

Interrupciones

Ing. Juan Sebastian Correa

Diciembre 2021

- 1 Concepto ¿Qué son?
- 2 Polling vs Interrupciones
- 3 ¿Como funciona la interrupcion?
- 4 Fuentes de Interrupcion
- 5 Elementos de una interrupcion
- 6 Reset y Vector Interrupcion
- 7 metodo ISR

Interrupciones - ¿Qué son?

Son peticiones realizadas a la unidad central de proceso por eventos de hardware. Estas a su vez están asociadas con funciones (desde la programación) que no reciben ni regresan parámetros.

En términos generales una interrupción se encarga de interrumpir la unidad central de proceso en su ejecución para atender cierta localidad de memoria.

Tecnica de polling: La CPU del microcontrolador consulta constantemente sobre las acciones fisicas o logicas del programa atendiendo un evento a la vez. Esta tecnica demanda mucho de la unidad central de proceso y genera un consumo elevado en el sistema MCU. Para mitigar el efecto de este fenomeno se debe colocar un tiempo de muestreo idoneo (calculado) para reducir el consumo y no perder eventos mientras la CPU duerme.

Interrupciones: La CPU del microcontrolador no necesita consultar constantemente el evento generado por el hardware, ya que posee un sistema dedicado para indicar que ha ocurrido una señal para ser atendida.

Funcionamiento de la interrupcion

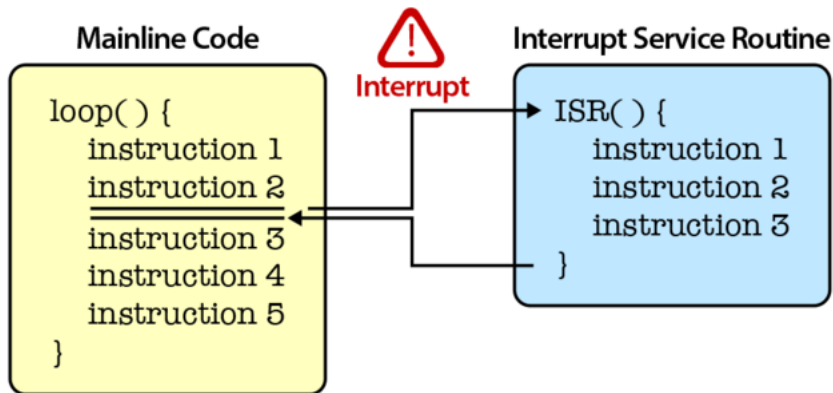


Figure: Funcionamiento global de una interrupcion

Funcionamiento de la interrupcion

```
while (1)
{
    while(PORTD <128){
        PORTD |= 1<<i;
        i++;
        _delay_ms(1000);
    }
    if(temperatura > 125.0){
        apagar_calentador();
    }
}

ISR(Fuente_Interrupcion){
    // ACCIONES A EJECUTAR EN LA INTERRUPCION
}
```

Figure: Codigo de muestra

Fuentes de interrupcion

La familia AVR cuenta con diferentes fuentes de interrupcion, sin embargo estas se pueden categorizar en:

- ➊ **Interrupciones externas(External Interrupts):** Estan asociadas con los pines INT_n , n varia de 0 a 7.
- ➋ **Interrupciones por cambio de estado(Change Interrupts):** Estan asociados con los pines $PCINT_m$, donde m varia de 0 a 3.
- ➌ **Interrupciones internas de perifericos (Internal Peripheral Interrupts):** Asociado a los perifericos del MCU como lo son, Timer, ADC, UART, SPI, I2C, etc.

Elementos de una interrupcion

Cada bloque de interrupcion tiene sus registros de control, de manera generalizada podemos decir que sin importar el tipo de interrupcion a utilizar, la operacionr recae en los siguientes dos conceptos:

- **Flag bit:** Encargado de informar a la CPU que un evento ha ocurrido
- **Enable bit:** Encargado de habilitar la interrupcion a utilizar.

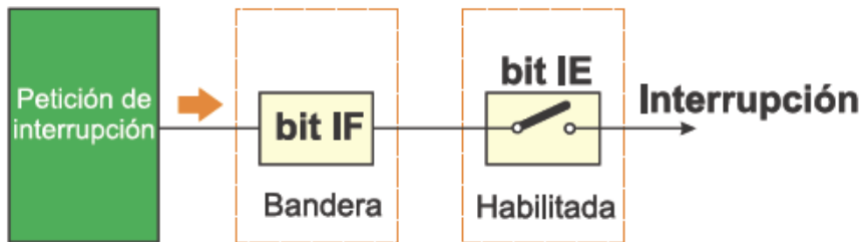


Figure: Enable and Flag bit

Reset y Vector Interrupcion

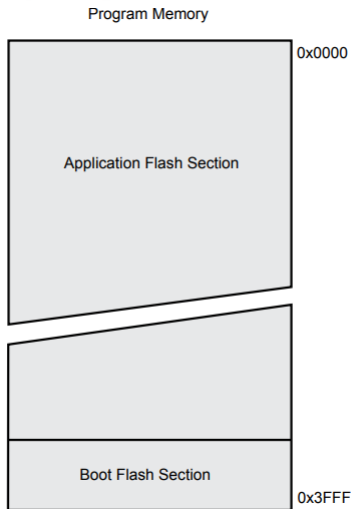


Figure: Memoria del MCU AVR

Reset y Vector Interrupcion

Vector No	Program Address	Source	Interrupts definition
1	0x0000	RESET	External Pin, Power-on Reset, Brown-out Reset and Watchdog System Reset
2	0x0002	INT0	External Interrupt Request 0
3	0x0004	INT1	External Interrupt Request 0
4	0x0006	PCINT0	Pin Change Interrupt Request 0
5	0x0008	PCINT1	Pin Change Interrupt Request 1
6	0x000A	PCINT2	Pin Change Interrupt Request 2
7	0x000C	WDT	Watchdog Time-out Interrupt
8	0x000E	TIMER2_COMPA	Timer/Counter2 Compare Match A
9	0x0010	TIMER2_COMPB	Timer/Counter2 Compare Match B
10	0x0012	TIMER2_OVF	Timer/Counter2 Overflow
11	0x0014	TIMER1_CAPT	Timer/Counter1 Capture Event
12	0x0016	TIMER1_COMPA	Timer/Counter1 Compare Match A
13	0x0018	TIMER1_COMPB	Timer/Counter1 Compare Match B
14	0x001A	TIMER1_OVF	Timer/Counter1 Overflow
15	0x001C	TIMER0_COMPA	Timer/Counter0 Compare Match A
16	0x001E	TIMER0_COMPB	Timer/Counter0 Compare Match B
17	0x0020	TIMER0_OVF	Timer/Counter0 Overflow
18	0x0022	SPI0_STC	SPI1 Serial Transfer Complete
19	0x0024	USART0_RX	USART0 Rx Complete
20	0x0026	USART0_UDRE	USART0, Data Register Empty
21	0x0028	USART0_TX	USART0, Tx Complete
22	0x002A	ADC	ADC Conversion Complete

Reset y Vector Interrupcion

Vector No	Program Address	Source	Interrupts definition
23	0x002C	EE READY	EEPROM Ready
24	0x002E	ANALOG COMP	Analog Comparator
25	0x0030	TWI	2-wire Serial Interface (I ² C)
26	0x0032	SPM READY	Store Program Memory Ready
27	0x0034	USART0_START	USART0 Start frame detection
28	0x0036	PCINT3	Pin Change Interrupt Request 3
29	0x0038	USART1_RX	USART0 Rx Complete
30	0x003A	USART1_UDRE	USART0, Data Register Empty
31	0x003C	USART1_TX	USART0, Tx Complete
32	0x003E	USART1_START	USART1 Start frame detection
33	0x0040	TIMER3_CAPT	Timer/Counter3 Capture Event
34	0x0042	TIMER3_COMPA	Timer/Counter3 Compare Match A
35	0x0044	TIMER3_COMPB	Timer/Counter3 Compare Match B
36	0x0046	TIMER3_OVF	Timer/Counter3 Overflow
37	0x0048	CFD	Clock failure detection interrupt
38	0x004A	PTC_EOC	PTC End of Conversion
39	0x004C	PTC_WCOMP	PTC Window comparator mode
40	0x004E	SPI1_STC	SPI1 Serial Transfer Complete
41	0x0050	TWI1	TWI1 Transfer complete
42	0x0052	TIMER4_CAPT	Timer/Counter3 Capture Event
43	0x0054	TIMER4_COMPA	Timer/Counter3 Compare Match A
44	0x0056	TIMER4_COMPB	Timer/Counter3 Compare Match B
45	0x0058	TIMER4_OVF	Timer/Counter3 Overflow

Metodo para utilizar una interrupcion

```
ISR(INT0_vect){  
  
    PORTB |= (1<<PINB5);  
    PORTB &= ~(1<<PINB4);  
  
}
```