8.9 Programming in KEIL:

To program the interfaces in Keil, the first step is to create a new uVision Project and selectingthe device which is STM32F103C8 just like we already did in the previous labs.

After saving the project in your desired directory, the device will be selected in the „SelectDevice for Target" dialogue box.

After that, include the libraries as shown in the figure below:

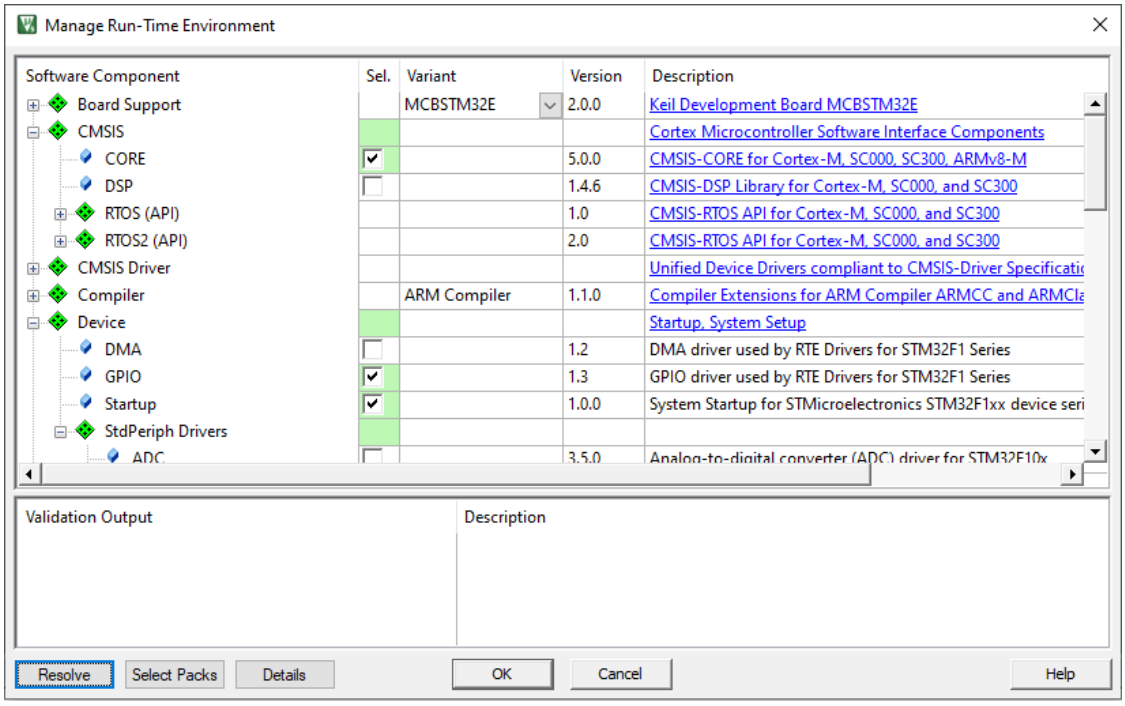


Figure 2: Adding required libraries

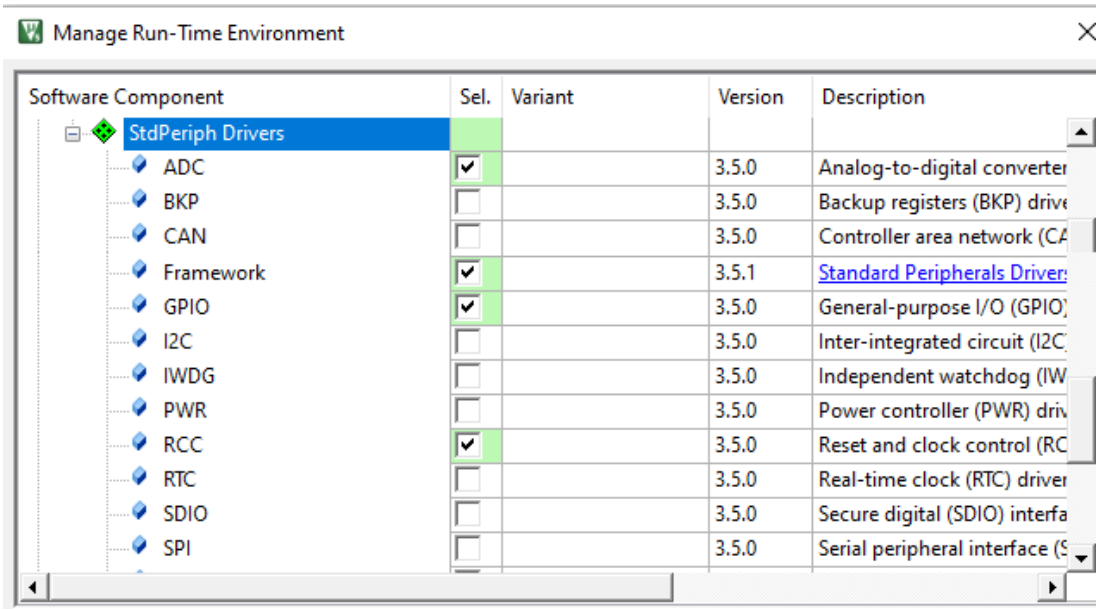


Figure 3: Adding required libraries

Then add a new item into your source group. Select the “C File (.c)” option, name your C fileand save it in your desired directory.

After adding the new C file add the required header files in the Source Group as shown below.

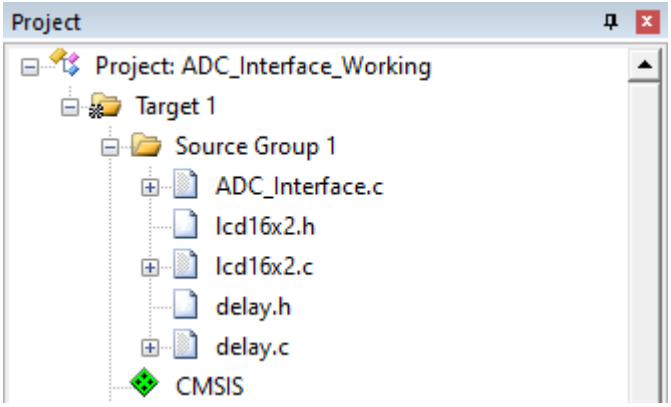


Figure 6: Adding required header files

Lab Task (Psychomotor)

Activity 01:

Write a C-Program that can take an analogue input from a potentiometer, the program will convert the analogue input into a digital output using the ADC of STM32F103 board. Program will convert the digital output into a string variable and displays the voltages on the LCD.

Activity 02:

Write C-Program which will convert the input analogue voltage into a digital output using the ADC of STM32F103 board. Program will write "Temperature =" in the first row of the LCD and will write the digital output (Temperature) converted by ADC in the second row. ADC will be getting the analogue input from a Temperature sensor. Implement the design using STM32F103 board and show the results to lab instructor.

Lab Exercise and Summary

Summary should cover Introduction, Procedure, Data Analysis and Evaluation.

Student's Signature: _____ Date: _____

Lab Task (Cognitive)

Question 1:

The 12-bit ADC of the microcontroller uses VREF+ = 3.3 V and VREF- = 0 V. Use the relation: ADC value = (VIN / VREF) x 4096.

a) Determine the number of quantization levels and the voltage step size (in mV) of the ADC. **[3 Marks]**

b) Determine the ADC output (decimal and hexadecimal) for an input of 1.65 V. **[3 Marks]**

c) An LM35 temperature sensor (10 mV/°C) is connected to the ADC and the conversion result is 372. Determine the input voltage and the temperature. **[4 Marks]**

LABORATORY SKILLS ASSESSMENT (Cognitive)

Obtained Marks	**/10**

Instructor's Signature: _____ Date: _____

LABORATORY SKILLS ASSESSMENT (Psychomotor)

Psychomotor Activity 1: Experimental Setup

S. No	Criteria	Performance	Description	Marks
1	**ADC Hardware & Sensor Setup (4)**	Level 4	Correctly prepares the required ADC hardware and sensor setup for the experiment independently and according to the laboratory requirements.	4
		Level 3	Correctly prepares the required ADC hardware and sensor setup with minor errors or assistance.	3
		Level 2	Partially prepares the required ADC hardware and sensor setup and requires considerable assistance.	2
		Level 1	Shows limited ability to prepare the required ADC hardware and sensor setup and requires substantial assistance.	1
2	**ADC Channel & Clock Configuration (4)**	Level 4	Correctly prepares the required ADC channel and clock configuration for the experiment independently and according to the laboratory requirements.	4
		Level 3	Correctly prepares the required ADC channel and clock configuration with minor errors or assistance.	3
		Level 2	Partially prepares the required ADC channel and clock configuration and requires considerable assistance.	2
		Level 1	Shows limited ability to prepare the required ADC channel and clock configuration and requires substantial assistance.	1
3	**Conversion Parameters (2)**	Level 2	Correctly sets the required conversion parameters for the experiment.	2
		Level 1	Sets the basic conversion parameters with minor errors or assistance.	1
Total Obtained				**/10**

If a student is absent:

Criteria	Description	Marks
Absent	Student is absent from the laboratory activity.	**0**

Psychomotor Activity 2: Implementation + Results

S. No	Criteria	Performance	Description	Marks
1	**ADC C Program (4)**	Level 4	Correctly implements the assigned ADC C program independently while following appropriate laboratory practices.	4
		Level 3	Implements the assigned ADC C program with minor errors or assistance and generally follows proper laboratory practices.	3
		Level 2	Partially implements the assigned ADC C program and requires considerable assistance or repeated guidance.	2
		Level 1	Implementation of the assigned ADC C program is largely incomplete, requires substantial assistance, or shows poor adherence to proper laboratory and safety practices.	1
2	**Execution & Debugging (2)**	Level 2	Correctly performs the required execution and debugging and verifies the expected operation.	2
		Level 1	Performs the basic execution and debugging with minor errors or assistance.	1
3	**ADC Reading & Display Analysis (4)**	Level 4	Obtains correct ADC reading and display analysis and appropriately verifies and interprets the required experimental results.	4
		Level 3	Obtains mostly correct ADC reading and display analysis with minor errors in verification or interpretation.	3
		Level 2	Obtains partially correct ADC reading and display analysis with noticeable errors or omissions.	2
		Level 1	ADC reading and display analysis is largely incorrect or incomplete and requires substantial assistance.	1
Total Obtained				**/10**

If a student is absent:

Criteria	Description	Marks
Absent	Student is absent from the laboratory activity.	**0**

Instructor's Signature: _____ Date: _____

LABORATORY SKILLS ASSESSMENT (Affective)

Affective Activity: Lab Report + Viva Voce

S. No	Criteria	Performance	Description	Marks
1	**Lab Report Completeness & Organization (4)**	Level 4	Report is complete, well-organized, properly presented, and demonstrates responsible completion of the assigned laboratory work.	4
		Level 3	Report is complete with minor omissions or organizational issues and demonstrates satisfactory responsibility for the assigned work.	3
		Level 2	Report is partially complete with several omissions and demonstrates limited responsibility for the assigned work.	2
		Level 1	Report has major omissions or poor organization and demonstrates minimal responsibility for the assigned work.	1
2	**ADC Output & Conclusion (2)**	Level 2	The ADC output and conclusion are correctly presented and discussed, demonstrating ownership of the assigned work.	2
		Level 1	The ADC output and conclusion are presented with minor errors or omissions.	1
3	**Viva Voce (4)**	Level 4	Clearly explains the assigned experiment, ADC resolution, channel configuration, program explanation, working, and results, demonstrating individual responsibility, participation, and effective communication.	4
		Level 3	Explains most of the assigned experiment and results correctly, with minor difficulty, and demonstrates good individual participation.	3
		Level 2	Explains basic aspects of the assigned experiment with some prompting and demonstrates basic participation.	2
		Level 1	Demonstrates limited understanding or involvement and requires substantial prompting to explain the assigned work.	1
Total Obtained				**/10**

The student has not submitted the report and has not appeared in the Viva Voce.

Criteria	Description	Marks
Lab Report and Results Not Submitted	Student has not submitted the report and results file.	**0**
Absent in Viva Voce	Student is absent from the viva voce.	**0**

Instructor's Signature: _____ Date: _____

Experiment No 9

General Purpose Timers in STM32 and their Programming

Objectives:

To have an introduction with all Timers of STM32F103 and to study the General Purpose Timersand their modes in detail and program them using Keil.

Timers in STM32F103:

The STM32F103xx devices include:

- One Advanced-control timers (TIM-1)
- Three general-purpose timers (TIM-2, TIM3 & TIM-4) All timer counters can be frozen in debug mode.

Peripheral		STM32F103Cx		STM32F103Rx			STM32F103Vx	
Flash - Kbytes		32	64	32	64	128	64	128
SRAM - Kbytes		10	20	10	20		20	
Timers	General purpose	2	3	2	3		3	
	Advanced Control	1		1			1	

Figure 1: Peripherals of STM32F103xx

General-purpose timers:

There are up to 3 synchronizable standard timers embedded in the STM32F103xx performance line devices. These timers are based on a 16-bit auto-reload up/down counter, a 16-bit prescaler and feature 4 independent channels each for input capture/output compare, PWM or one pulse mode output. This gives up to 12 input captures / output compares / PWMs on the largest packages. They can work together with the Advanced Control Timer via the Timer Link feature for synchronization or event chaining. The counter can be frozen in debug mode.

TIM2 to TIM4 introduction:

The general-purpose timers consist of a 16-bit auto-reload counter driven by a programmable prescaler. They may be used for a variety of purposes, including measuring the pulse lengths of input signals (input capture) or generating output waveforms (output compare and PWM). Pulse lengths and waveform periods can be modulated from a few microseconds to several milliseconds using the timer prescaler and the RCC clock controller prescalers. The timers are completely independent, and do not share any resources. They can be synchronized together

TIMx main features:

General-purpose TIMx timer features include:

- 16-bit up, down, up/down auto-reload counter.
- 16-bit programmable prescaler used to divide (also “on the fly”) the counter clock frequency byany factor between 1 and 65536.
- Up to 4 independent channels for:
  - Input capture
  - Output compare
  - PWM generation (Edge- and Center-aligned modes)
  - One-pulse mode output
- Synchronization circuit to control the timer with external signals and to interconnect severaltimers.
- Interrupt/DMA generation on the following events:
  - Update: counter overflow/underflow, counter initialization (by software or internal/external trigger)
  - Trigger event (counter start, stop, initialization or count by internal/external trigger)
  - Input capture
  - Output compare
- Supports incremental (quadrature) encoder and hall-sensor circuitry for positioning purposes
- Trigger input for external clock or cycle-by-cycle current management

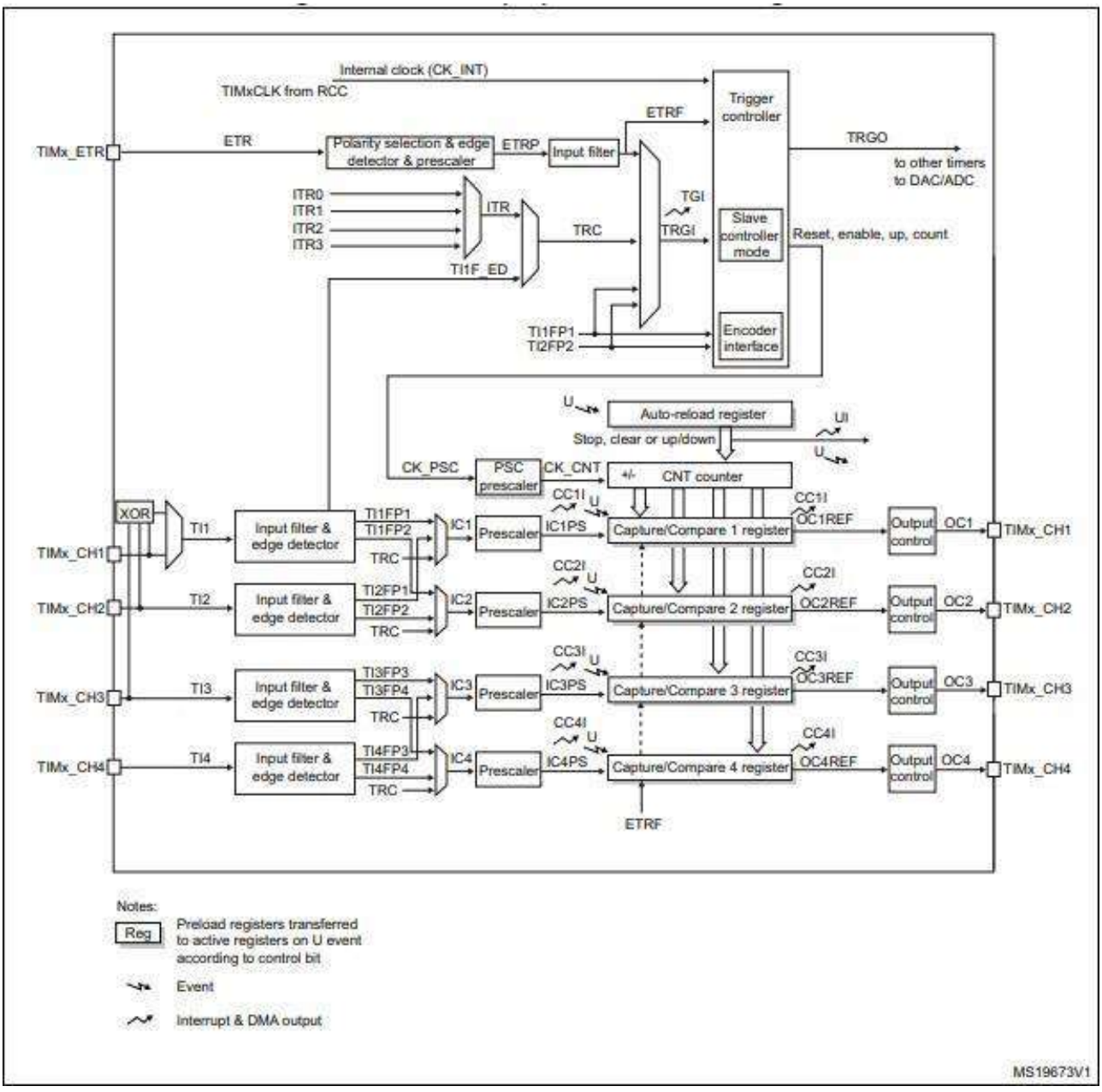


Figure 2: General Purpose Timer Block Diagram

Timer Simulation in Keil:

First of all, you need to implement the following code in Keil.

```
#include "stm32f10x.h"
#include "stm32f10x_rcc.h"
#include "stm32f10x_tim.h"
#include "stm32f10x_gpio.h"

void LED_Config();
void delay(unsigned int nCount);
void TIM_config(void);

TIM_TimeBaseInitTypeDef MyTimer;
GPIO_InitTypeDef My_LED;

int main (void)
{

    LED_Config();
    TIM_config();

    TIM_Cmd(TIM3,ENABLE);

    while (1)
    {
      if(TIM_GetCounter(TIM3)> 0 && TIM_GetCounter(TIM3) < 7999)
      {
        GPIO_ResetBits(GPIOA, GPIO_Pin_1);
      }
      else
      {
        GPIO_SetBits(GPIOA, GPIO_Pin_1);
      }
    }
}

// Function for LED_Configuration
void LED_Config()
{
     // Enable clock for GPIOA
     RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA, ENABLE);

     // Configure PA1 as push-pull output
     My_LED.GPIO_Pin = GPIO_Pin_1;
     My_LED.GPIO_Speed = GPIO_Speed_2MHz;
     My_LED.GPIO_Mode = GPIO_Mode_Out_PP;

     GPIO_Init(GPIOA, &My_LED);
}
```

```
48  void TIM_config(void)
49  {
50      RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM3,ENABLE);
51
52
53      MyTimer.TIM_ClockDivision = TIM_CKD_DIV1;
54      MyTimer.TIM_CounterMode = TIM_CounterMode_Up;
55      MyTimer.TIM_RepetitionCounter=0;
56
57      MyTimer.TIM_Prescaler =499;
58      MyTimer.TIM_Period=15999;
59
60      TIM_TimeBaseInit(TIM3,&MyTimer);
61
62  }
63
64  // Delay function
65  void delay(unsigned int nCount)
66  {
67    unsigned int i, j;
68      for (i = 0; i < nCount; i++)
69        for (j = 0; j < 0x2AFF; j++);
70  }
```

Lab Task (Psycomotor)

Activity 01:

Build and Run the given Code.

- **Follow the steps and Simulate your code and show the output.**
- **Observe the output and evaluate that how many times the LED will blink in 15 seconds. Write the time period as well.**