PWM and GPIO

Week 3 Monday

Liu Yuchen

Leo

- Leader of the fighter robots, including infantry, hero and sentry
- Year2, CS
- Come from Harbin, China

Objective

1. How to configure GPIO pins?
2. How to implement PWM on STM32?
3. How to implement softtimer?

Preparation----Configure Keil MDK-ARM 5

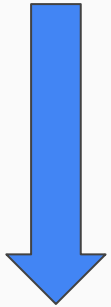
- reply to the post sent last night with your CID
- set the output simulator type

How to configure GPIO pins?

GPIO refers to **General-purpose input/output**

INPUT patterns

- pull-down → by default 0
- pull-up → by default 1
- analog
- floating

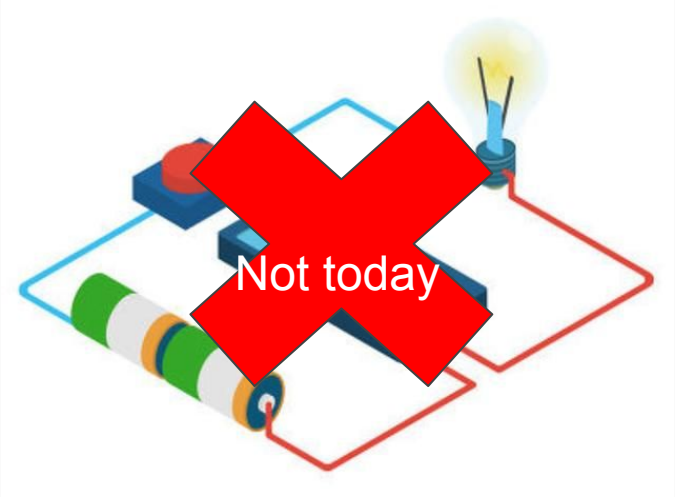


OUTPUT patterns

- push-pull
- Open drain

How to configure GPIO pins?

The most simple but tedious case practice: use a switch to control an LED



Use pin PD1(pull down) to detect the input signal
Then
Use pin PI3(push-pull) to control the LED on/off

If input = 0;
output = low;
If input = 1;
output = high;

How to configure GPIO pins?

https://www.st.com/zh/development-tools/stm32cubemx.html

type the URL or search for “cubemx”

How to translate your ideas into code?

Sample code:

```
174  static void MX_GPIO_Init(void)
175  {
176
177    GPIO_InitTypeDef GPIO_InitStruct;
178
179    /* GPIO Ports Clock Enable */
180    __HAL_RCC_GPIOD_CLK_ENABLE();
181    __HAL_RCC_GPIOI_CLK_ENABLE();
182
183    /*Configure GPIO pin Output Level */
184    HAL_GPIO_WritePin(LED_pin_GPIO_Port, LED_pin_Pin, GPIO_PIN_RESET);
185
186    /*Configure GPIO pin : switch_pin_Pin */
187    GPIO_InitStruct.Pin = switch_pin_Pin;
188    GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
189    GPIO_InitStruct.Pull = GPIO_PULLDOWN;
190    HAL_GPIO_Init(switch_pin_GPIO_Port, &GPIO_InitStruct);
191
192    /*Configure GPIO pin : LED_pin_Pin */
193    GPIO_InitStruct.Pin = LED_pin_Pin;
194    GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
195    GPIO_InitStruct.Pull = GPIO_NOPULL;
196    GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
197    HAL_GPIO_Init(LED_pin_GPIO_Port, &GPIO_InitStruct);
198
199  }
200
```

What if you need to configure GPIO on a existing project?

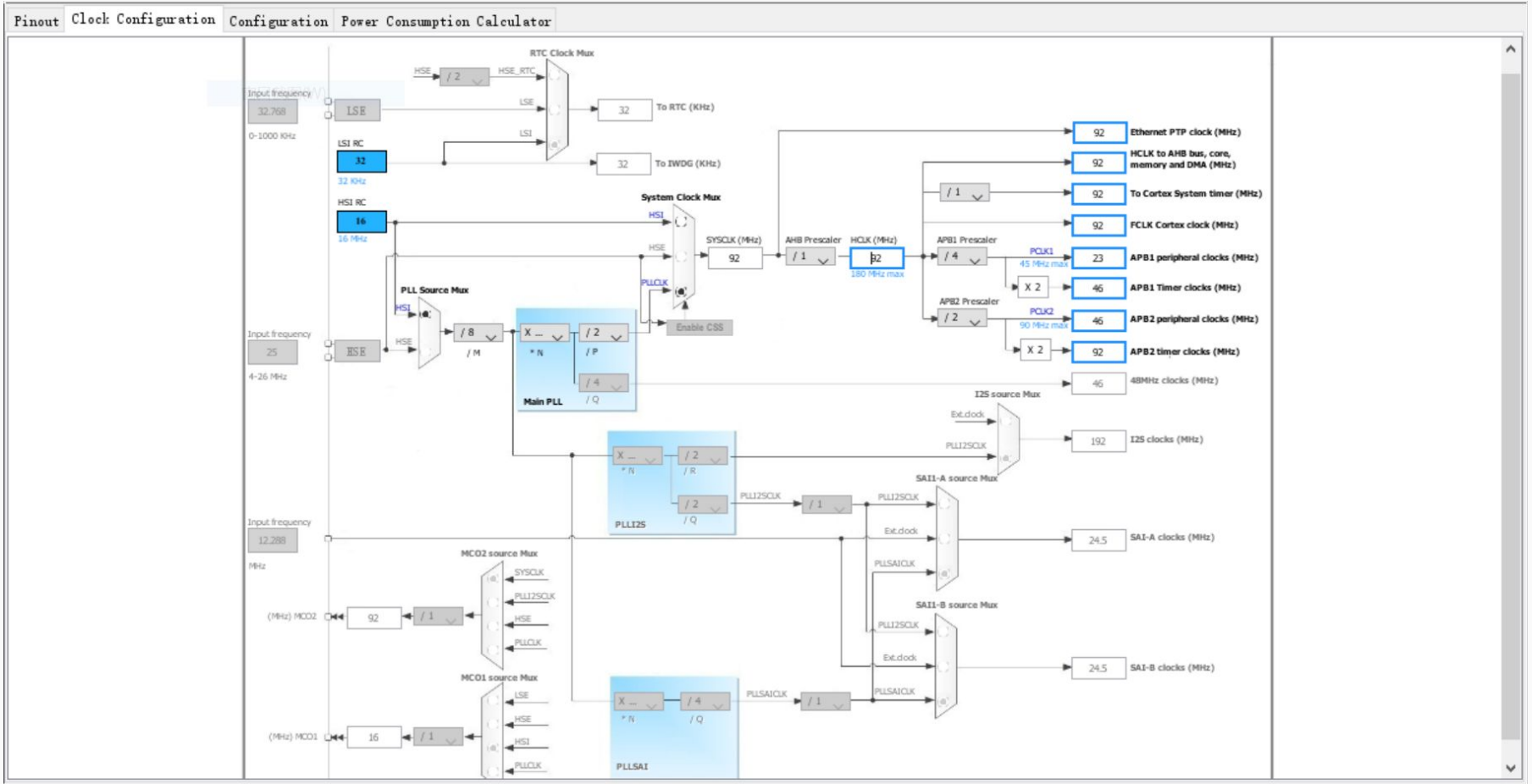
-Copy and paste! (Be careful not to miss any part!)

What is timer?

microboard

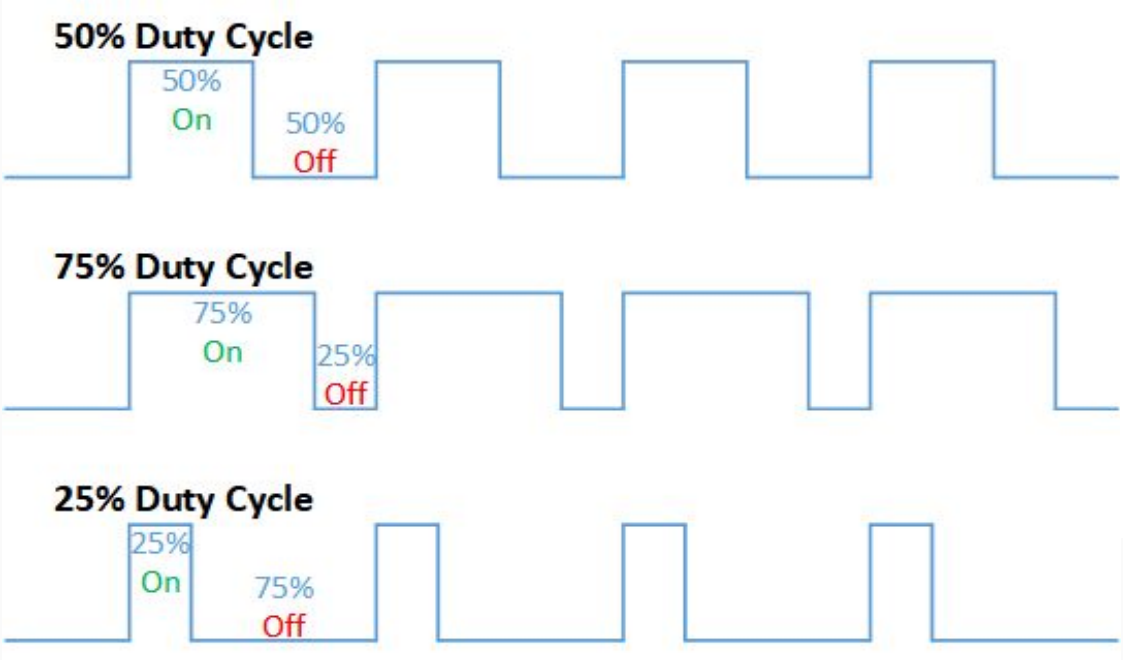
- modulate the speed of communication
- control the process of memory reading and writing in RAM(random access memory)
- help generate PWM signal
- ……

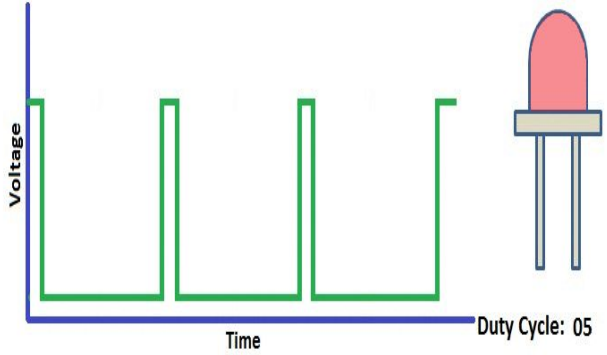
The math in clock tree

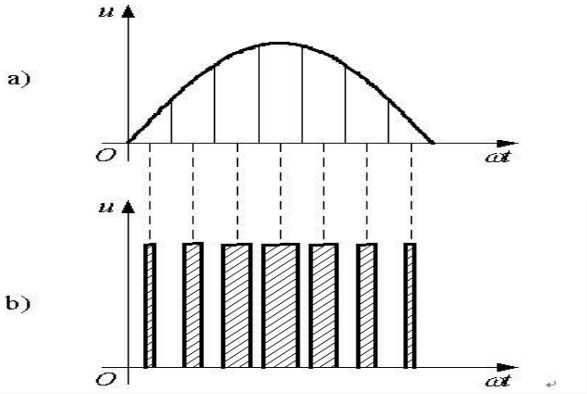


What is PWM?

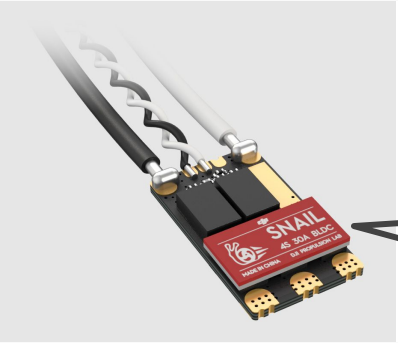
PWM refers to Pulse-width Modulation







The calculation of period (T) and duty cycle for PWM



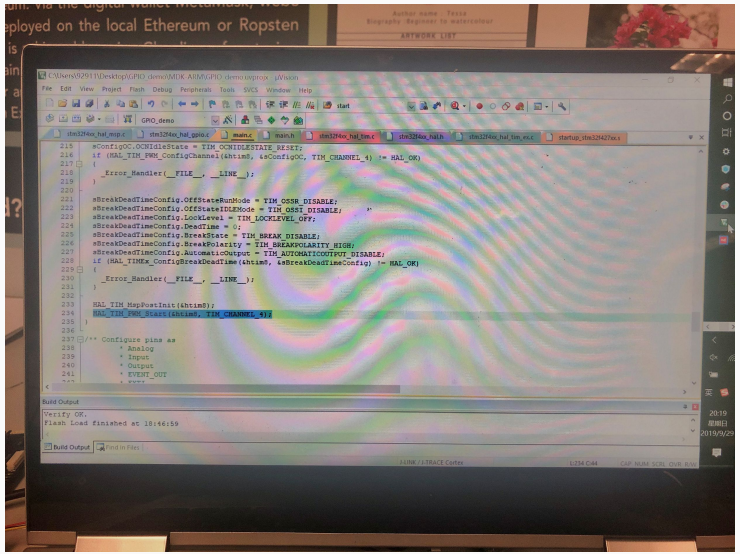
robomaster 2019-3-19 18:45:06

板凳

snail电机是开机的时候 PWM快速从 1ms到2ms 然后进行校准

electronic speed controllor

使用道具 举报 回复 支持 反对



Experiments has found that a light blinks with a period under 40ms is constantly shining to human’s eyes.

How to configure PWM on STM32?

Option one:
Use cubeMX

Option two:
Refer to last year's code **tim.c**

Sample code:

```
htim8.Instance = TIM8;
htim8.Init.Prescaler = 1200-1;
htim8.Init.CounterMode = TIM_COUNTERMODE_UP;
htim8.Init.Period = 10000-1;
htim8.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
htim8.Init.RepetitionCounter = 0;
```

```
HAL_TIM_PWM_Start(&htim8, TIM_CHANNEL_4);
```

Change pulse of the PWM

- Pulse determines the duration of high voltage
- The ratio of pulse and period determines the duty cycle

```
TIMx->CCRx = pulse;
```

or

```
__HAL_TIM_SET_COMPARE(&htimx, TIM_CHANNEL_x, pulse);
```

How to implement softtimer?

- Hard Ware Timer: Count clicks => Signals
- Interrupt
- Softwane Timer: Similar
- Usual thread: Hault (similarity)
- Callback function -> Handler (Difference)

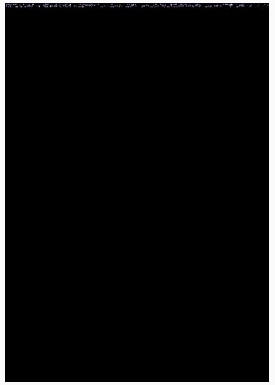
Step One: Callback Registration
Step Two

How it works?

- The code & Graph

Homework

Make a breathing light



Requirements:

Output 0v when closed, 5v when open!!!!!

1. use a switch to control the on/off of the light

   suggestion: let PI0 be the input pin and PI2 be the output pin

2. make the light to breath periodically

3. make the period to be 2s (1s to brighten and 1s to darken)