

Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh

Khoa Cơ khí Chế tạo Máy

Bộ môn Cơ Điện tử

# BÁO CÁO DỰ ÁN

## 1. Thông tin người thực hiện:

STT	Người thực hiện	MSSV	Ngày
01	Nguyễn Hữu Chí	20146479	17/5/2023

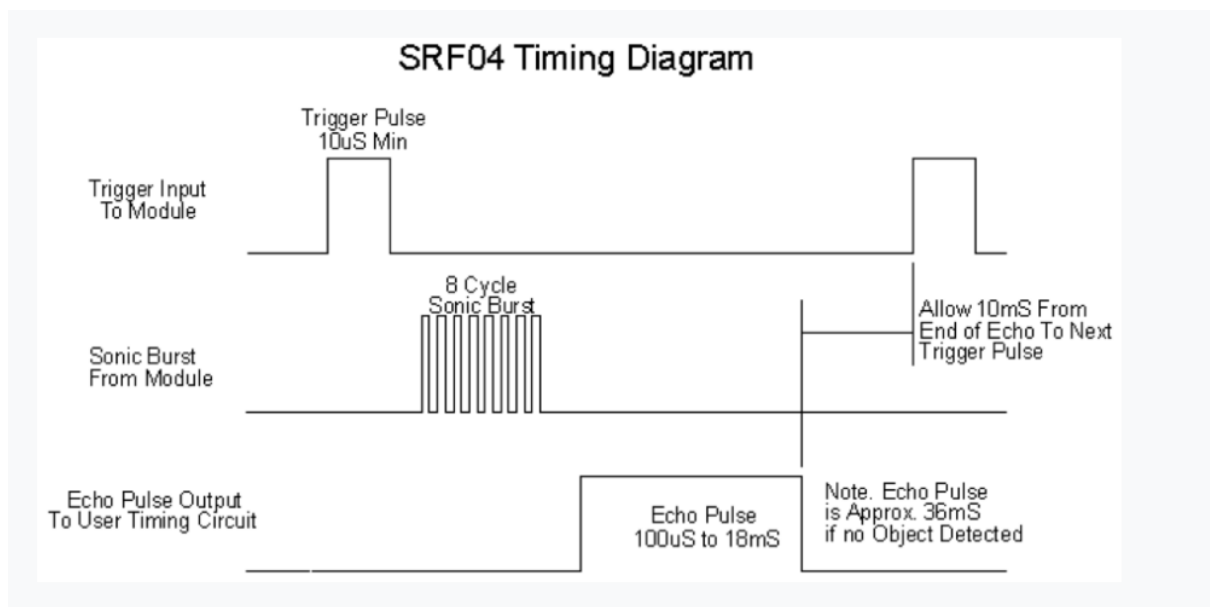
## 2. Yêu cầu dự án

- Tạo dự án lập trình vi điều khiển STM32 sử dụng timer để giao tiếp cảm biến SRF04 đo khoảng cách. Kết quả đo tính theo milimet và in ra UART1

## 3. Nội dung:

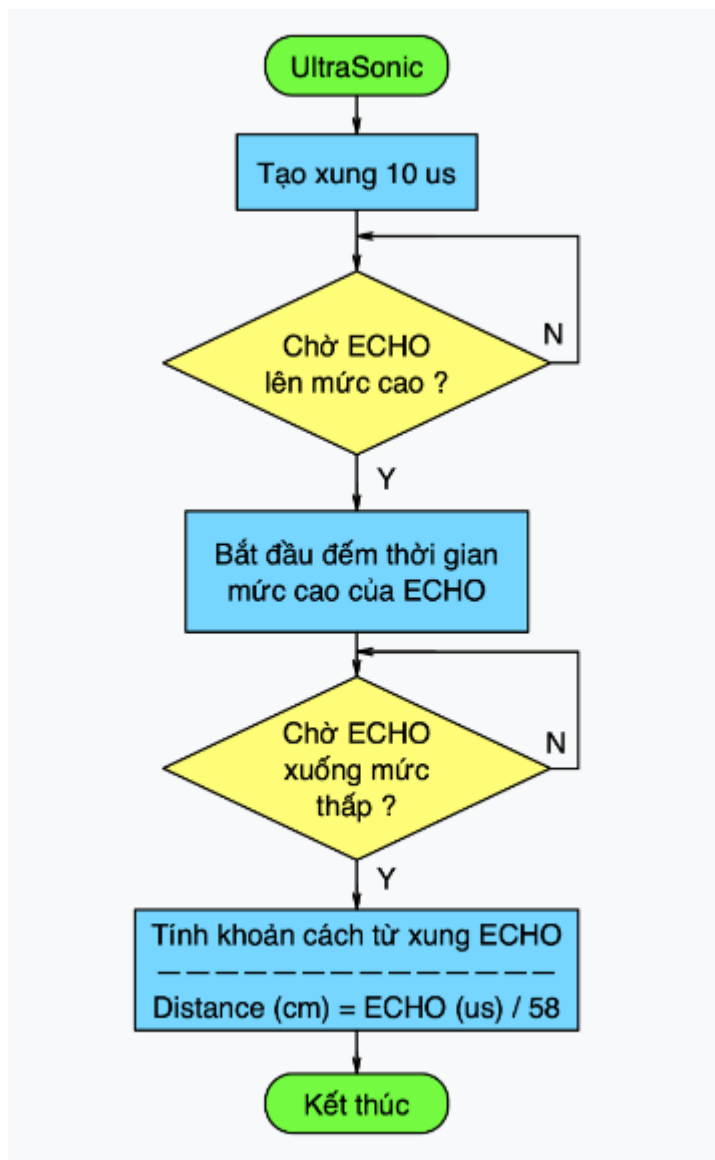
### 3.1. Phân tích dự án:

- Để giải quyết được bài toán ta cần làm theo datasheet của nhà sản xuất
- Khởi tạo SRF04 theo hình dưới đây



- Sau bước trên ta sẽ đo được thời gian  $t = \text{số ticks} * 0.25\mu s$
- Sử dụng công thức tính  $s = v * t'$  để đo được khoảng cách ( $v$  là vận tốc âm thanh,  $t' = t/2$  do thời gian truyền đi = thời gian nhận về)

### 3.2. Lưu đồ lập trình:



### 3.3. Mã nguồn chương trình:

- Tạo hàm printf để truyền giá trị qua UART

```
52  
53  /* USER CODE BEGIN PV */  
54  FILE __stdout;  
55  int fputc(int ch, FILE *f)  
56  {  
57      HAL_UART_Transmit(&huart1, (uint8_t *)&ch, 1, 100);  
58      return ch;  
59  }  
60
```

- Khởi động timer, khai báo các biến cần thiết

```
107 /* USER CODE BEGIN 2 */
108 HAL_TIM_Base_Start(&htim1); // start timer
109 uint16_t EchoCnt = 0;
110 float MeasuredDistance = 0; // in mm
111 /* USER CODE END 2 */
```

- Phần tiếp theo lập trình theo flowchart

- Trong vòng lặp chính. Đầu tiên cho chân TRIG xuống 0, cho bộ đếm bắt đầu đếm từ 0, tiếp theo cho chân TRIG lên 1

```
114 /* USER CODE BEGIN WHILE */
115 while (1)
116 {
117     /* USER CODE END WHILE */
118     HAL_GPIO_WritePin(TRIG_GPIO_Port,TRIG_Pin,0);
119     __HAL_TIM_SET_COUNTER(&htim1,0);
120     HAL_GPIO_WritePin(TRIG_GPIO_Port,TRIG_Pin,1);
```

- Chờ counter đếm (đếm ít nhất 40 ticks = 10us mới qua câu lệnh tiếp theo). Do cấu hình vi điều khiển thạch anh ngoại 4MHz tương đương 1 tick =  $1/4\text{MHz} = 0.25\mu\text{s}$

```
121 while (__HAL_TIM_GET_COUNTER(&htim1) < 49) { // min 10us -> min 40 ticks
122     // empty
123 }
```

- Cho chân TRIG xuống 0

```
123 }
124 HAL_GPIO_WritePin(TRIG_GPIO_Port,TRIG_Pin,0);
125
```

- Chờ giá trị chân ECHO khác 0 thì counter bắt đầu đếm từ 0, trong quá trình đếm nếu chân ECHO = 0 thì sẽ lưu giá trị của counter tại thời điểm đó cho biến Echocnt

```
126 while(HAL_GPIO_ReadPin(ECHO_GPIO_Port,ECHO_Pin)==0); // do not thing, wait for ECHO = 1
127 __HAL_TIM_SET_COUNTER(&htim1,0);
128 while(HAL_GPIO_ReadPin(ECHO_GPIO_Port,ECHO_Pin)==1); // do not thing, wait for ECHO = 0
129 EchoCnt = __HAL_TIM_GET_COUNTER(&htim1);
```

- Dùng công thức để tính khoảng cách từ giá trị Echocnt đo được. Truyền giá trị này qua UART. Delay 500ms giữa 2 lần đo

```
130 MeasuredDistance = (EchoCnt*0.25)*10/58; // 1 lần đếm 0.25 us
131 printf("D= %f mm ", MeasuredDistance);
132 HAL_Delay(500);
133 /* USER CODE BEGIN 3 */
```

## 4. Ghi chú khác (nếu có)