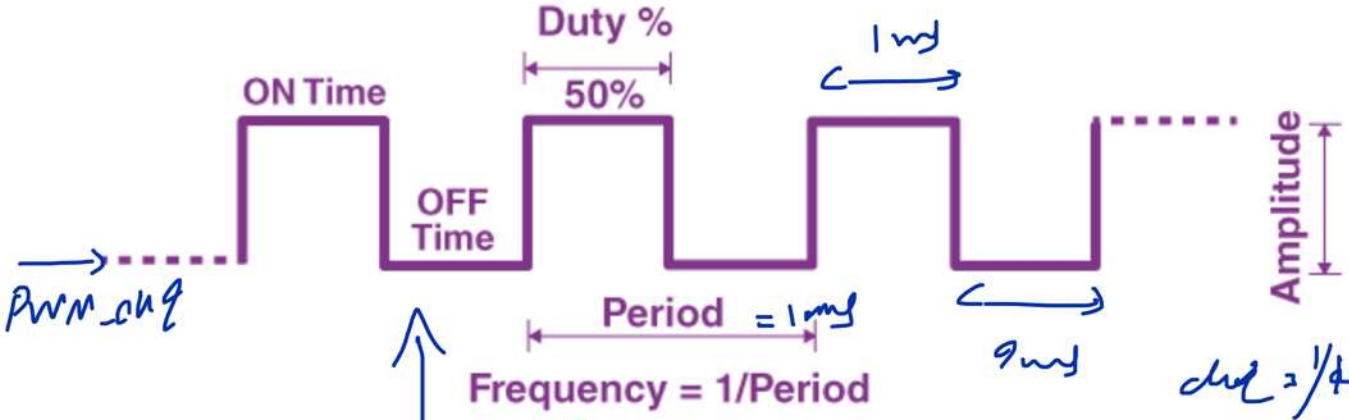


Figure Q3a

START
PAUSE
001
INDESIT


DIGITAL CLOCK v1.0
By Being Engineers

PWM ساختن ②



out pnt

duty = 1/10

Period = 10ms

PA0 → High
delay_ms(1);
PA0 → Low
delay_ms(0)

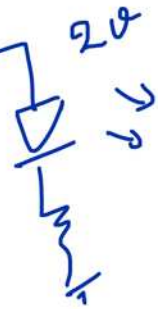
Loop

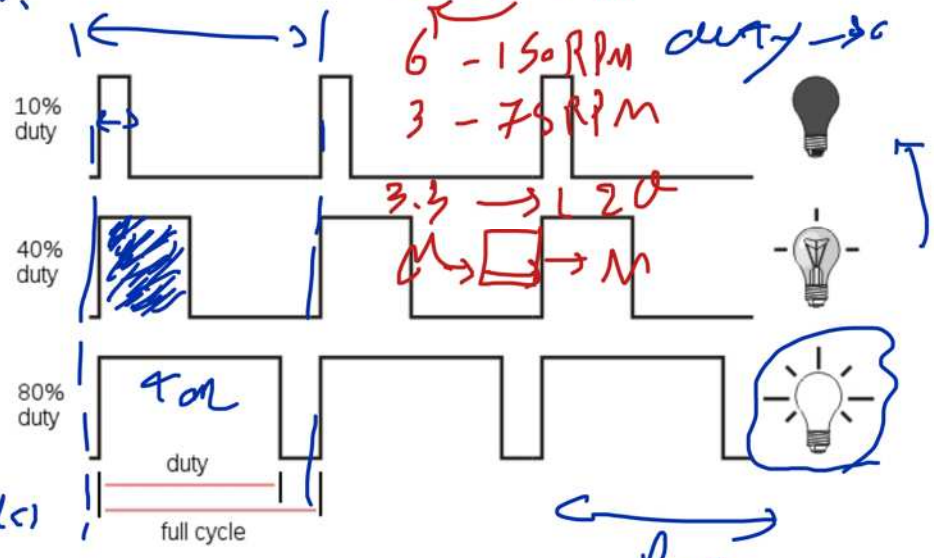
fref

do
Lo

High → 3.3 → duty = 1/10
Low → 0 → duty = 1/0

PA0 2V

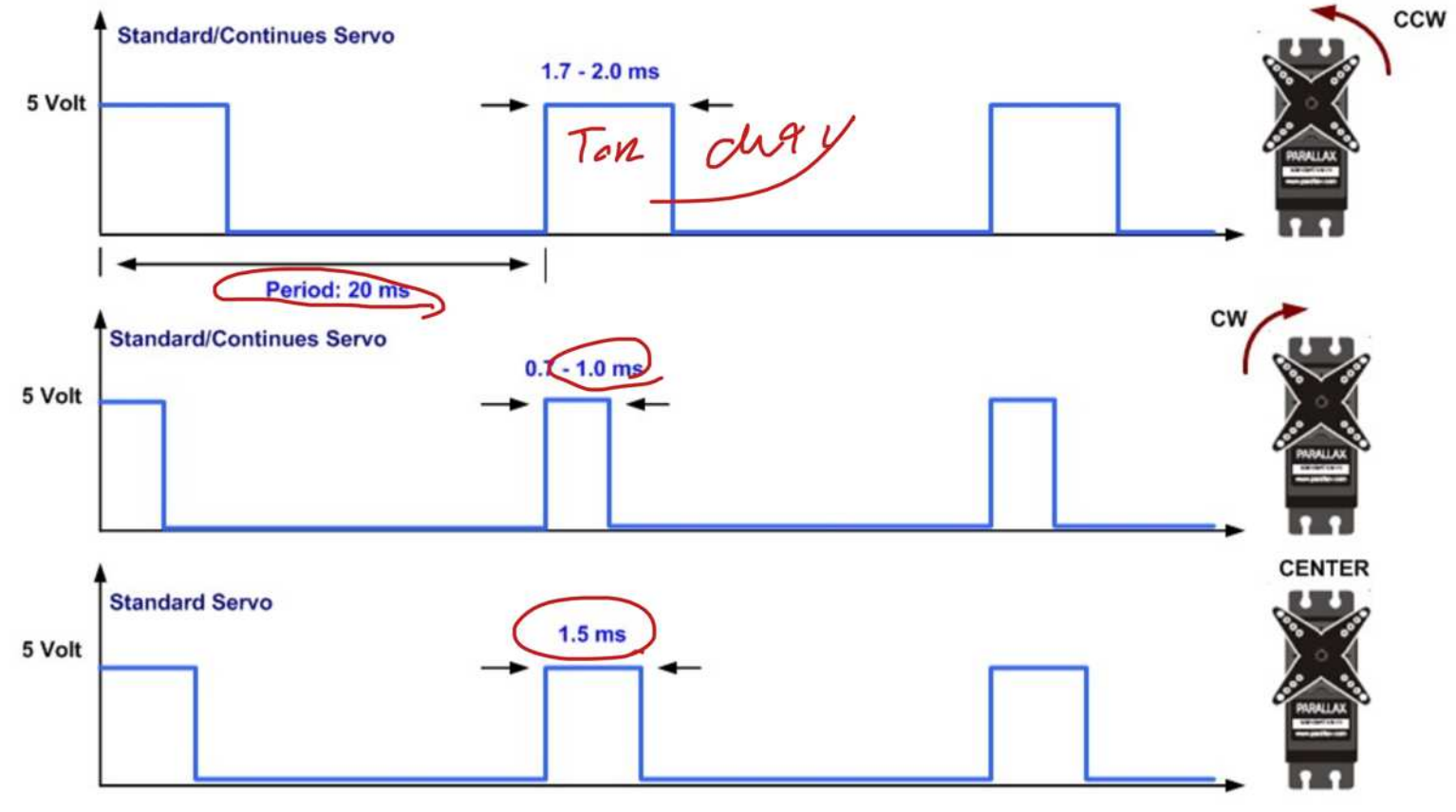




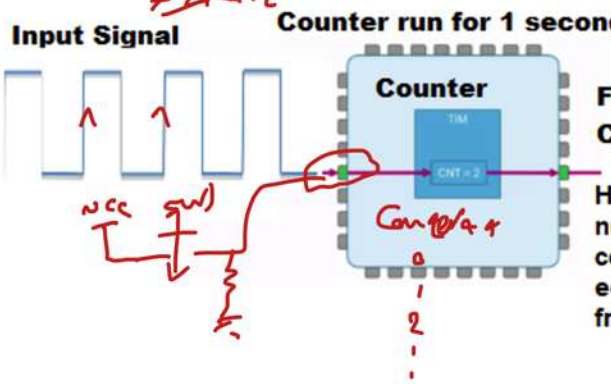
12 - 300 RPM
6 - 150 RPM
3 - 75 RPM
3.3 → 12V
M → M
LED off
duty → 0
4 on
fref

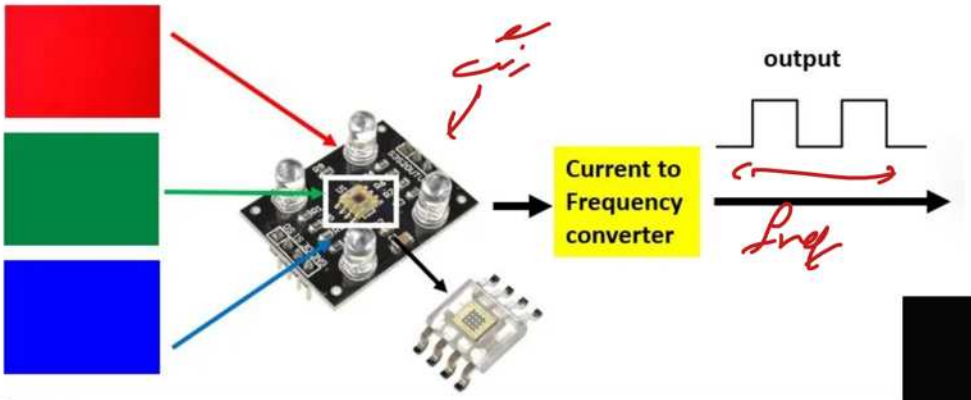
$V_{out} = d\, V_{in}$

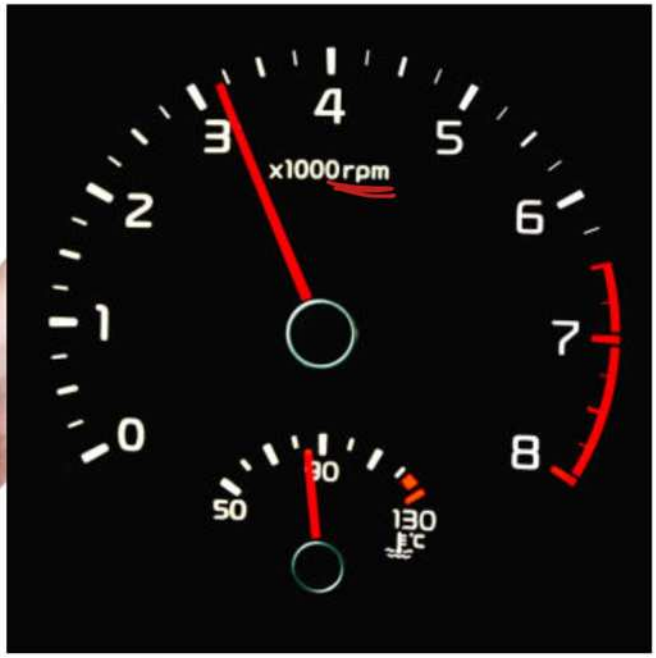
$V_{out} = 3.3\, d \rightarrow 0 < d < 1$



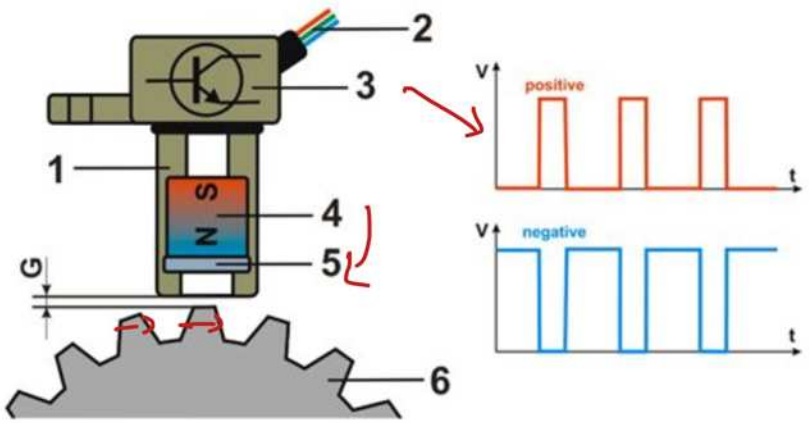
Servo Motor PWM Timing Diagram

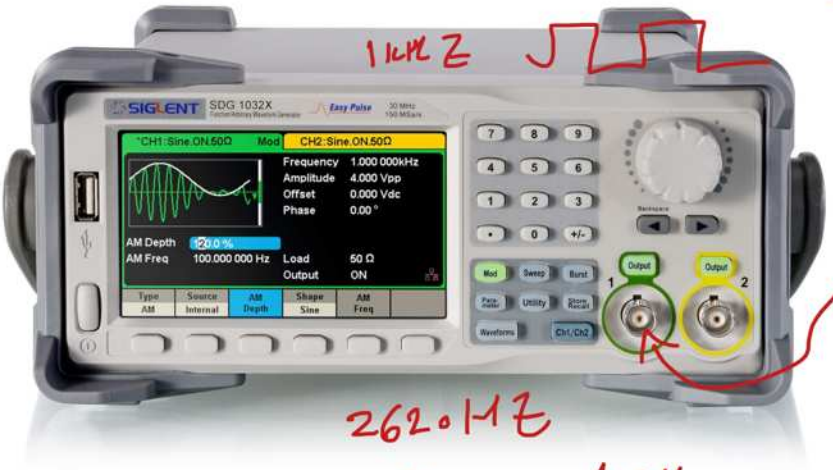

Input Signal
Counter run for 1 second
Counter
Frequency =
Cycles/Second
Hence in 1 second
number of
counts(positive
edge) is equal to
frequency

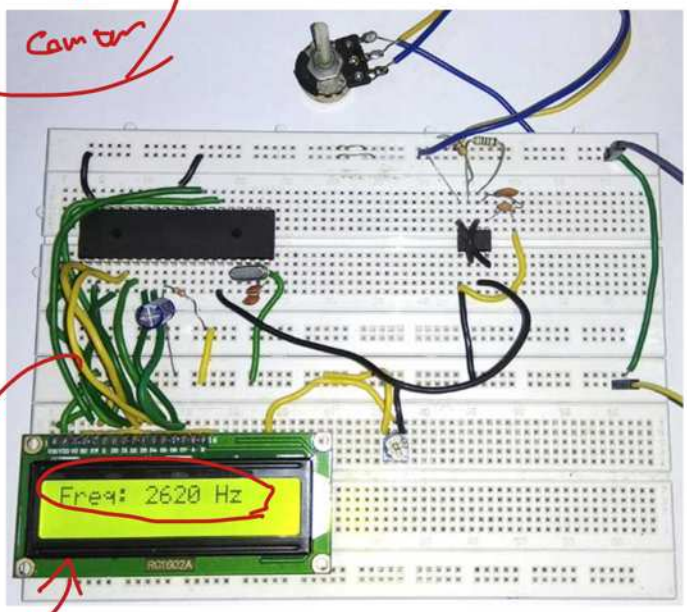

output
Current to
Frequency
converter

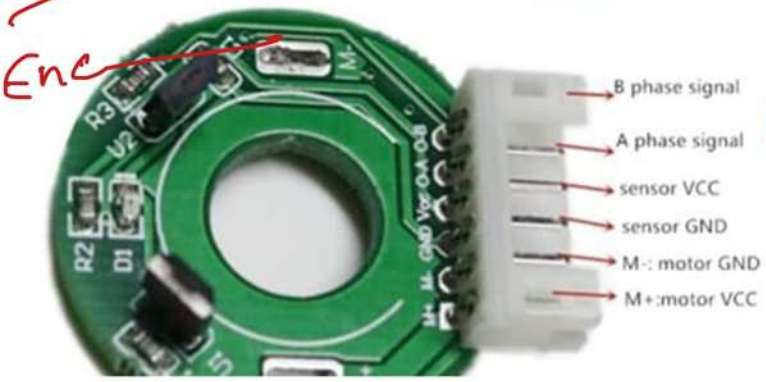

x1000rpm

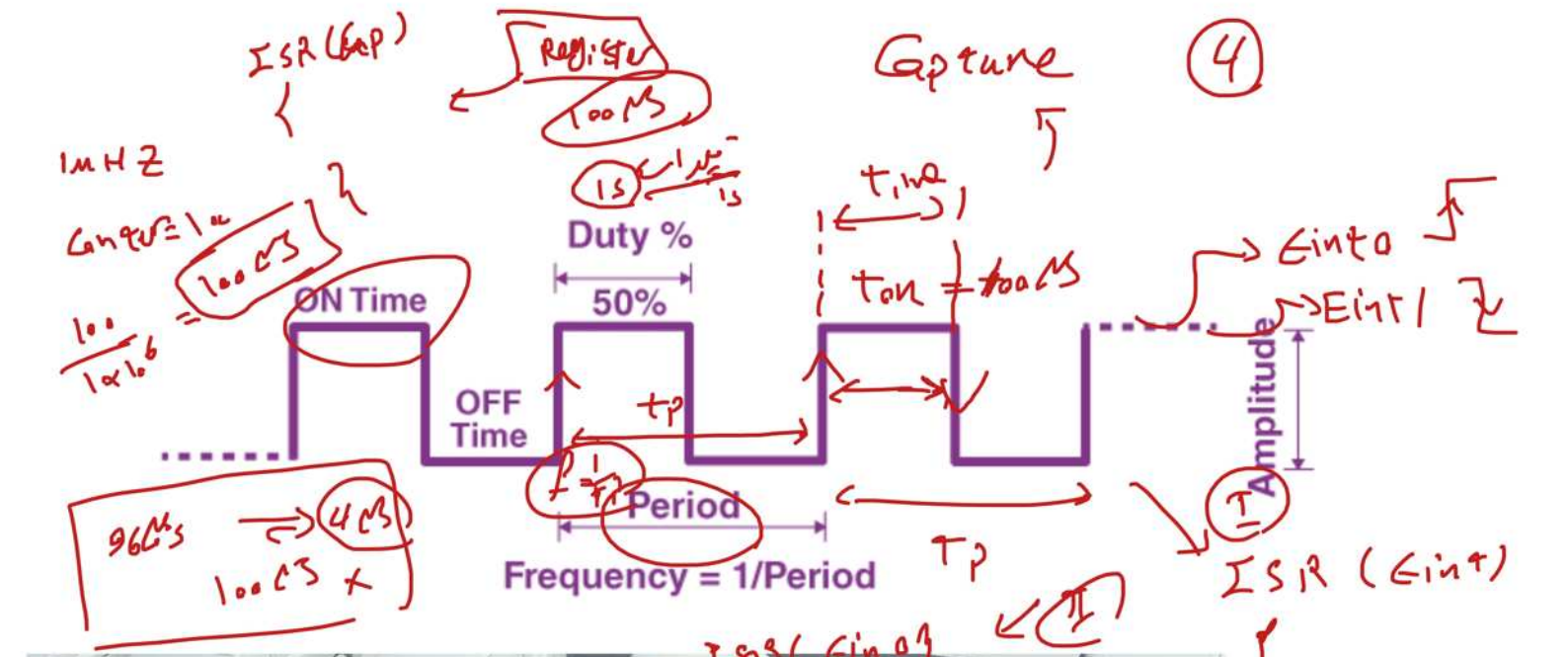
RPM
RPS
625.8
CHECKLINE

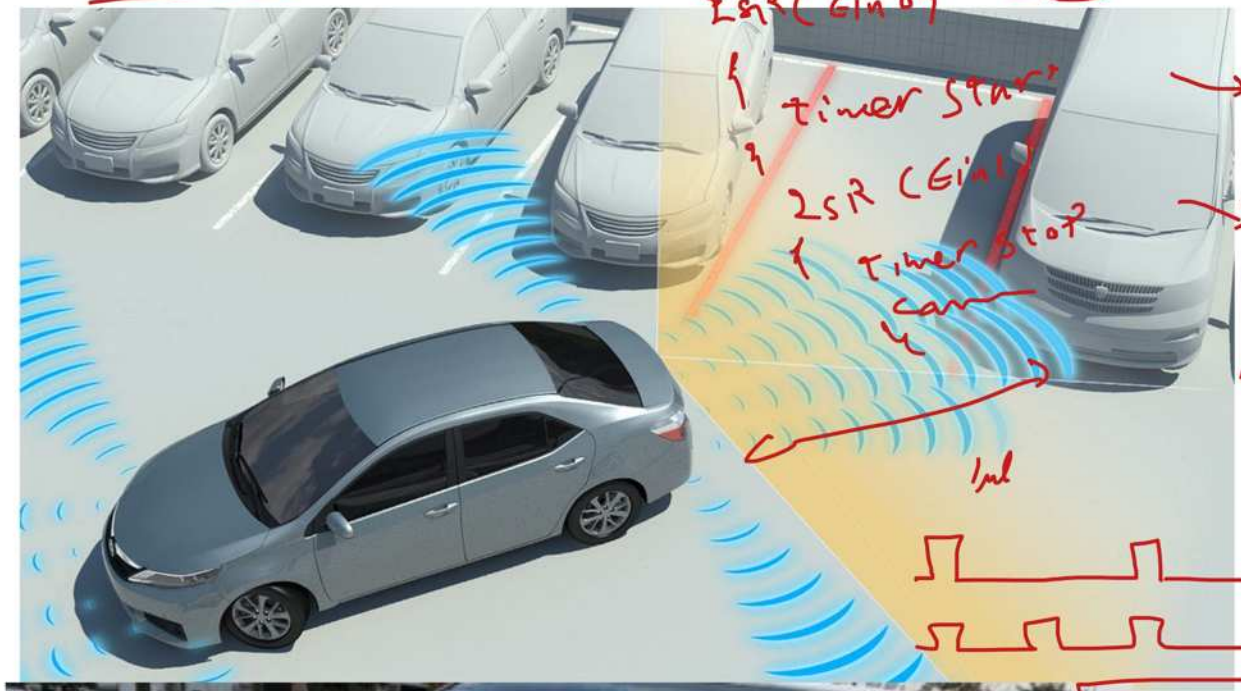

positive
negative


SIGLENT
SDG 1032X
CH1:Sine.ON.50Ω
CH2:Sine.ON.50Ω
Frequency 1.000 000kHz
Amplitude 4.000 Vpp
Offset 0.000 Vdc
Phase 0.00 °
AM Depth 120.0 %
AM Freq 100.000 000 Hz
Load 50 Ω
Output ON

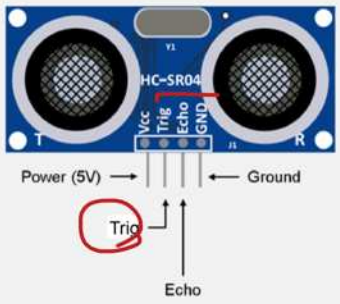

Freq: 2620 Hz

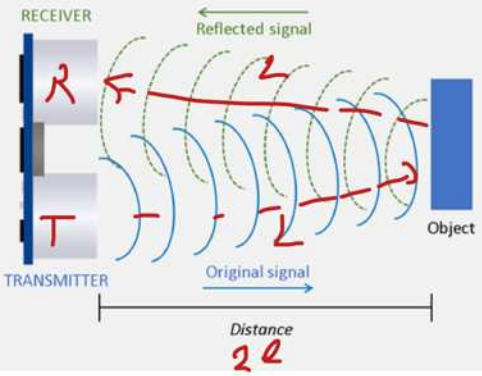

B phase signal
A phase signal
sensor VCC
sensor GND
M-: motor GND
M+:motor VCC


ISR (Cap)
Register
100 µs
Capture
4
1MHZ
Counter = 100
100 µs
1s
time
Duty %
50%
ON Time
OFF Time
Period
Frequency = 1/Period
Amplitude
Eint0
Eint1
96 µs
4 µs
100 µs
Tp
ISR (Eint)
ISR (Eint0)
timer start
ISR (Eint1)
timer stop
Counter
1µs

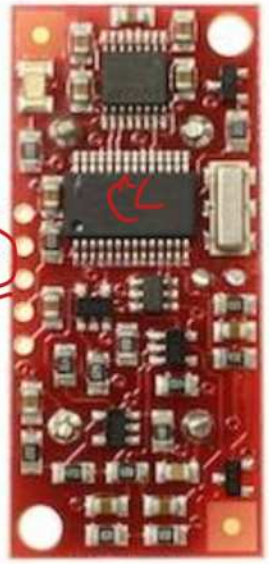


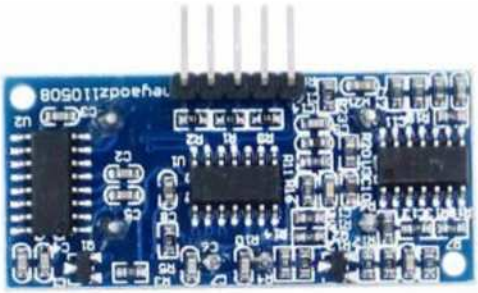
PEUGEOT
PARS ELX

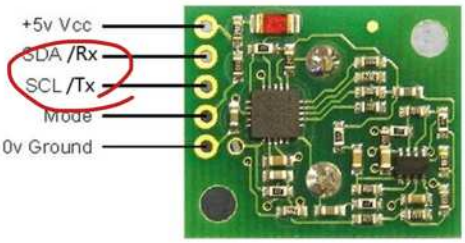

HC-SR04
Vcc
Trig
Echo
GND
Power (5V)
Ground
Trig
Echo

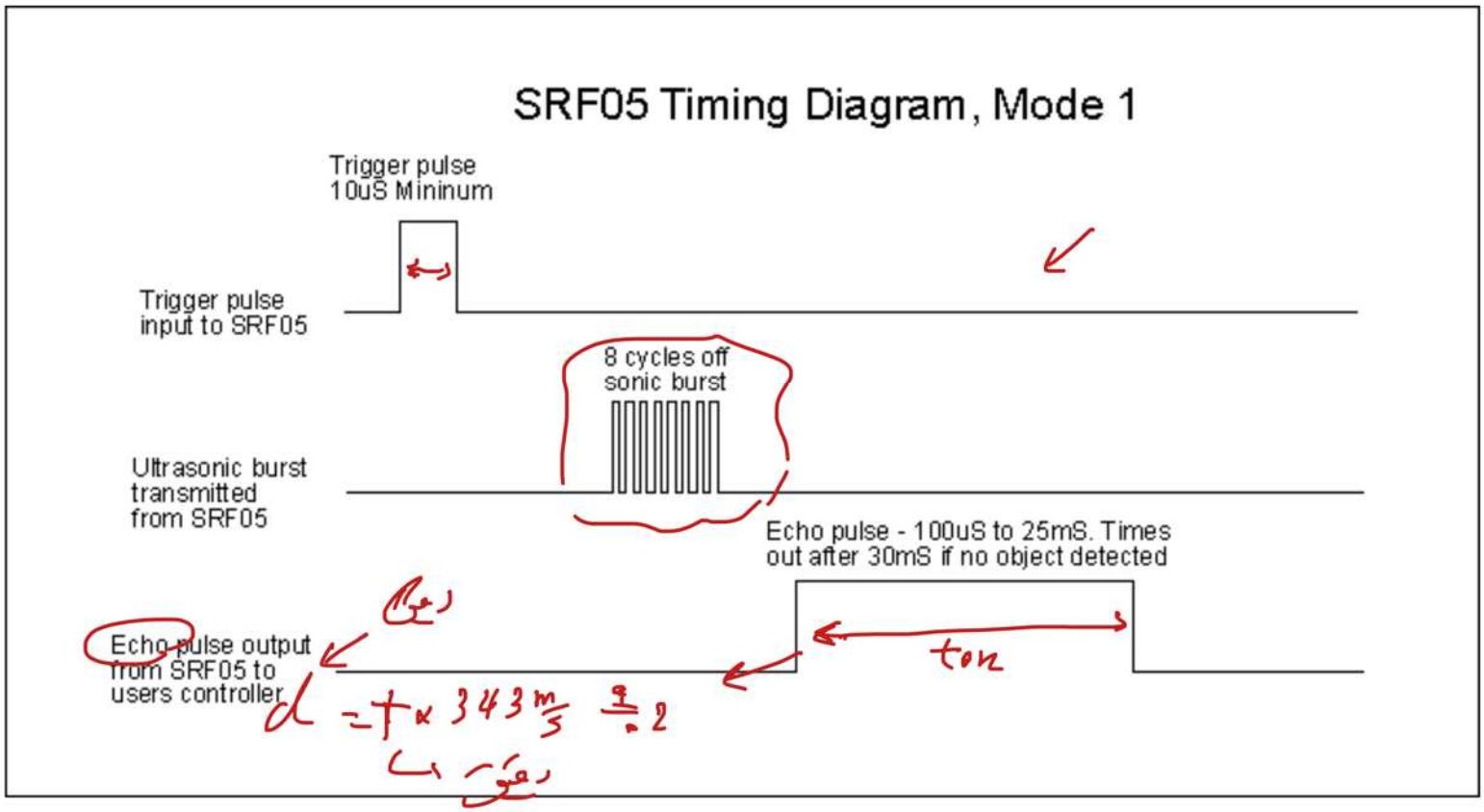

RECEIVER
Reflected signal
R
T
Object
TRANSMITTER
Original signal
Distance

SRF05
Vcc
Trig
Echo
OUT
GND
T
R

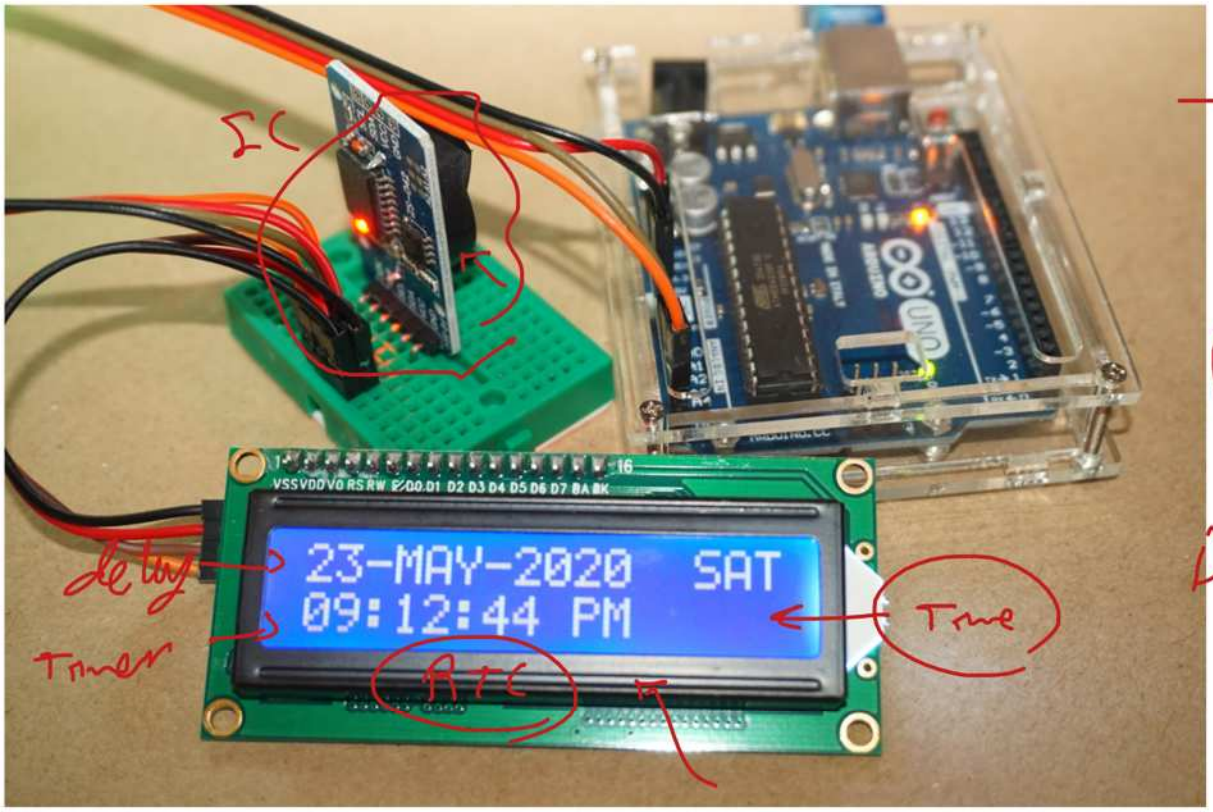

5V
SDA
SCL
Do Not Connect
GND

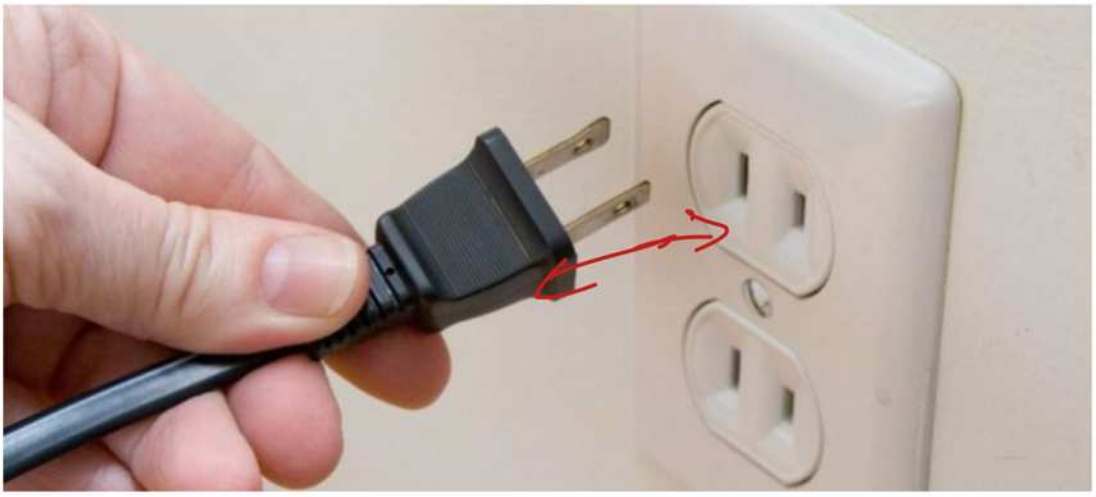



+5v Vcc
SDA /Rx
SCL /Tx
Mode
0v Ground


SRF05 Timing Diagram, Mode 1
Trigger pulse
10uS Mininum
Trigger pulse
input to SRF05
8 cycles off
sonic burst
Ultrasonic burst
transmitted
from SRF05
Echo pulse - 100uS to 25mS. Times
out after 30mS if no object detected
Echo pulse output
from SRF05 to
users controller
ton

WESTCLOX
PM
12:08
AM
ALARM
6:30
MONTH
DATE
7/ 4
TEMP
72.1


IC
23-MAY-2020 SAT
09:12:44 PM
Delay
Timer
RTC
Time



:(

Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

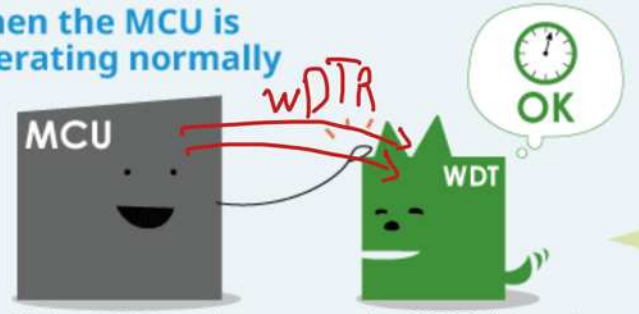
20% complete

For more information about this issue and possible fixes, visit https://www.windows.com/stopcode

If you call a support person, give them this info:
Stop code: CRITICAL_PROCESS_DIED

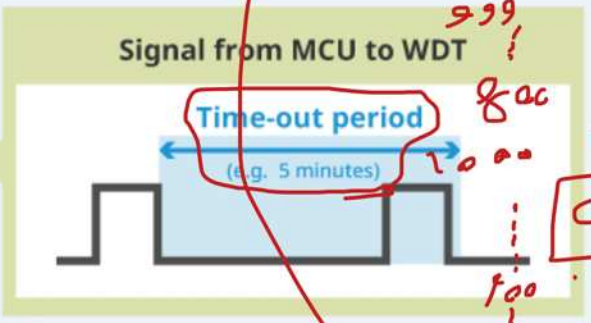
● WDT operation (Time-out mode)

When the MCU is operating normally



If the MCU strokes (= initializes) the WDT once every 5 minutes,

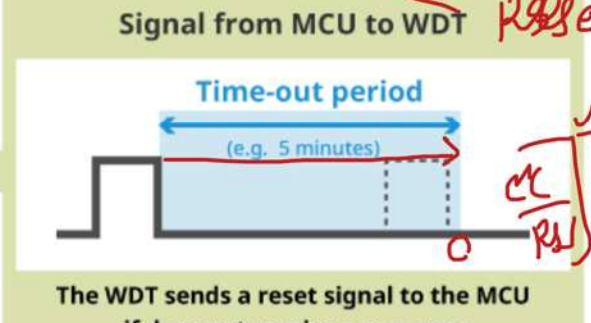
the WDT determines the MCU is operating normally.



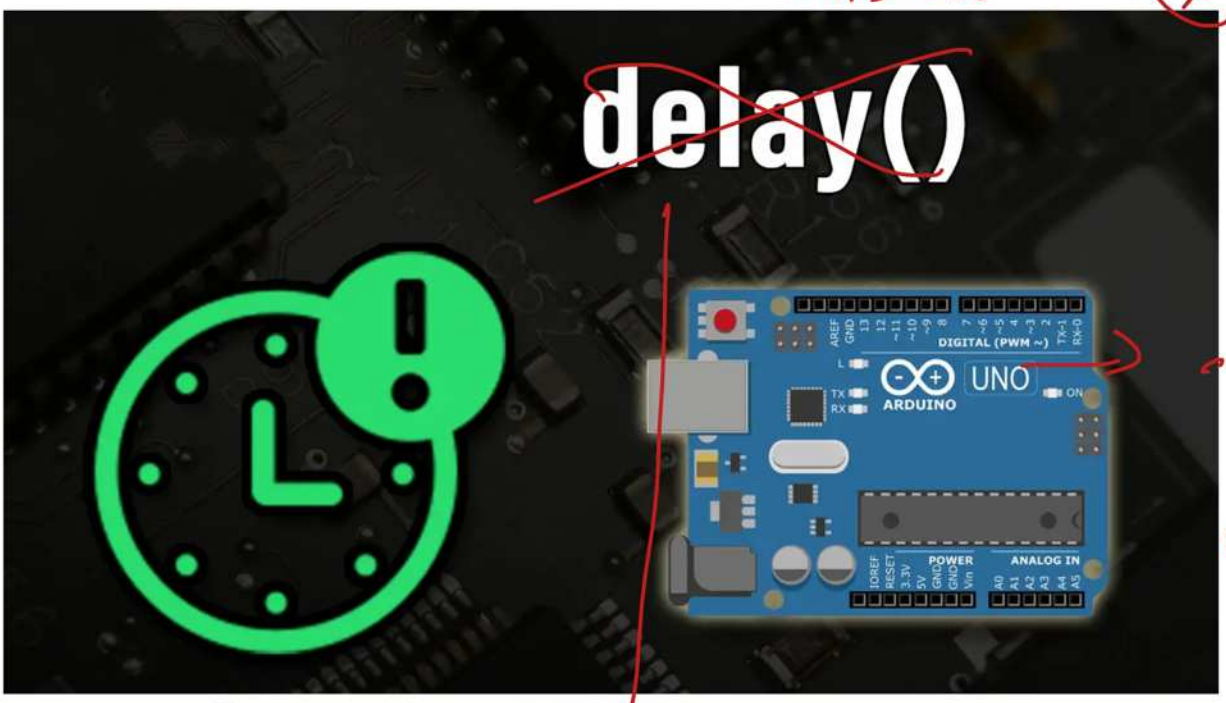
When the MCU is malfunctioning

If the MCU does not stroke (= initialize) the WDT in a 5 minute period,

the WDT detects the MCU fault and barks (= reboots).



The WDT sends a reset signal to the MCU if does not receive a response within the set timeout period.

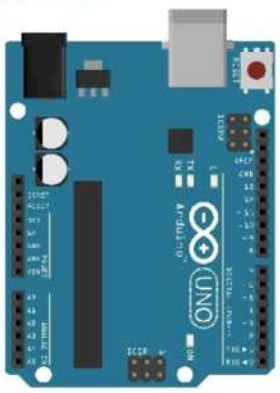


main.c | main.c | main.h | main.c | syscalls.c | syscalls.c

```
10 #include<stdint.h>
11 #include<stdio.h>
12
13 void delay(void)
14 {
15     for(uint32_t i =0 ; i < 300000 ; i++);
16
17 }
18
```

```
LD HL, $0220
Init_Delay:
NOP
NOP
DEC L
JP NZ, Init_Delay
DEC H
JP NZ, Init_Delay
```

MULTITASKING



millis()

Simple Way!!

TASK 1

TASK 2

TASK 3

TASK 4