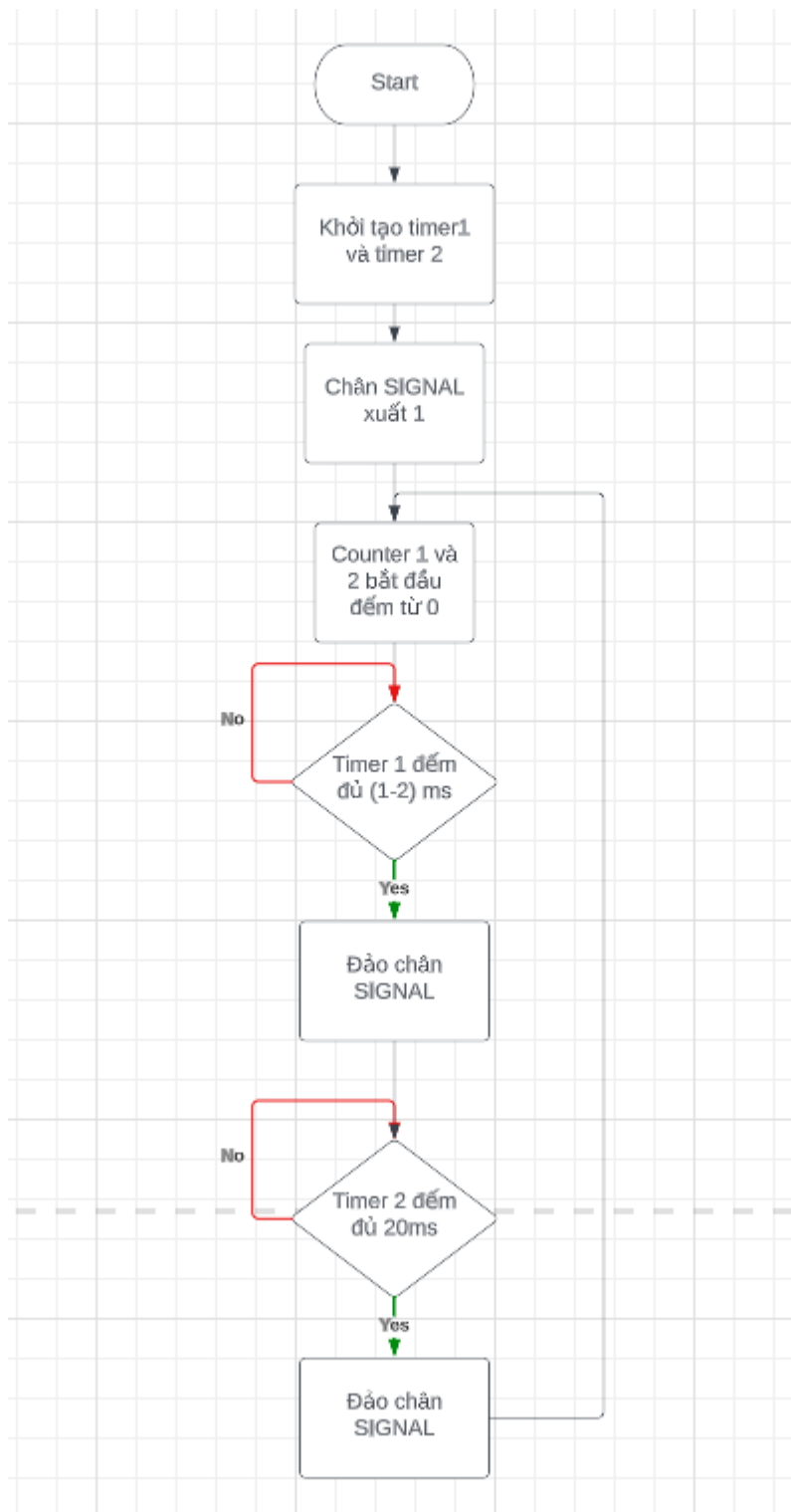


### 3.2. Lưu đồ lập trình:



### 3.3. Mã nguồn chương trình:

- Cấu hình 2 timer đếm 1 tick = 0.25us
- Tạo hàm RCservo có biến đầu vào là góc quay

```
60  /* USER CODE BEGIN 0 */
61  void RCservo(float Angle) {
62      __HAL_TIM_SET_COUNTER(&htim1,0);
63      __HAL_TIM_SET_COUNTER(&htim2,0);
64      while(__HAL_TIM_GET_COUNTER(&htim1) < 6000+(uint16_t)(200*Angle/9)) {
65          // empty
66      }
67      HAL_GPIO_TogglePin(SIGNAL_GPIO_Port, SIGNAL_Pin);
68
69      while(__HAL_TIM_GET_COUNTER(&htim2) < 40000) {
70          // empty
71      }
72      HAL_GPIO_TogglePin(SIGNAL_GPIO_Port, SIGNAL_Pin);
73  }
74  /* USER CODE END 0 */
```

- Set cho timer đếm bắt đầu từ 0

```
62  __HAL_TIM_SET_COUNTER(&htim1,0);
63  __HAL_TIM_SET_COUNTER(&htim2,0);
```

- Chờ timer1 đếm đủ thời gian t (hay số ticks). Giá trị t phụ thuộc vào biến Angle theo công thức dưới

```
64  while(__HAL_TIM_GET_COUNTER(&htim1) < 6000+(uint16_t)(200*Angle/9)) {
65      // empty
66  }
```

- Khi timer1 đếm đủ thì đảo chân SIGNAL

```
67  HAL_GPIO_TogglePin(SIGNAL_GPIO_Port, SIGNAL_Pin);
```

- Tương tự khi timer2 đếm đủ 20ms thì đảo chân SIGNAL

```
69  while(__HAL_TIM_GET_COUNTER(&htim2) < 40000) {
70      // empty
71  }
72  HAL_GPIO_TogglePin(SIGNAL_GPIO_Port, SIGNAL_Pin);
73  }
```

- Trong hàm main khởi tạo timer1 và 2. Đưa chân SIGNAL lên 1

```
106  /* USER CODE BEGIN 2 */  
107  HAL_TIM_Base_Start(&htim1);  
108  HAL_TIM_Base_Start(&htim2);  
109  HAL_GPIO_WritePin(SIGNAL_GPIO_Port,SIGNAL_Pin,1);  
110  /* USER CODE END 2 */
```

- Gọi hàm RCservo trong while chính. Nhập góc quay mong muốn

```
113  /* USER CODE BEGIN WHILE */  
114  while (1)  
115  {  
116      /* USER CODE END WHILE */  
117      RCservo(-80);  
118      /* USER CODE BEGIN 3 */  
119  }
```

#### 4. Ghi chú khác (nếu có)