**1. LED Blinking in Assembly:**

;for GPIO activate the PORT clock

;Configure the DIR register for Direction

;Enable the pin in DEN register

;Write/Read the data in GPIODATA register

;As per STM32F407 datasheet and reference manual

RCC\_AHB1ENR EQU 0x40023830 ;Clock control for AHB1 peripherals (includes GPIO)

;GPIO-D control registers

GPIOD\_MODER EQU 0x40020C00 ;set GPIO pin mode as Input/Output/Analog

GPIOD\_OTYPER EQU 0x40020C04 ;Set GPIO pin type as push-pull or open drain

GPIOD\_OSPEEDR EQU 0x40020C08 ;Set GPIO pin switching speed

GPIOD\_PUPDR EQU 0x40020C0C ;Set GPIO pin pull-up/pull-down

GPIOD\_ODR EQU 0x40020C14 ;GPIO pin output data

;Delay interval

;The delay loop takes 313 nsec to execute at 16MHz

;Time delay = DELAY\_INTERVAL \* 313 nsec

;Overheads are ignored

DELAY\_INTERVAL EQU 0x306008

AREA led\_blink,CODE,READONLY

EXPORT \_\_main

;Port Initialization

\_\_main

BL GPIO\_Init

;Infinite Loop

loop BL LED\_BLINK

BL DELAY

B loop

GPIO\_Init

; Enable GPIO clock

LDR R1, =RCC\_AHB1ENR ;Pseudo-load address in R1

LDR R0, [R1] ;Copy contents at address in R1 to R0

ORR.W R0, #0x08 ;Bitwise OR entire word in R0, result in R0

STR R0, [R1] ;Store R0 contents to address in R1

; Set mode as output

LDR R1, =GPIOD\_MODER ;Two bits per pin so bits 24 to 31 control pins 12 to 15

LDR R0, [R1]

ORR.W R0, #0x55000000 ;Mode bits set to '01' makes the pin mode as output

AND.W R0, #0x55FFFFFF ;OR and AND both operations reqd for 2 bits

STR R0, [R1]

; Set type as push-pull (Default)

LDR R1, =GPIOD\_OTYPER ;Type bit '0' configures pin for push-pull

LDR R0, [R1]

AND.W R0, #0xFFFF0FFF

STR R0, [R1]

; Set Speed slow

LDR R1, =GPIOD\_OSPEEDR ;Two bits per pin so bits 24 to 31 control pins 12 to 15

LDR R0, [R1]

AND.W R0, #0x00FFFFFF ;Speed bits set to '00' configures pin for slow speed

STR R0, [R1]

; Set pull-up

LDR R1, =GPIOD\_PUPDR ;Two bits per pin so bits 24 to 31 control pins 12 to 15

LDR R0, [R1]

AND.W R0, #0x00FFFFFF ;Clear bits to disable pullup/pulldown

STR R0, [R1]

BX LR

;Toggle the PINS to blink LEDs

LED\_BLINK

LDR R1, =GPIOD\_ODR;

MOV R2, #0x0F;

LDR R0,[R1];

EOR R0,R0,R2, LSL #0x0C;

STR R0,[R1]

BX LR

DELAY

LDR r1,=DELAY\_INTERVAL

AGAIN

NOP

NOP

NOP

SUBS r1,r1, #1

BNE AGAIN

BX LR

ALIGN

END

**2. LED Blinking in C:**

#include "stm32f4xx.h"

void configureLED(void);

void msDelay(uint32\_t msTime);

int main(void)

{

//Configure LED

configureLED();

//Define Delay function

msDelay(1000);

while(1)

{

GPIOD->ODR ^= (0xFUL<<12);

msDelay(1000);

}

}

void configureLED(void)

{

RCC->AHB1ENR |=(1UL<<3);

GPIOD->MODER &= ~(0xFFUL<<12\*2);

GPIOD->MODER |= (0x55UL<<12\*2);

}

void msDelay(uint32\_t msTime)

{

/\* For loop takes 4 clock cycles to get executed. Clock frequency of stm32f407 by default is 16MHz

So, 16MHz/4=4MHz. If we want 1000ms delay, 4MHz/1000=4000, so we have to multiply by 4000 to get a delay of 1s

\*/

for(uint32\_t i=0;i<msTime\*4000;i++)

{

\_\_NOP();

}

}