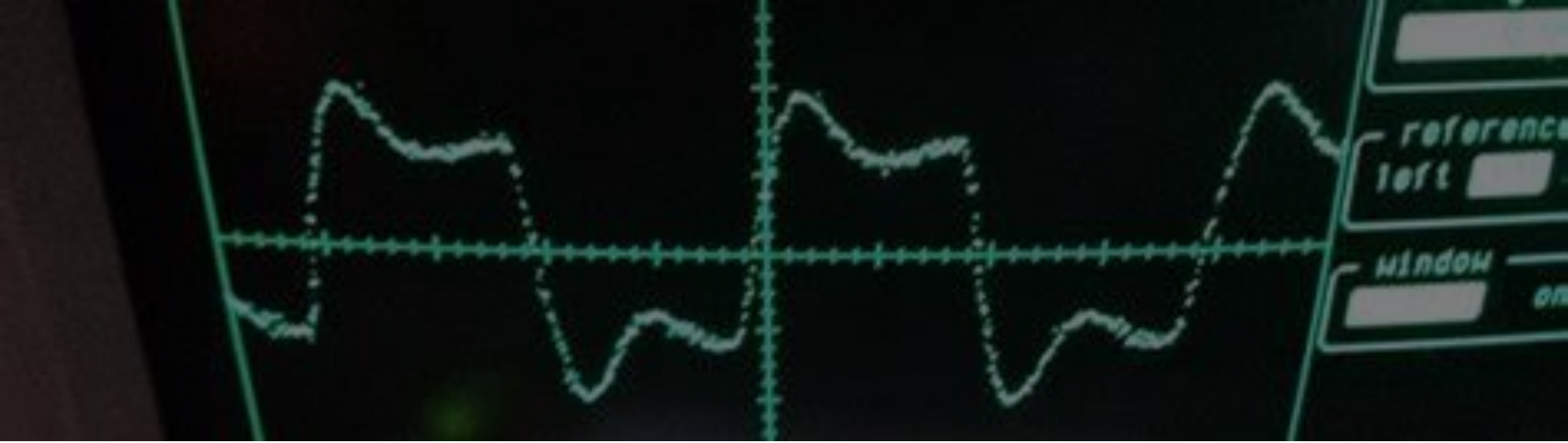


PWM DRAWER

AMIT Graduation Project

Nourhan Osman Donia

Contents

Introduction

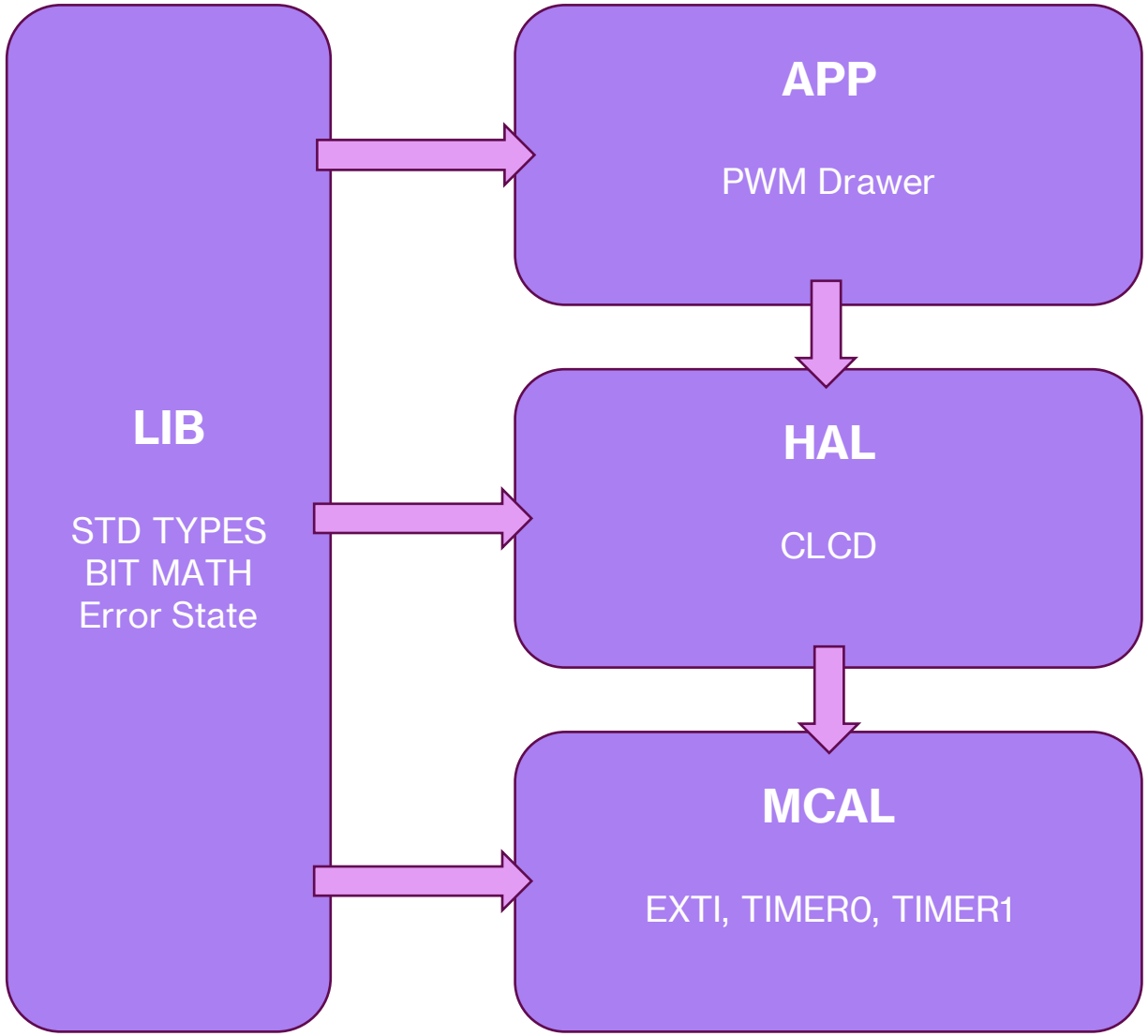
- Pulse Width Modulation (PWM) is a type of signal that is generated from a digital IC (such as a microcontroller)
- It is produced in the form of pulses/ square waves. Meaning, the signal will be a train of high levels and low levels. e.g., 5 v and 0 v.

We use generally PWM signal for two functions:

1. Communication.
2. Load Control (e.g. Motor Speed, Light Intensity, etc.)

- In this project, we are going to use PWM as a way of communication.
- We are going to generate a PWM signal from our microcontroller (Atmega32) then receive it on the same microcontroller, calculate the Duty Cycle and Frequency of the signal, then print it on the LCD.

Layered Architecture



Specs

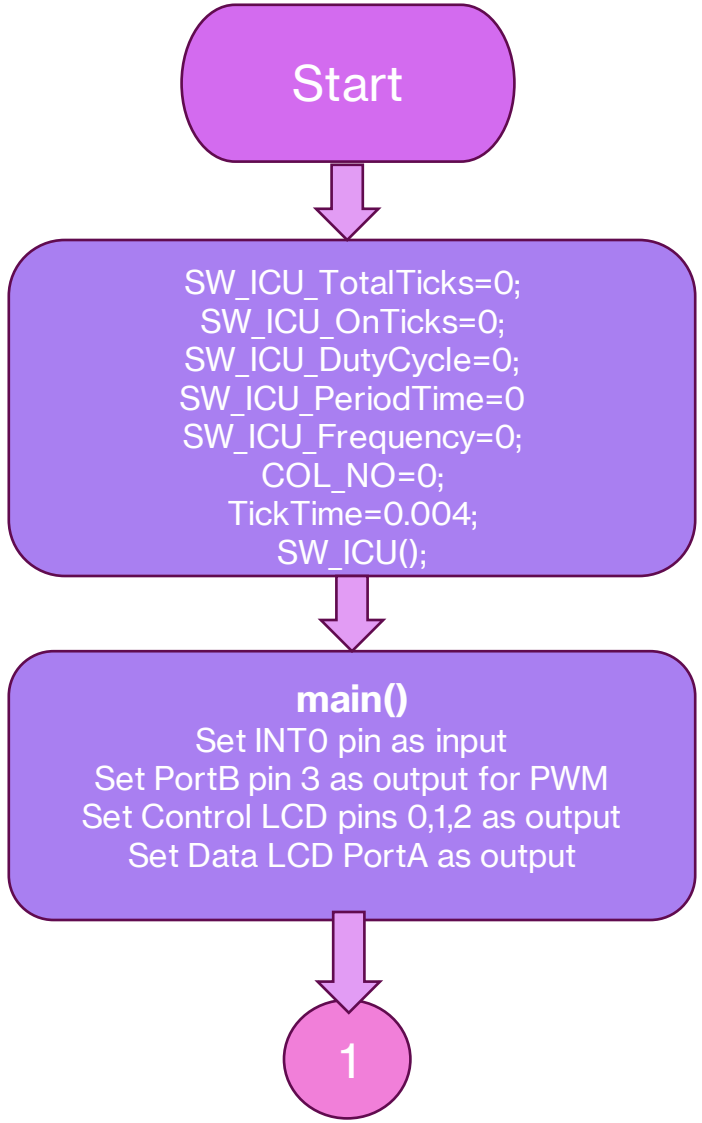
1. **Microcontroller:** ATMEGA32
2. **EXTI :** INT0 and sense rising edge.
3. **Timer0 :** Generates PWM signal.

- *Mode:* Fast PWM mode.
- *Prescaler:* division by 64.
- *OCR0 value:* 127.

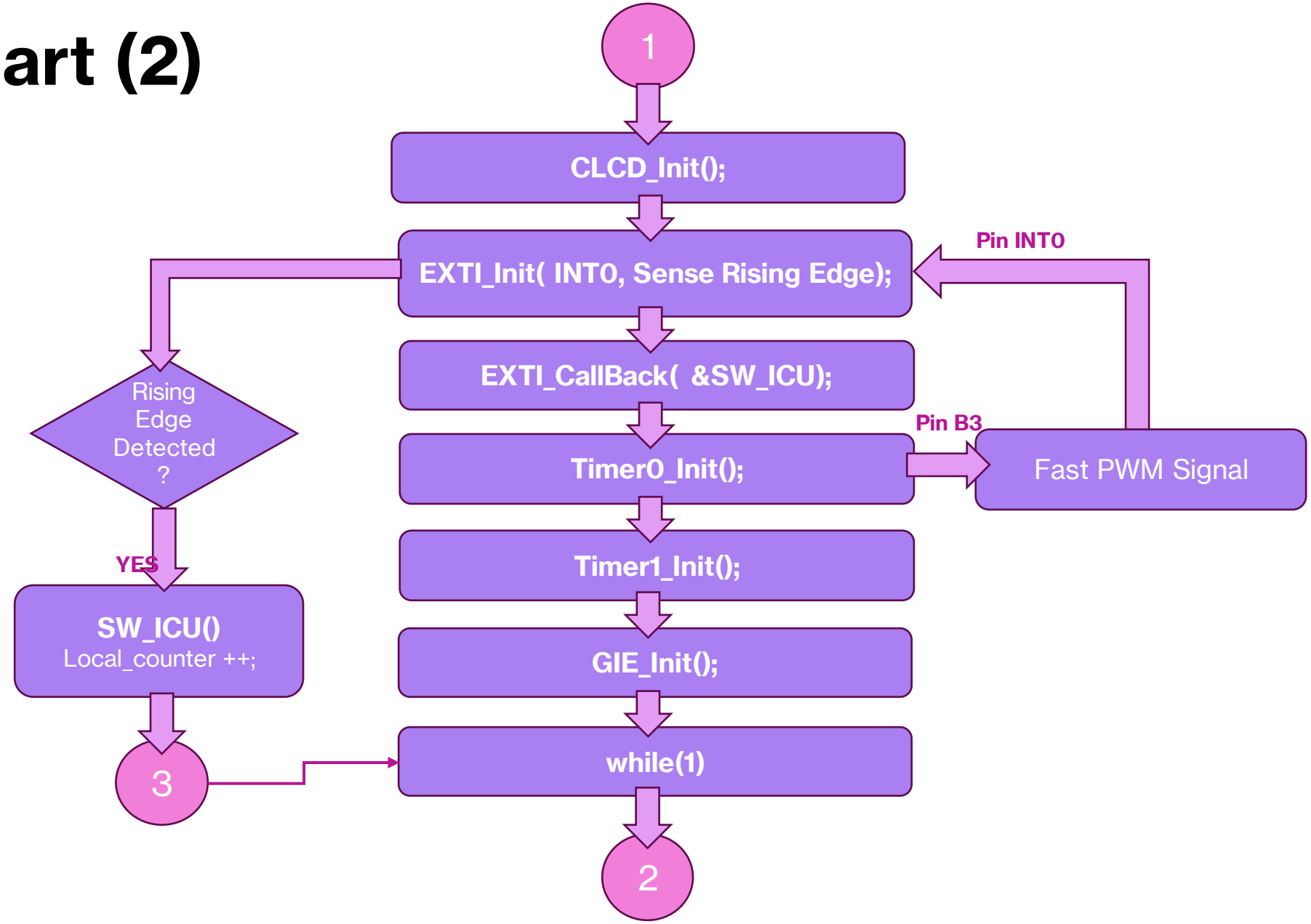
3. **Timer1 :** Calculates Duty Cycle, Period Time, & Frequency.

- *Mode:* Normal mode
- *Prescaler:* division by 64.

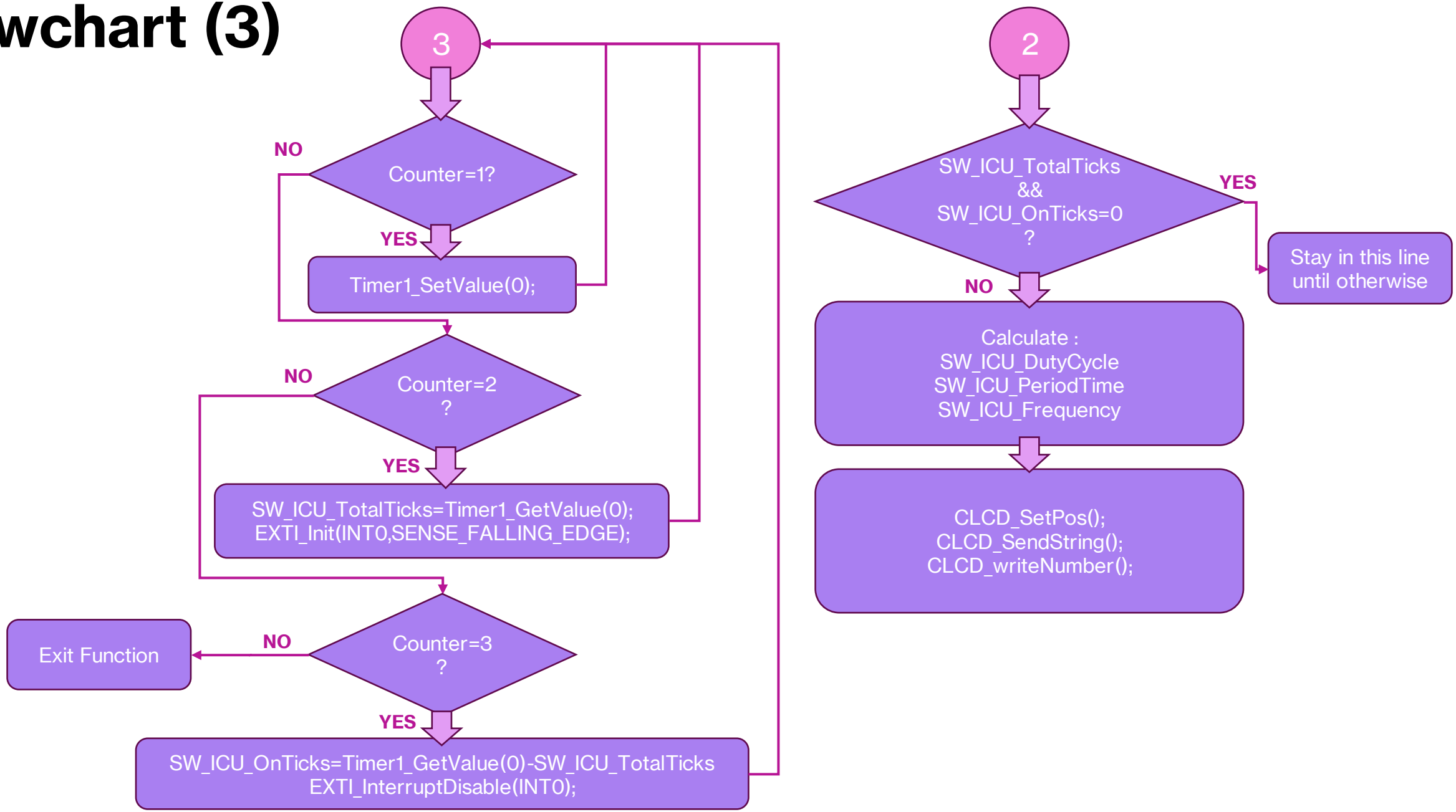
Flowchart (1)


Start
SW_ICU_TotalTicks=0;
SW_ICU_OnTicks=0;
SW_ICU_DutyCycle=0;
SW_ICU_PeriodTime=0
SW_ICU_Frequency=0;
COL_NO=0;
TickTime=0.004;
SW_ICU();
main()
Set INT0 pin as input
Set PortB pin 3 as output for PWM
Set Control LCD pins 0,1,2 as output
Set Data LCD PortA as output
1

Flowchart (2)


1
CLCD_Init();
EXTI_Init(INT0, Sense Rising Edge);
Pin INT0
EXTI_CallBack(&SW_ICU);
Timer0_Init();
Pin B3
Fast PWM Signal
Timer1_Init();
GIE_Init();
while(1)
2
Rising Edge Detected ?
YES
SW_ICU()
Local_counter ++;
3

Flowchart (3)


3
Counter=1?
NO
YES
Timer1_SetValue(0);
Counter=2
?
NO
YES
SW_ICU_TotalTicks=Timer1_GetValue(0);
EXTI_Init(INT0,SENSE_FALLING_EDGE);
Counter=3
?
NO
Exit Function
YES
SW_ICU_OnTicks=Timer1_GetValue(0)-SW_ICU_TotalTicks
EXTI_InterruptDisable(INT0);
2
SW_ICU_TotalTicks
&&
SW_ICU_OnTicks=0
?
YES
Stay in this line
until otherwise
NO
Calculate :
SW_ICU_DutyCycle
SW_ICU_PeriodTime
SW_ICU_Frequency
CLCD_SetPos();
CLCD_SendString();
CLCD_writeNumber();