

# **LAB-11 REPORT**

Subject: **Embedded Hardware Design**

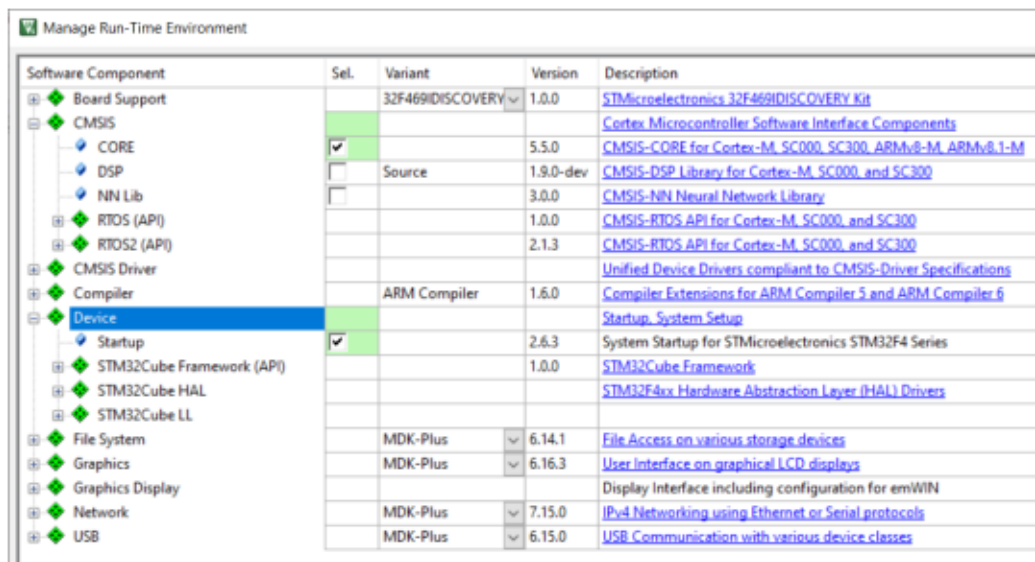
Subject Code: **EL203**

Members: 1) Shantanu Tyagi (**201801015**)  
2) Nikhil Mehta (**201801030**)  
3) Sudhanshu Mishra (**201801114**)  
4) Bhavana Kolli (**201701082**)

**Objective:** Analog-to-Digital driver development and PWM (Pulse-Width\_modulation) Driver Development.

**Method:**

- 1) Creating a new project and setup configuration.
- 2) Select STM32F407VGTx in the search bar shown in the dialog box given below and then select it and click OK.
- 3) Now, tick the checkboxes against the following from the dialog box that appeared on the screen:  
CMSIS → CORE  
Device → Startup
- 4) Click on the Resolve button if any error occurs after ticking the checkboxes.
- 5) Click OK.



**CODE:**

```
//Analog to digital driver developement
#include "stm32f4xx.h" //device header
int analogValue;
//configuring ADH CH1
int main(void){
    RCC->AHB1ENR |= 1;
    GPIOA->MODER |= 0xC; //Setup PA1 to analog
    //Setup ADC1
    RCC->APB2ENR |= 0x100; //Enable clock address to ADC1
    ADC1->CR2 = 0; //TriggerADC1 /Disable ADC1
    ADC1->SQR3 = 1;
```

```

ADC1->CR2 = 1; //Enable ADC1

while(1){
    ADC1->CR2 |= 0x40000000; //start A-to-D conversion
    while(!(ADC1->SR & 2)){ } //wait for conversion to be complete
    analogValue = ADC1->DR;
}
}

//PWM(Pulse Width Modulation) driver developement
#include "stm32f4xx.h" //device header
#define rate 255
void GPIO_Init(void);
void Timer2_Init(void);
void delayMS (int ms);
int main(void){
    GPIO_Init();
    Timer2_Init();

    while(1){
        for(int i=0; i<rate; i++){
            TIM2->CCR1 = i;
            TIM1->CCR2 = rate-1;
            delayMS(5);
        }
        for(int i=255; i>0; i--){
            TIM2->CCR1 = i;
            TIM2->CCR2 = rate-1;
            delayMS(5);
        }
    }
}

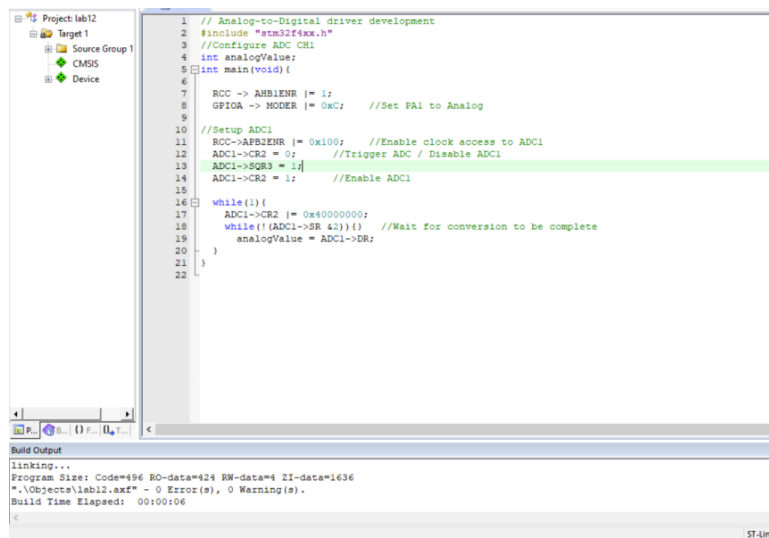
void GPIO_Init(void){
    RCC->AHB1ENR |= RCC_AHB1ENR_GPIOAEN;
    GPIOA->AFR[0] |= (1<<0)|(1<<4);
    GPIOA->MODER |= (1<<3)|(1<<4);
}

void Timer2_Init(void){
    RCC->AHB1ENR |= RCC_AHB1ENR_TIM2EN; //Enable clock access to TM2
    TIM2->PSC = 0; //Set prescaler to 0 ( )
    TIM2->ARR = 255; //Set the maximum count value
    TIM2->CNT = 0; //set the current count
    TIM2->CCMR1 = (1<<5)|(1<<6)|(1<<13)|(1<<14); //configure the pins as PWM
    TIM2->CCER |= 0x11; //enable channel1 and channel2
    TIM2->CR1 = 1; //Enable timer
}

void delayMS (int ms) {
    int i;
    for (; ms>0 ; ms--) {
        for(i=0; i<3195; i++);
    }
}
}

```

## Screenshot:



```
1 // Analog-to-Digital driver development
2 #include "stm32f4xx.h"
3 //Configure ADC CH1
4 int analogValue;
5 int main(void)
6 {
7     RCC->AHB1ENR |= 1; //Set PA1 to Analog
8     GPIOA->MODER |= 0x0C; //Set PA1 to Analog
9
10    //Setup ADC1
11    RCC->APB1ENR |= 0x100; //Enable clock access to ADC1
12    ADC1->CR2 = 0; //Trigger ADC / Disable ADC1
13    ADC1->SQR3 = 1; //Enable ADC1
14    ADC1->CR2 = 1; //Enable ADC1
15
16    while(1){
17        ADC1->CR2 |= 0x40000000;
18        while(!ADC1->SR & 0x1) {} //Wait for conversion to be complete
19        analogValue = ADC1->DR;
20    }
21 }
22
```

Build Output

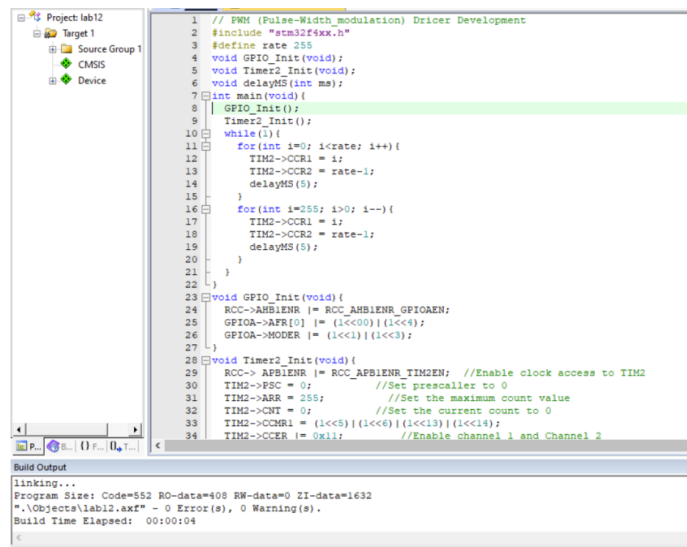
linking...

Program Size: Code=496 RO-data=424 RW-data=4 ZI-data=1636

".\Objects\lab12.axf" - 0 Error(s), 0 Warning(s).

Build Time Elapsed: 00:00:06

## Build Screenshot above for Analog To digital Driver Development



```
1 // PWM (Pulse-Width Modulation) Driver Development
2 #include "stm32f4xx.h"
3 #define rate 255
4 void GPIO_Init(void);
5 void Timer2_Init(void);
6 void delayMS(int ms);
7 int main(void)
8 {
9     GPIO_Init();
10    Timer2_Init();
11    while(1){
12        for(int i=0; i<rate; i++){
13            TIM2->CCR1 = i;
14            TIM2->CCR2 = rate-i;
15            delayMS(5);
16        }
17        for(int i=255; i>0; i--){
18            TIM2->CCR1 = i;
19            TIM2->CCR2 = rate-i;
20            delayMS(5);
21        }
22    }
23 }
24 void GPIO_Init(void){
25     RCC->AHB1ENR |= RCC_AHB1ENR_GPIOAEN;
26     GPIOA->AFR[0] |= (1<<0)|(1<<4);
27     GPIOA->MODER |= (1<<1)|(1<<3);
28 }
29 void Timer2_Init(void){
30     RCC->APB1ENR |= RCC_APB1ENR_TIM2EN; //Enable clock access to TIM2
31     TIM2->PSC = 0; //Set prescaler to 0
32     TIM2->ARR = 255; //Set the maximum count value
33     TIM2->CNT = 0; //Set the current count to 0
34     TIM2->CCR1 = (1<<5)|(1<<6)|(1<<13)|(1<<14);
35     TIM2->CCR2 = 0x11; //Enable channel 1 and Channel 2
36 }
37
```

Build Output

linking...

Program Size: Code=552 RO-data=408 RW-data=0 ZI-data=1632

".\Objects\lab12.axf" - 0 Error(s), 0 Warning(s).

Build Time Elapsed: 00:00:04

## PWM Driver Development

