



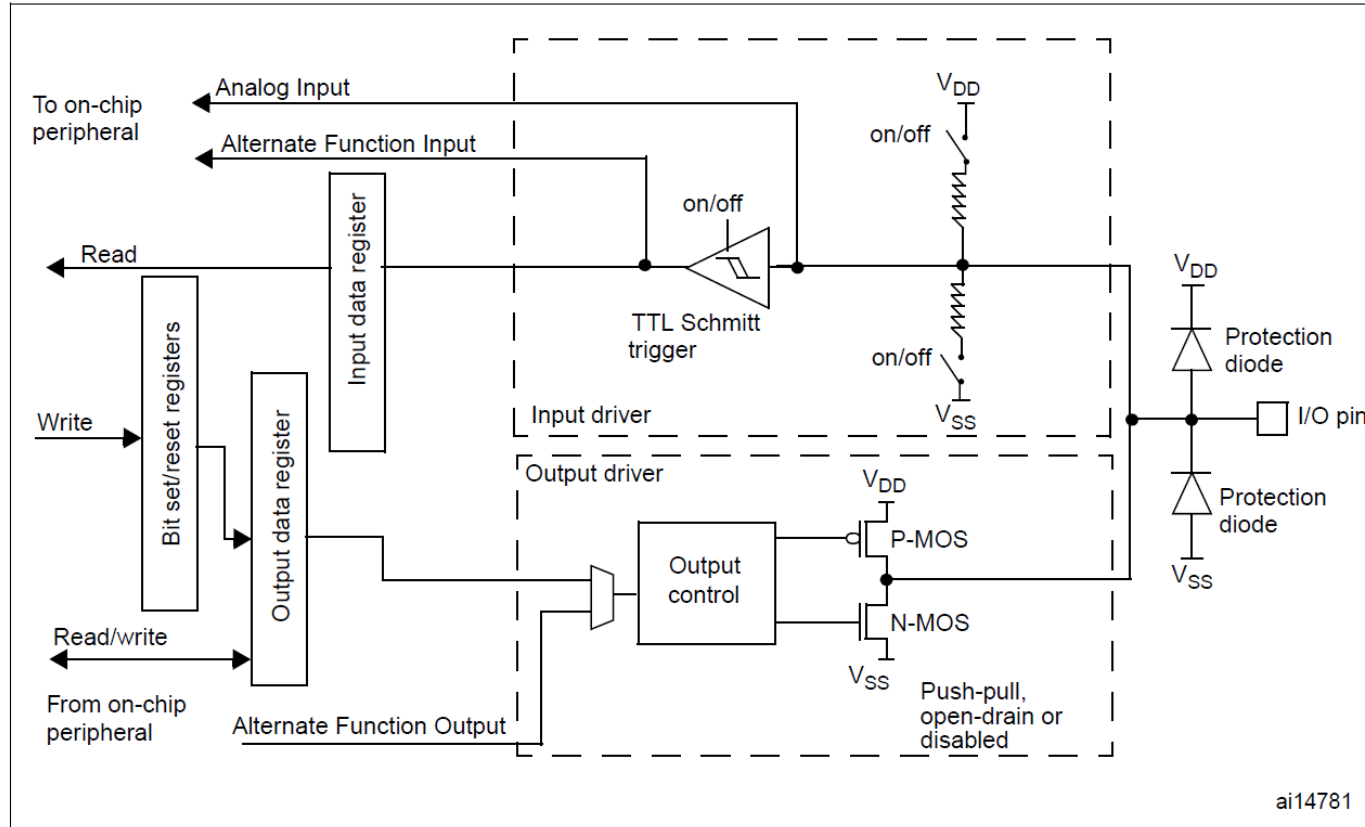
IUST

Iran University of
Science and Technology

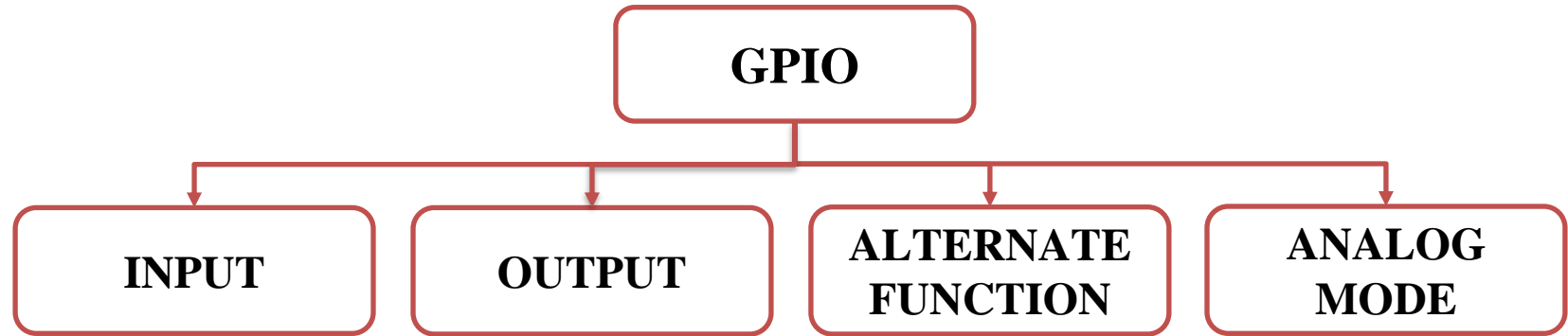
Microcontroller

Department of Electrical Engineering
Iran University of Science and Technology

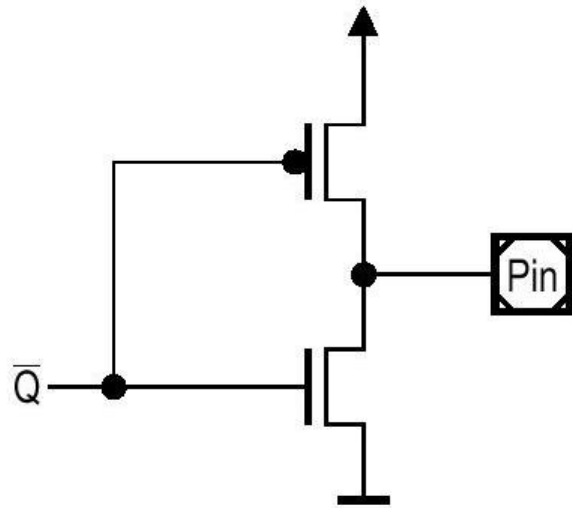
GPIO MAP



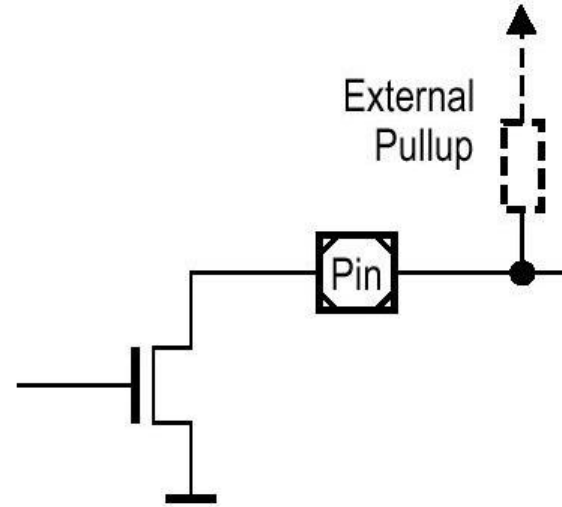
GPIO MODE



PUSH PULL / OPEN DRAIN

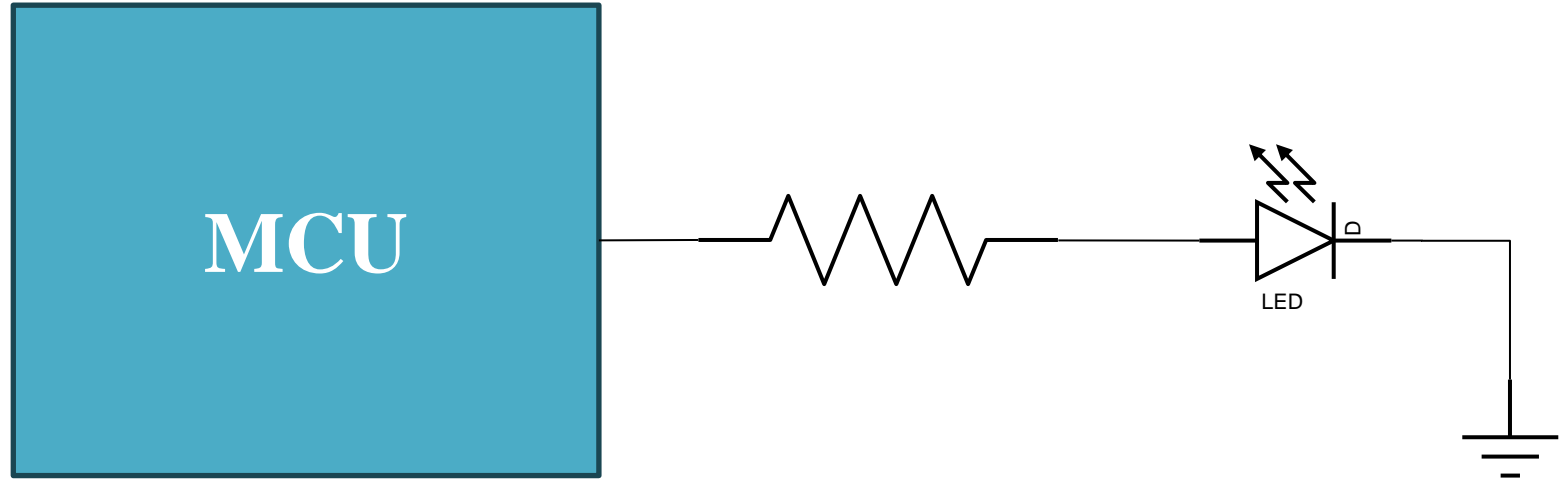


Push/Pull Output Driver



Open-Drain Output Driver

LED



RCC_APB2ENR :IOPCEN

RCC->APB2ENR |= RCC_APB2ENR_IOPCEN

GPIO_x_CRH :MODE_y[1:0]

GPIO_x_CRH :MODE_y[1:0]

```
RCC->APB2ENR |= RCC_APB2ENR_IOPCEN;
```

```
GPIOC->CRH |= GPIO_CRH_MODE9;
```

GPIOn_CRH :CNFy[1:0]

GPIO_x_CRH :CNF_y[1:0]

```
RCC->APB2ENR |= RCC_APB2ENR_IOPCEN;
```

```
GPIOC->CRH |= GPIO_CRH_MODE9;
```

```
GPIOC->CRH &= ~GPIO_CRH_CNF9;
```

[illegible]

GPIOx_ODR

```
GPIOC->ODR |= GPIO_ODR_ODR13;
```

```
GPIOC->ODR &= ~ GPIO_ODR_ODR13;
```

[illegible]

GPIOx_BSRR

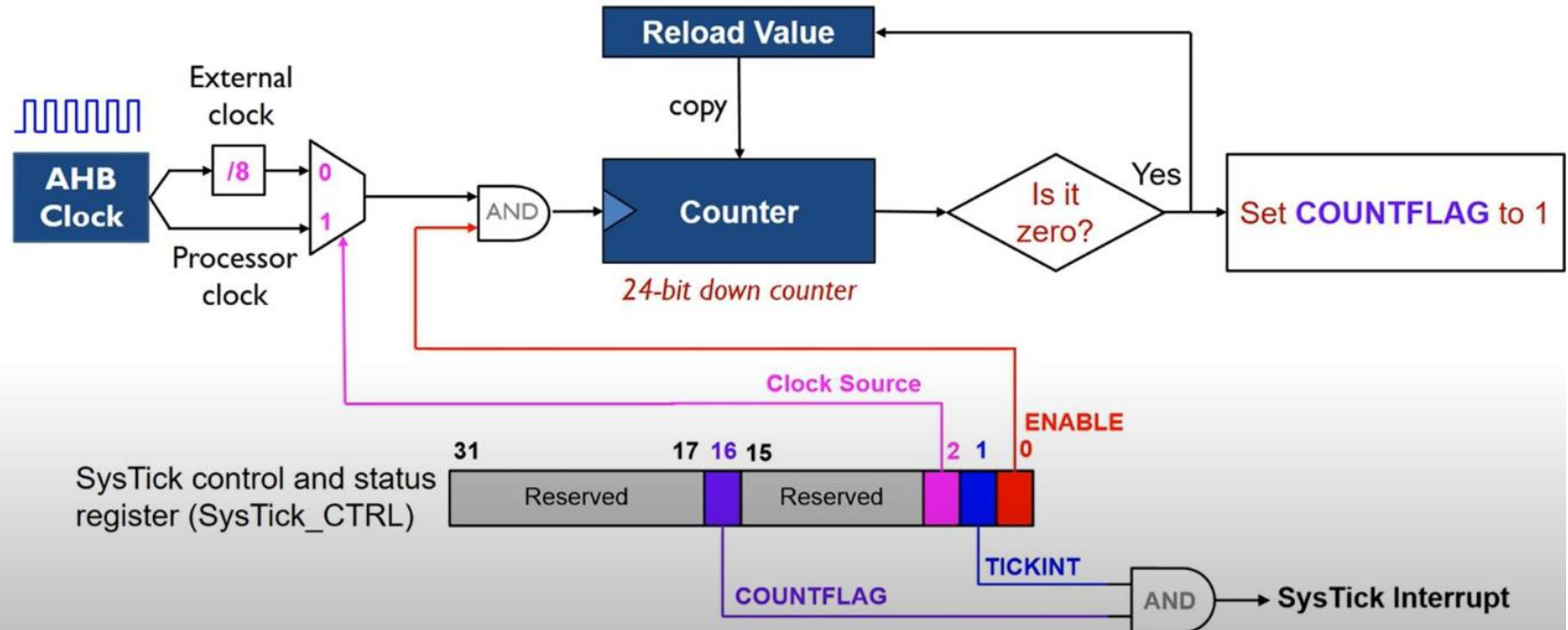
```
GPIOC->BSRR = GPIO_BSRR_BS13;
```

```
GPIOC->BSRR = GPIO_BSRR_BR13;
```

SYSTICK

The processor has a 24-bit system timer, SysTick, that counts down from the reload value to zero, reloads (wraps to) the value in the LOAD register on the next clock edge, then counts down on subsequent clocks.

SYSTICK



STK_LOAD

```
uint32_t ticks = SystemCoreClock / 1000;  
SysTick->LOAD = ticks - 1;
```

[illegible]

STK_LOAD

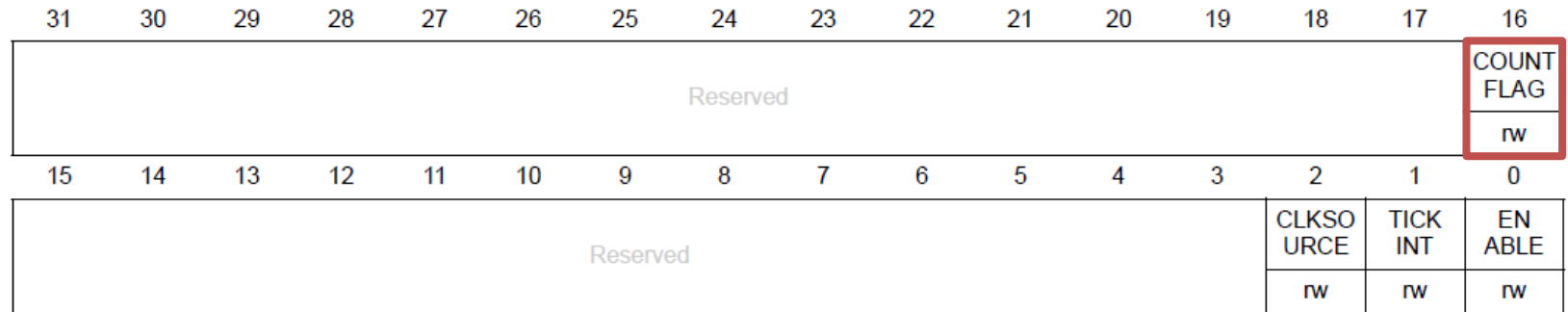
```
uint32_t ticks = SystemCoreClock / 1000;
```

```
SysTick->LOAD = ticks - 1;
```

```
SysTick->VAL = 0;
```

STK_CTRL

COUNTFLAG:
Returns 1 if timer
counted to 0 since last
time this was read



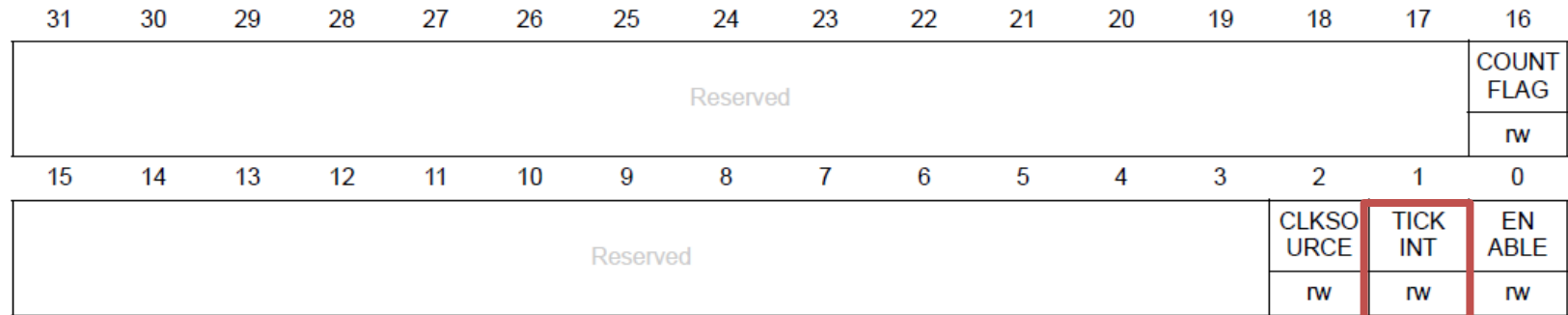
STK_CTRL

CLKSOURCE: Clock source selection
Selects the clock source.
0: AHB/8
1: Processor clock (AHB)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															COUNT FLAG
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CLKSO URCE	TICK INT	EN ABLE	
												rw	rw	rw	

STK_CTRL

TICKINT: SysTick exception request enable
 0: Counting down to zero does not assert the SysTick exception request
 1: Counting down to zero to asserts the SysTick exception request.



STK_CTRL

ENABLE: Counter enable

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															COUNT FLAG
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CLKSO URCE	TICK INT	EN ABLE	
												rw	rw	rw	

STK_LOAD

```
uint32_t ticks = SystemCoreClock / 1000;
```

```
SysTick->LOAD = ticks - 1;
```

```
SysTick->VAL = 0;
```

```
SysTick->CTRL |= SysTick_CTRL_CLKSOURCE |  
                  SysTick_CTRL_TICKINT |  
                  SysTick_CTRL_ENABLE;
```

Delay function

```
volatile uint32_t msTicks = 0;  
void SysTick_Handler(void) {  
    if (msTicks > 0) {  
        msTicks--;  
    }  
}  
void delay_ms(uint32_t ms) {  
    msTicks = ms;  
    while (msTicks > 0) { }  
}
```