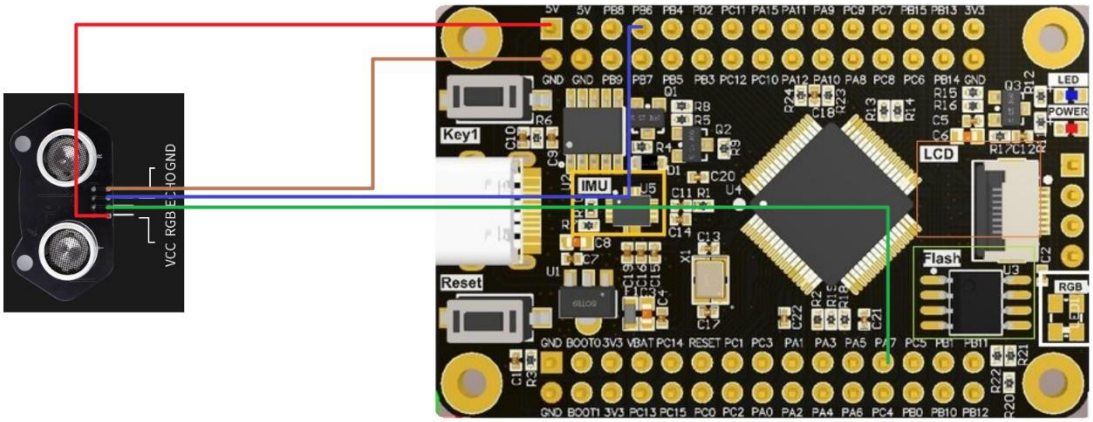
Colorful ultrasonic

1. Learning objectives

In this course, we will learn how to use STM32F103RCT6 to control colorful ultrasonic module.

2. Prepare before class

The hardware required is several DuPont lines, STM32F103RCT6 development board, and colorful ultrasonic module.



Colorful ultrasound	STM32F103RCT6
VCC	5V
RGB	PA7
ECHO	PB6
GND	GND

Ultrasonic

Code interpretation

Ultrasonic trigger and receive pin initialization

```
void ultasonic_Trig(void)
{
  GPIO_InitTypeDef GPIO_InitStructure;
  RCC_APB2PeriphClockCmd(TRIG_RCC,ENABLE);
  GPIO_InitStructure.GPIO_Mode=GPIO_Mode_Out_PP;
  GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz;
  GPIO_InitStructure.GPIO_Pin=TRIG_PIN;
  GPIO_Init(TRIG_PORT,&GPIO_InitStructure);
  GPIO_ResetBits(TRIG_PORT,TRIG_PIN);
}

void  ultasonic_Echo(void)
{
  GPIO_InitTypeDef GPIO_InitStructure;
  RCC_APB2PeriphClockCmd(ECHO_RCC,ENABLE);
  GPIO_InitStructure.GPIO_Mode=GPIO_Mode_IN_FLOATING;
  GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz;
  GPIO_InitStructure.GPIO_Pin=ECHO_PIN;
  GPIO_Init(ECHO_PORT,&GPIO_InitStructure);
}
```

Ultrasound acquisition distance function

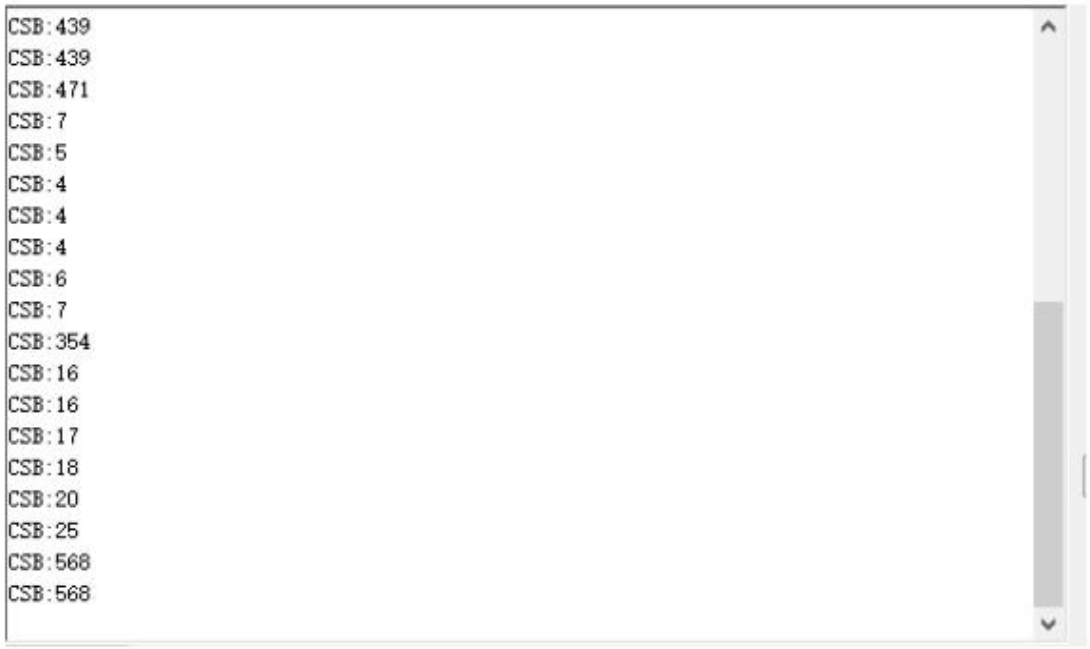
```
30  float bsp_getUltrasonicDistance(void)
31  {
32    float length = 0, sum = 0;
33    u16 tim;
34    unsigned int  i = 0;
35
36    while(i != 5)
37    {
38      ultasonic_Trig();
39      GPIO_SetBits(TRIG_PORT, TRIG_PIN);
40      delay_us(20);
41      GPIO_ResetBits(TRIG_PORT, TRIG_PIN);
42
43      ultasonic_Echo();
44      while(GPIO_ReadInputDataBit(ECHO_PORT, ECHO_PIN) == RESET);
45      TIM_Cmd(TIM3,ENABLE);
46
47      i+=1;
48      while(GPIO_ReadInputDataBit(ECHO_PORT, ECHO_PIN) == SET);
49      TIM_Cmd(TIM3, DISABLE);
50
51      tim = TIM_GetCounter(TIM3);
52
53      length = (tim + overcount * 1000) / 58.0;
54
55      sum = length + sum;
56      TIM3->CNT = 0;
57      overcount = 0;
58      delay_ms(1);
59    }
60    length = sum / 5;
61    return length;
62  }
```

Timer initialization

```
65  void bsp_Ultrasonic_Timer3_Init(void)
66  {
67    TIM_TimeBaseInitTypeDef TIM_TimeBaseInitStructer;
68    NVIC_InitTypeDef NVIC_InitStructer;
69
70
71    RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM3, ENABLE);
72
73    TIM_DeInit(TIM3);
74    TIM_TimeBaseInitStructer.TIM_Period = 999;
75    TIM_TimeBaseInitStructer.TIM_Prescaler = 71;
76    TIM_TimeBaseInitStructer.TIM_ClockDivision = TIM_CKD_DIV1;
77    TIM_TimeBaseInitStructer.TIM_CounterMode = TIM_CounterMode_Up;
78    TIM_TimeBaseInit(TIM3, &TIM_TimeBaseInitStructer);
79
80    TIM_ITConfig(TIM3, TIM_IT_Update, ENABLE);
81
82    NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);
83
84    NVIC_InitStructer.NVIC_IRQChannelPreemptionPriority = 0;
85    NVIC_InitStructer.NVIC_IRQChannelSubPriority = 0;
86    NVIC_InitStructer.NVIC_IRQChannel = TIM3_IRQn;
87    NVIC_InitStructer.NVIC_IRQChannelCmd = ENABLE;
88
89    NVIC_Init(&NVIC_InitStructer);
90    TIM_Cmd(TIM3, DISABLE);
91
92  }
93
94  void TIM3_IRQHandler(void)
95  {
96    if(TIM_GetITStatus(TIM3,TIM_IT_Update) != RESET)
97    {
98      TIM_ClearITPendingBit(TIM3, TIM_IT_Update);
99      overcount++;
100   }
101 }
```

Experimental phenomenon

After flashing the program, press the reset key, and the baud rate is 9600. The serial port shows the ultrasonic ranging distance.



RGB Lights

Code interpretation

GPIO PA7 initialization

```
void ws2812_GPIO_Init(void){
  GPIO_InitTypeDef    GPIO_InitStructure;

  RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA, ENABLE);

  GPIO_InitStructure.GPIO_Pin = GPIO_Pin_7;
  GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF_PP;
  GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
  GPIO_Init(GPIOA, &GPIO_InitStructure);
}
```

WS2812 initialization

```
void ws2812_Init(void){
  ws2812_GPIO_Init();
  ws2812_SPI_Init();
  ws2812_DMA_Init();
  ws2812_AllShutOff();
  delay_ms(WS2812_LED_NUM * 10);
}
```

Breathing lamp implementation function

```
void ws2812_All_LED_one_Color_breath(uint16_t interval_time, uint32_t GRB_color){
  uint8_t i = 0;
  uint16_t j = 0;
  rgb_color.G = GRB_color>>16;
  rgb_color.R = GRB_color>>8;
  rgb_color.B = GRB_color;
  for(i=1;i<=100;i++){
    __brightnessAdjust(i/100.0f, rgb_color);
    for(j=0;j<WS2812_LED_NUM;j++){
      ws2812_Set_one_LED_Color(j, ((rgb_color.G<<16) | (rgb_color.R<<8) | (rgb_color.B)));
    }
    ws2812_Send_Data();
    delay_ms(interval_time);
  }
  for(i=100;i>=1;i--){
    __brightnessAdjust(i/100.0f, rgb_color);
    for(j=0;j<WS2812_LED_NUM;j++){
      ws2812_Set_one_LED_Color(j, ((rgb_color.G<<16) | (rgb_color.R<<8) | (rgb_color.B)));
    }
    ws2812_Send_Data();
    delay_ms(interval_time);
  }
}
```

The marquee implementation function

```
void horse_race_lamp(uint16_t interval_time)
{
  u8 i,color;


  for(i = 0; i < WS2812_LED_NUM; i++)
  {
//    ws281x_setPixelRGB(i,255,255,0);
    color = rand()%7;
    set_pixel_rgb(i,color);
    ws281x_ShutoffPixel(i-1);
    delay_ms(interval_time);
  }
  ws281x_ShutoffPixel(WS2812_LED_NUM-1);
  delay_ms(interval_time);
}
```

The flow light implements the function.

```
void Running_water_lamp( uint8_t green ,uint8_t red ,uint8_t blue, uint16_t interval_time )
{
  uint16_t i;

  for(i = 0; i < WS2812_LED_NUM; i++)
  {
    ws281x_setPixelRGB(i,green,red,blue);
    delay_ms(interval_time);
  }
  ws2812_AllShutOff();
  delay_ms(interval_time);
}
```

Light up all lights.

```
void ws2812_AllOpen(uint8_t red ,uint8_t green ,uint8_t blue)
{
  uint16_t i,j;

  for(j = 0;j<WS2812_LED_NUM;j++)
  {
    for(i = 0; i < 24; ++i)
    {
      ws2812_data_buffer[j][i] = (((ws281x_color(red,green,blue) << i) & 0X800000) ? SIG_1 : SIG_0);
    }
  }
  ws2812_Send_Data();
  delay_ms(10);
}
```

Randomly turn on a light

```
void srand_lamp(uint16_t interval_time)
{
  static uint8_t tmp,i;
  uint8_t k,color;

  tmp = rand()%(WS2812_LED_NUM);
  color = rand()%7;
  if(i==0)
  {
    memset(tmp_flag,50,WS2812_LED_NUM);
    tmp_flag[i] = tmp;
    set_pixel_rgb(tmp,color);
    delay_ms(interval_time);
    i++;

  }
  else if(i>=WS2812_LED_NUM)
  {
    return ;
  }

  for(k=0;k<i;k++)
  {
    if(tmp == tmp_flag[k])
    {
      return ;
    }

  }
```

2. Experimental phenomenon

After flashing the program, press the reset key; Colorful ultrasound achieves different RGB effects.