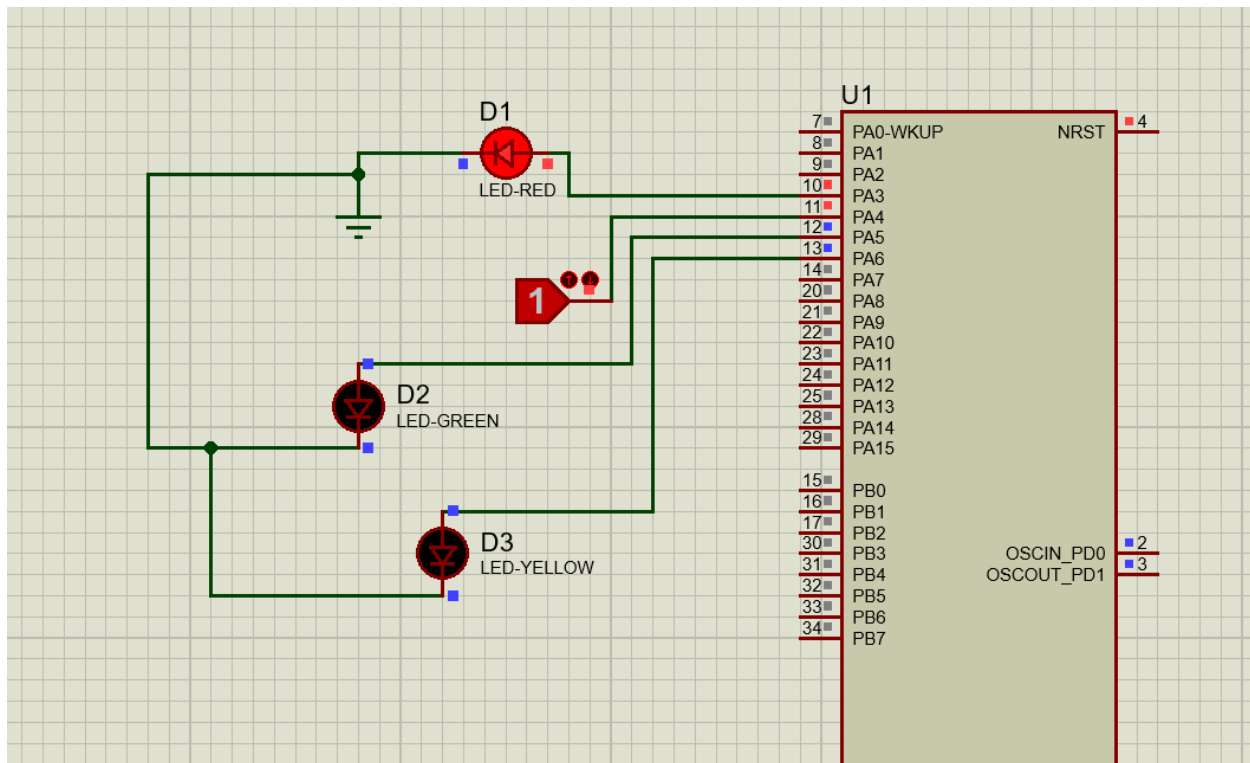


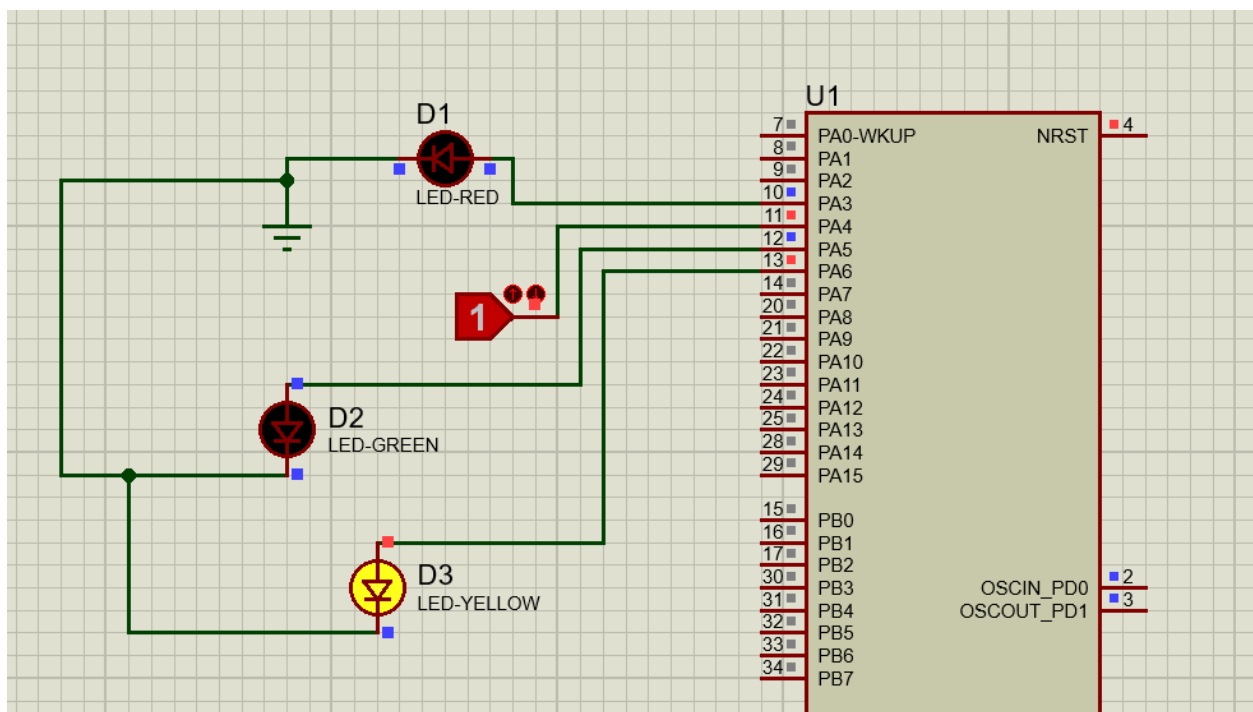
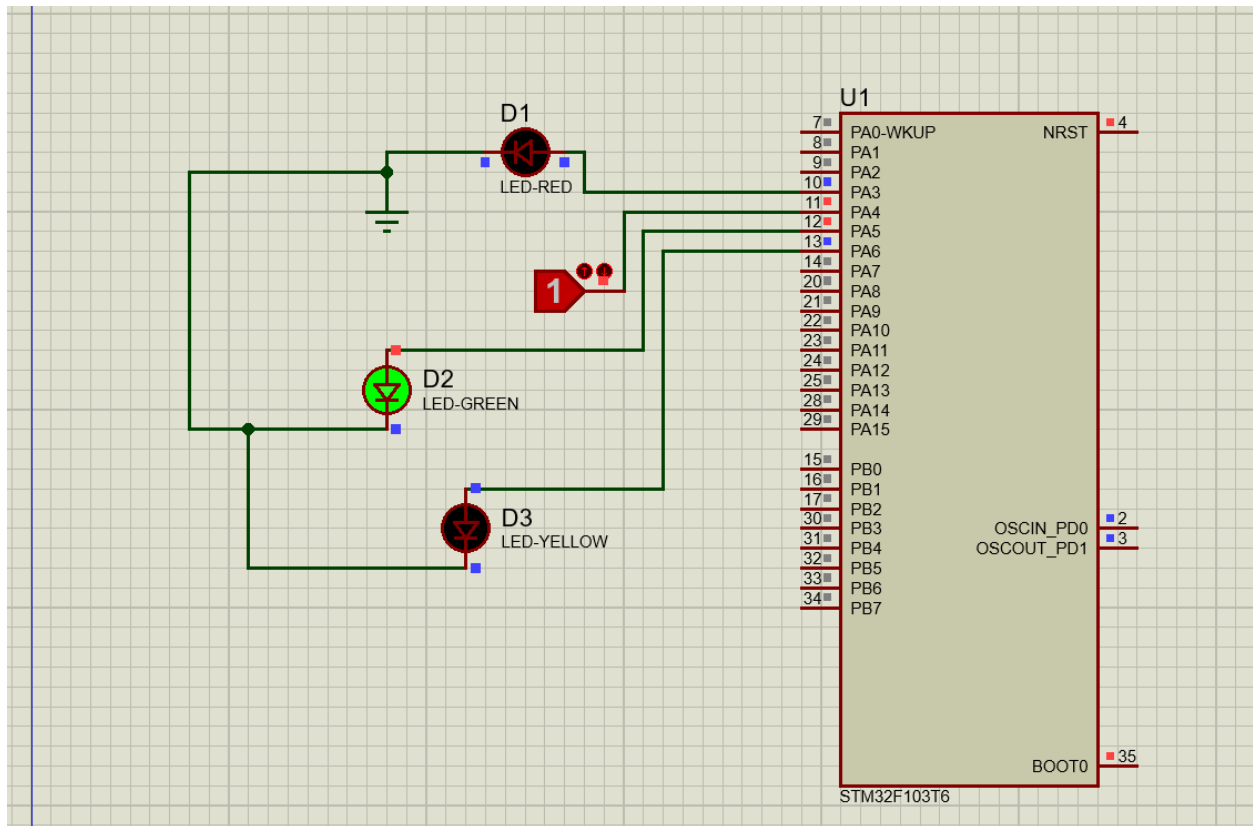
به نام خدا

امیرعباس اجیل چی ۸۱۰۱۰۱۳۶۹

سوال اول)

در این سوال یک پایه را برای کلید و سه پایه را برای خروجی تنظیم میکنیم حالا که تنظیمات انجام شد خروجی به شکل زیر میشود





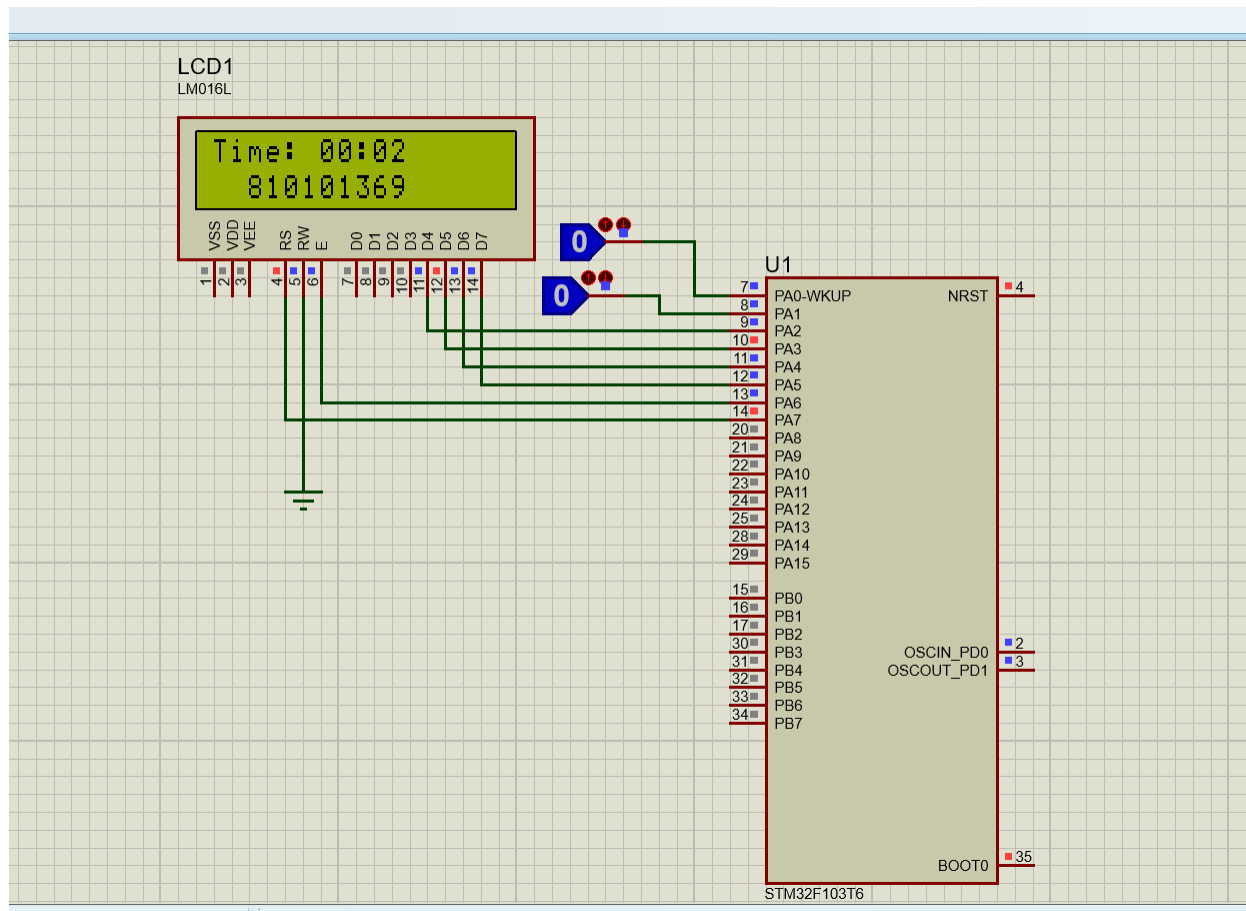
سوال دوم)

در این سوال تنظیمات مربوطه را انجام میدیم (از توضیحات اضافی خودداری میکنم)

صرفاً اعداد را باید ۱۹۹۹۹ و ۹۹۹ تنظیم کنیم تا فرکانس یک هرتز و تایمر به شکل درست بشمارد

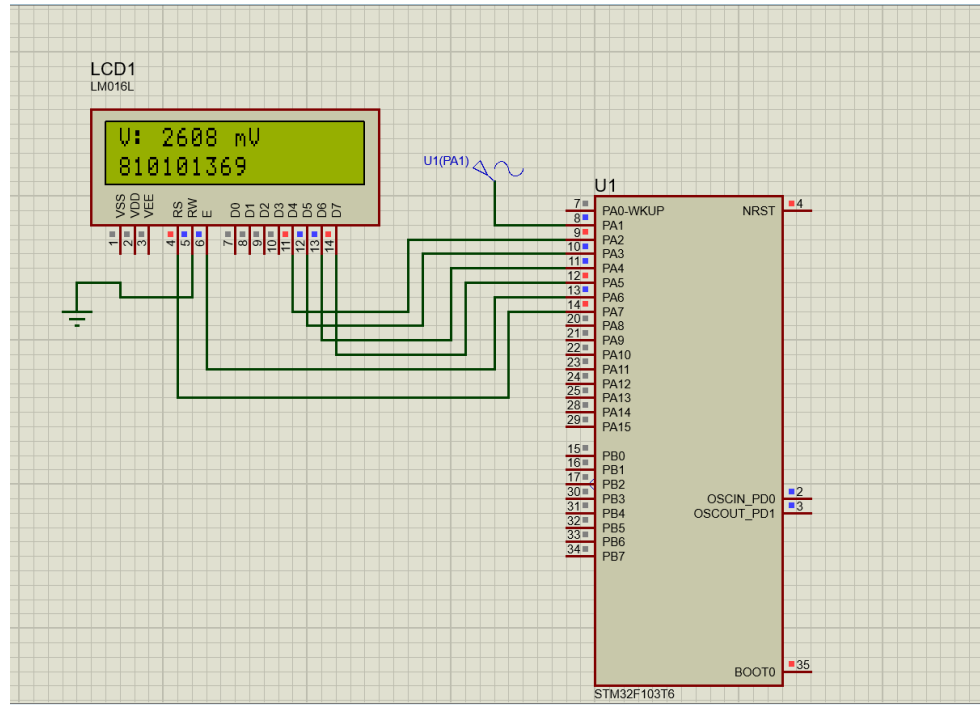
خروجی نهایی نیز به شکل زیر است

بعد از فشردن کلید استارت شروع میکند و بعد از فشردن استپ ال سی دی متوقف میشه و کلمه *stopped* نمایش داده میشه



سوال سوم)

خب در این سوال هم تنظیمات رو انجام میدیم و به خروجی زیر میرسیم

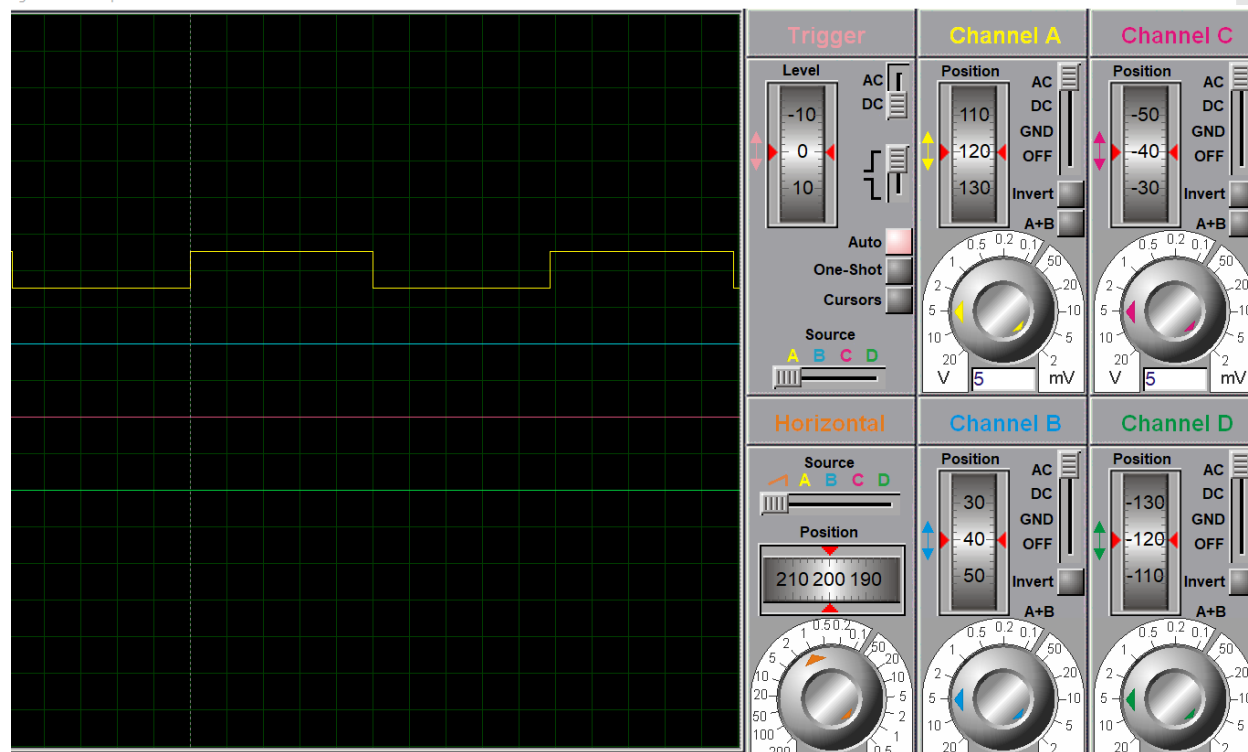


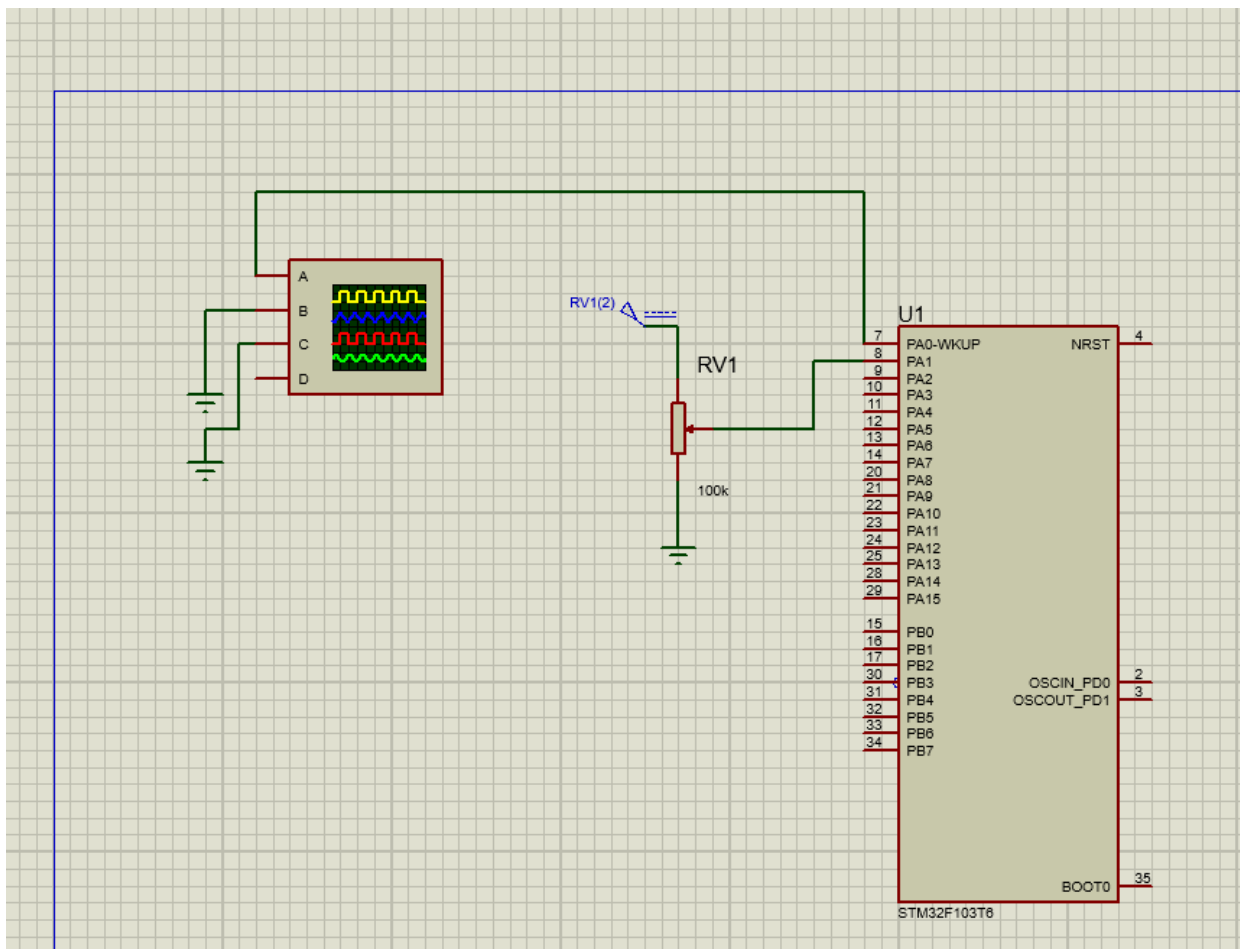
```

3.ioc  main.c x
9  /* Initialize all configured peripherals */
0  MX_GPIO_Init();
1  MX_ADC1_Init();
2  /* USER CODE BEGIN 2 */
3  LCD16X2_Init(MyLCD);
4  HAL_Delay(1000);
5  LCD16X2_Clear(MyLCD);
6  uint32_t adc_val=0;
7  uint32_t voltage=0;
8  char buffer[16];
9
0  /* USER CODE END 2 */
1
2  /* Infinite loop */
3  /* USER CODE BEGIN WHILE */
4  while (1)
5  {
6
7      HAL_ADC_Start(&hadc1);
8      HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
9      adc_val=HAL_ADC_GetValue(&hadc1);
0      voltage=(adc_val*3300)/4095;
1      LCD16X2_Set_Cursor(MyLCD, 1, 1);
2      sprintf(buffer,"V: %1u mV",voltage);
3      LCD16X2_Write_String(MyLCD, buffer);
4      LCD16X2_Set_Cursor(MyLCD, 2, 1);
5      LCD16X2_Write_String(MyLCD, "810101369");
6      HAL_Delay(1000);
7

```

Digital Oscilloscope





تغییراتی که در کل اعمال شده نیز به شکل زیر است

```

32  /* Private function prototypes
33  void SystemClock_Config(void);
34  static void MX_GPIO_Init(void);
35  static void MX_ADC1_Init(void);
36  static void MX_TIM2_Init(void);
37  /* USER CODE BEGIN PFP */
38  void SystemClock_Config(void);
39  void MX_GPIO_Init(void);
40  void MX_ADC1_Init(void);
41  void MX_TIM2_Init(void);
42  /* USER CODE END PFP */
43  /* Private user code -----
44  */
45  /* Private define -----
46  */
47  /* Private macro -----
48  */
49  /* Private variables -----
50  */
51  /* Private user code -----
52  */
53  /* Private user code -----
54  */
55  /* Private user code -----
56  */
57  /* Private user code -----
58  */
59  /* Private user code -----
60  */
61  /* Private user code -----
62  */
63  /* Private user code -----
64  */
65  /* Private user code -----
66  */
67  /* Private user code -----
68  */
69  /* Private user code -----
70  */
71  /* Private user code -----
72  */
73  /* Private user code -----
74  */
75  /* Private user code -----
76  */
77  /* Private user code -----
78  */
79  /* Private user code -----
80  */
81  /* Private user code -----
82  */
83  /* Private user code -----
84  */
85  /* Private user code -----
86  */
87  /* Private user code -----
88  */
89  /* Private user code -----
90  */
91  /* Private user code -----
92  */
93  /* Private user code -----
94  */
95  /* Private user code -----
96  */
97  /* Private user code -----
98  */
99  /* Private user code -----
100  */

```

بخش main به شکل زیر است

```

171  * @retval None
172  */
173 void MX_ADC1_Init(void)
174 {
175     __HAL_RCC_ADC1_CLK_ENABLE();
176
177     hadc1.Instance = ADC1;
178     hadc1.Init.ScanConvMode = ADC_SCAN_DISABLE;
179     hadc1.Init.ContinuousConvMode = DISABLE;
180     hadc1.Init.DiscontinuousConvMode = DISABLE;
181     hadc1.Init.ExternalTrigConv = ADC_SOFTWARE_START;
182     hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
183     hadc1.Init.NbrOfConversion = 1;
184     HAL_ADC_Init(&hadc1);
185
186     ADC_ChannelConfTypeDef sConfig = {0};
187     sConfig.Channel = ADC_CHANNEL_1;
188     sConfig.Rank = ADC_REGULAR_RANK_1;
189     sConfig.SamplingTime = ADC_SAMPLETIME_239CYCLES_5;
190     HAL_ADC_ConfigChannel(&hadc1, &sConfig);
191 }

```

```

/* Initialize all configured peripherals */
MX_GPIO_Init();
MX_ADC1_Init();
MX_TIM2_Init();
/* USER CODE BEGIN 2 */
HAL_TIM_PWM_Start(&htim2, TIM_CHANNEL_1);
/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    uint32_t adc_val=0;
    uint32_t duty =0;
    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
    adc_val=HAL_ADC_GetValue(&hadc1);
    duty = 100+(adc_val * 800 )/4095;
    htim2.Instance->CCR1=duty;
    HAL_Delay(100);
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */

```

```

6  * @retval None
7  */
8 void MX_TIM2_Init(void)
9 {
10     __HAL_RCC_TIM2_CLK_ENABLE();
11
12     htim2.Instance = TIM2;
13     htim2.Init.Prescaler = 199;
14     htim2.Init.CounterMode = TIM_COUNTERMODE_UP;
15     htim2.Init.Period = 999;
16     htim2.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
17     htim2.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE;
18     HAL_TIM_PWM_Init(&htim2);
19
20     TIM_OC_InitTypeDef sConfigOC = {0};
21     sConfigOC.OCMode = TIM_OCMODE_PWM1;
22     sConfigOC.Pulse = 400; // 40 % دیوتی سائیکل اولیہ
23     sConfigOC.OCPolarity = TIM_OCPOLARITY_HIGH;
24     sConfigOC.OCFastMode = TIM_OCFAST_DISABLE;
25
26     HAL_TIM_PWM_ConfigChannel
27 }
28 /**

```

```

/** @defgroup TIM_Output_Fast_State TIM Output F
 * @{
 */
#define TIM_OCFAST_DISABLE 0x0000

```