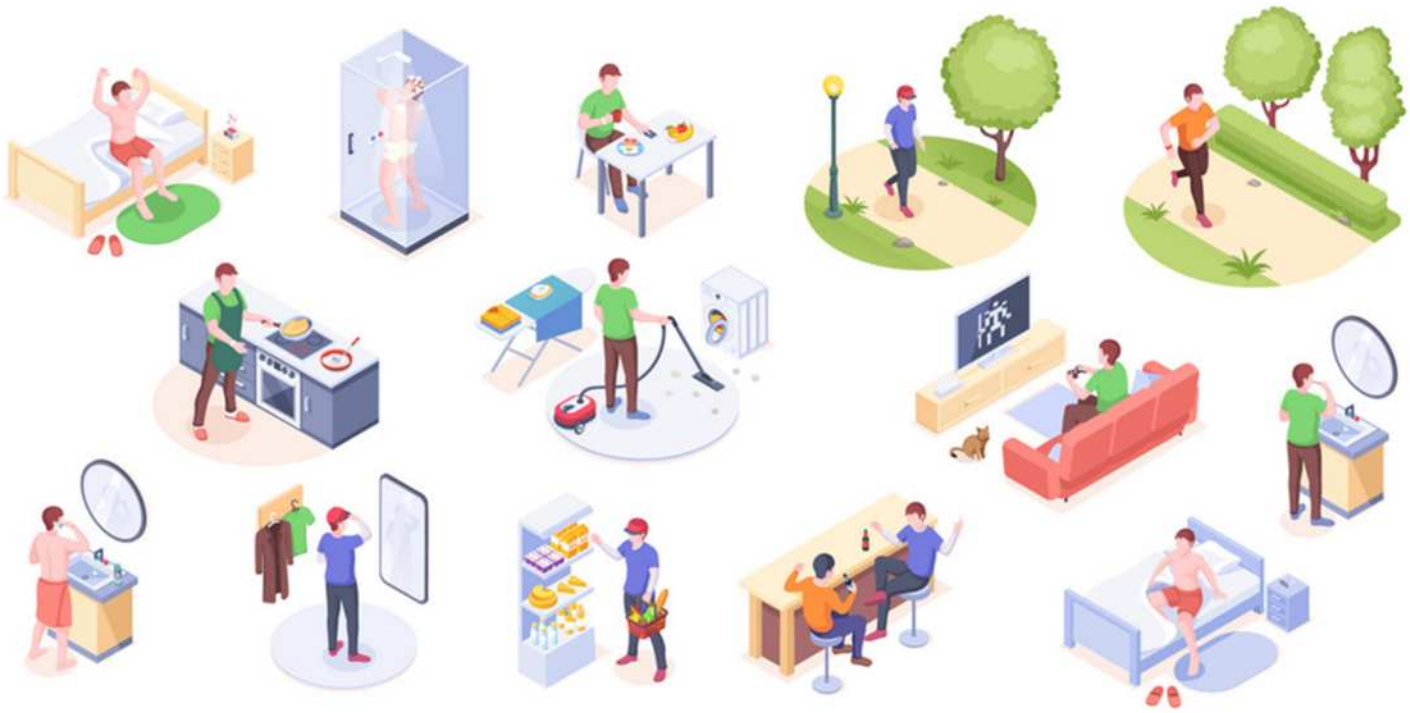
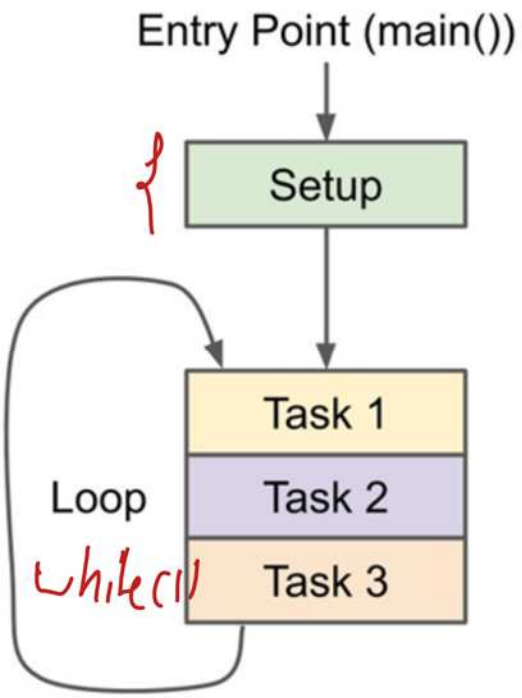

CPU
Rod


10 m/h

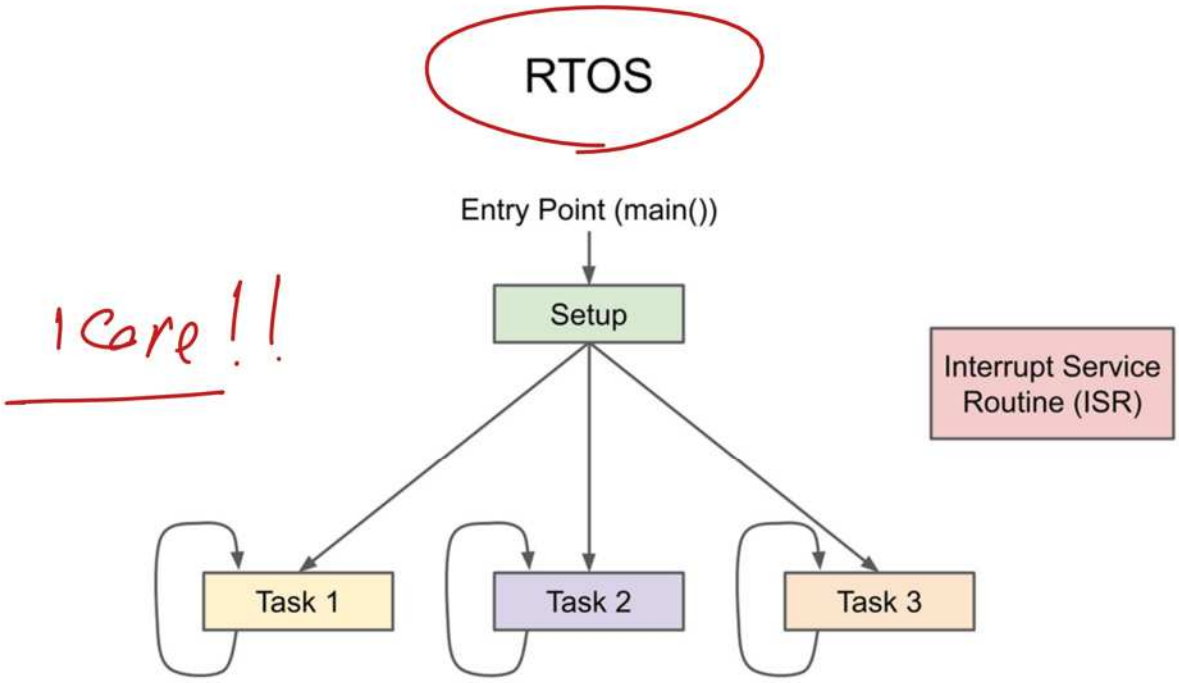


Super Loop



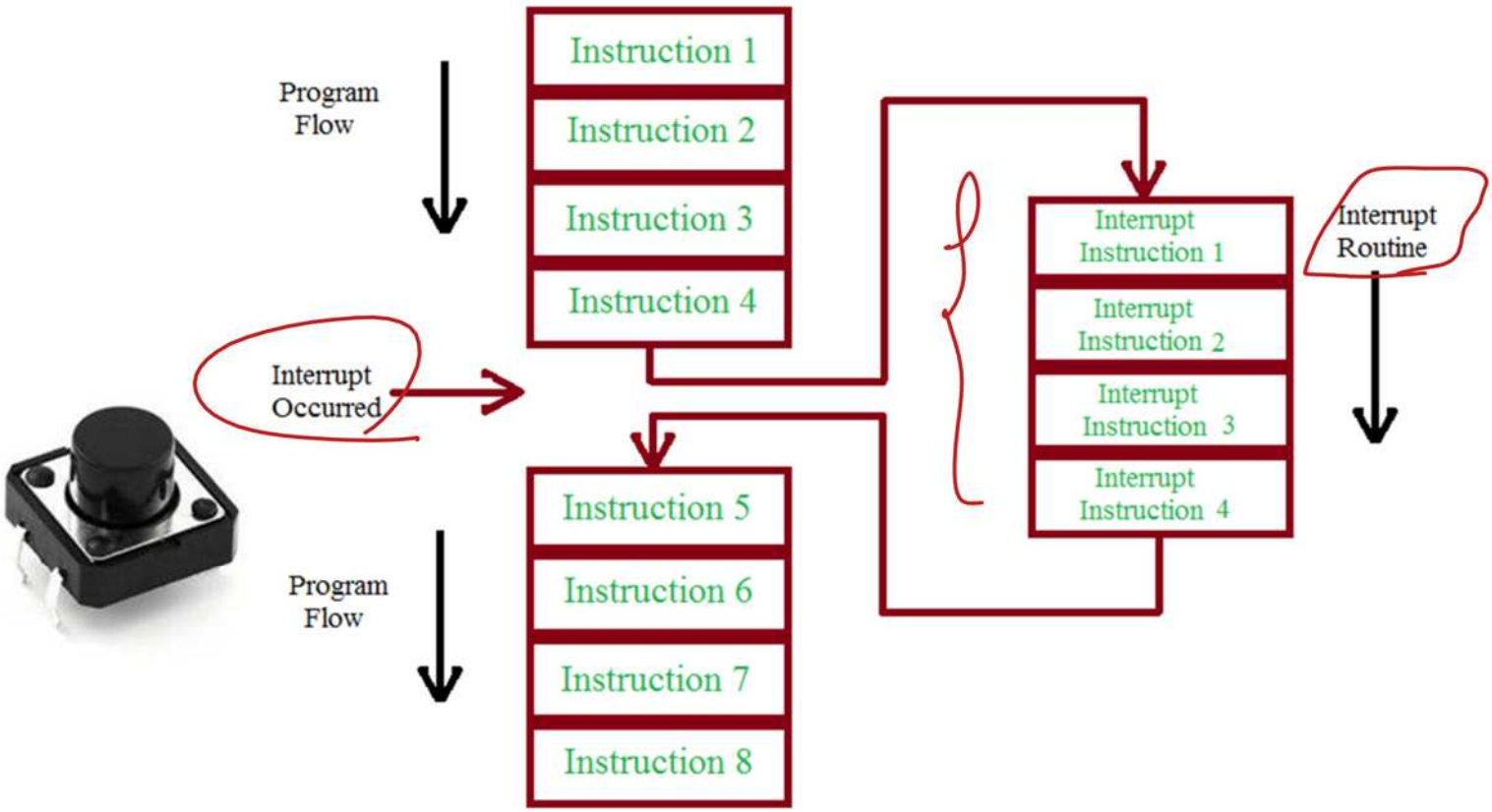

```
int main(void)
{
    setBit(DDRB, 0);
    clearBit(DDRD, 0);
    clearBit(DDRD, 4);
    setBit(PORTD, 4);
    clearBit(MCUCR, PUD);
    while(1)
    {
        if(checkBit(PIND, 0))
        {
            while(checkBit(PIND, 0));
            setBit(PORTB, 0);
        };

        if(!checkBit(PIND, 4))
        {
            while(!checkBit(PIND, 4));
            clearBit(PORTB, 0);
        };
    };
};
```

Super Loop



I care!!


Program Flow
Instruction 1
Instruction 2
Instruction 3
Instruction 4
Instruction 5
Instruction 6
Instruction 7
Instruction 8
Program Flow

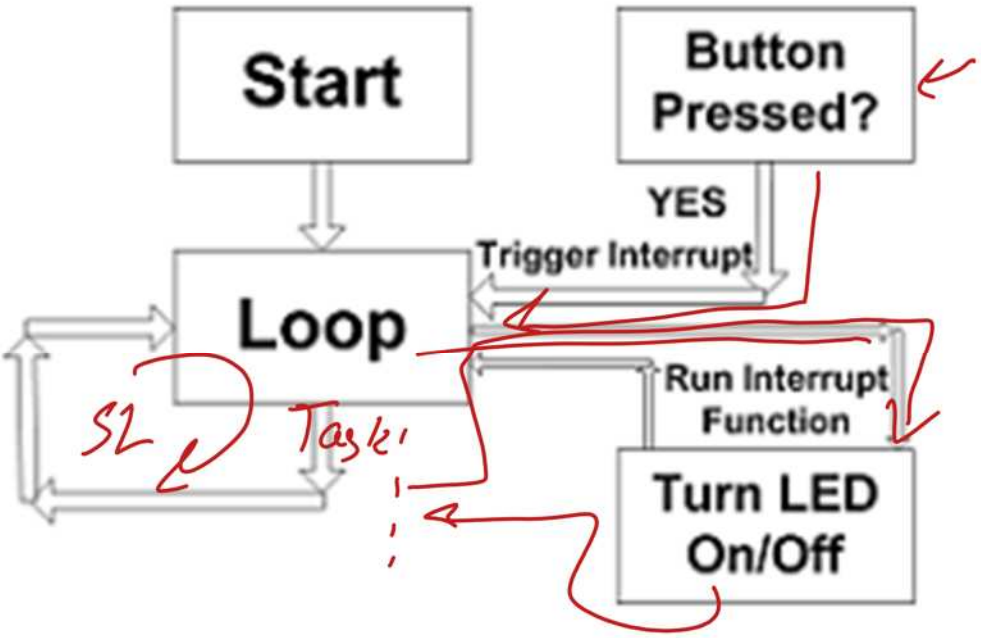

Program Flow
Instruction 1
Instruction 2
Instruction 3
Instruction 4
Interrupt Occurred
Interrupt Instruction 1
Interrupt Instruction 2
Interrupt Instruction 3
Interrupt Instruction 4
Interrupt Routine
Instruction 5
Instruction 6
Instruction 7
Instruction 8
Program Flow

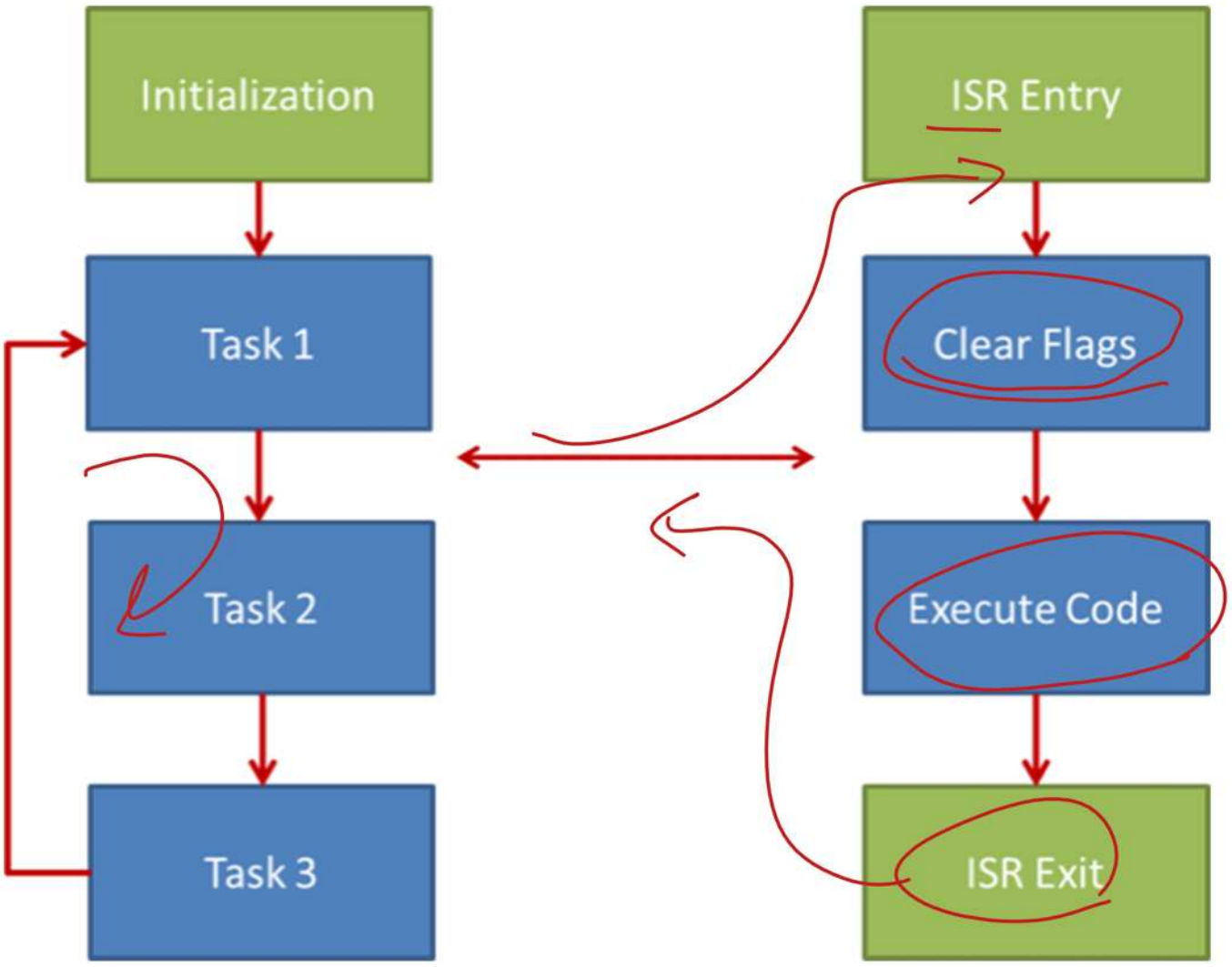
```
int main(void)
{
    setBit(DDRB, 0);
    clearBit(DDRD, 0);
    clearBit(DDRD, 4);
    setBit(PORTD, 4);
    clearBit(MCUCR, PUD);
    while(1)
    {
        if(checkBit(PIND, 0))
        {
            while(checkBit(PIND, 0));
            setBit(PORTB, 0);
        };

        if(!checkBit(PIND, 4))
        {
            while(!checkBit(PIND, 4));
            clearBit(PORTB, 0);
        };
    };
};
```

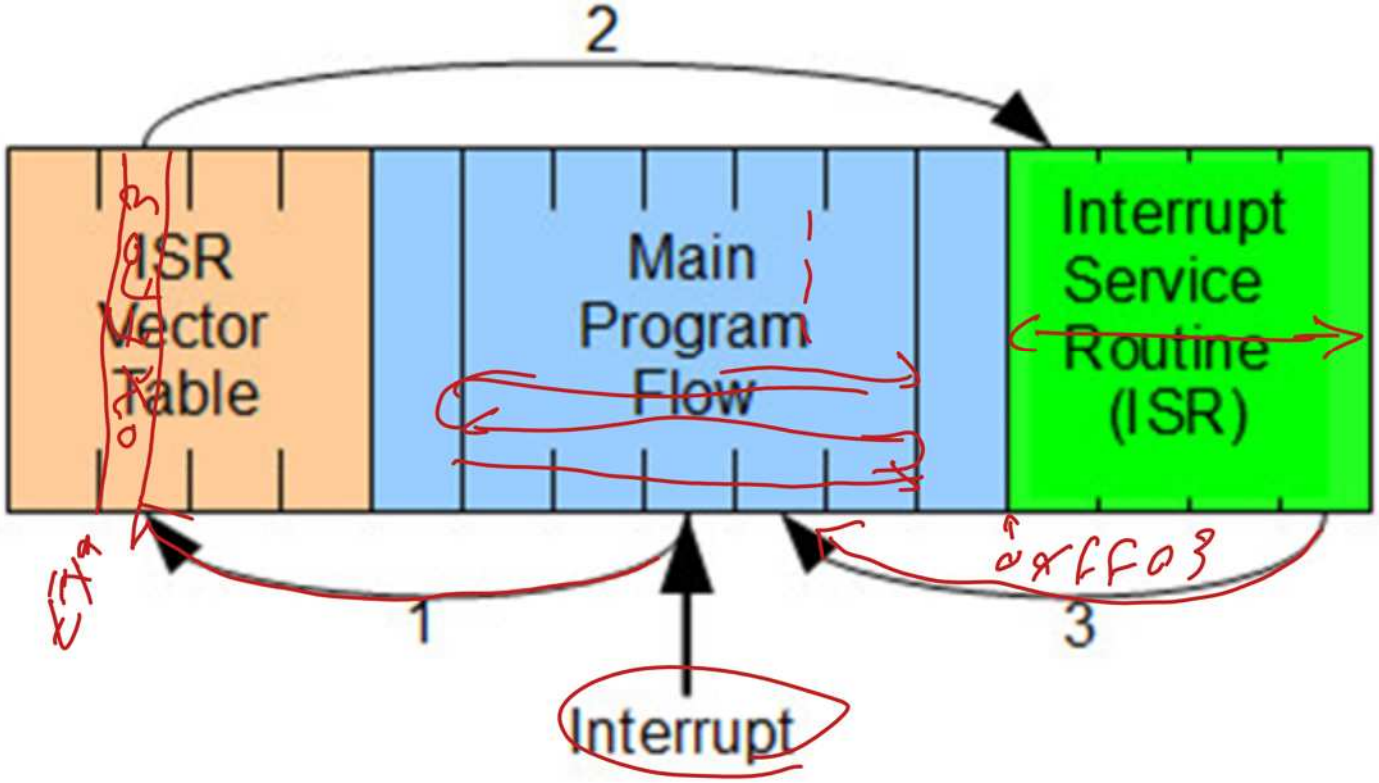
Microcontroller Interrupts

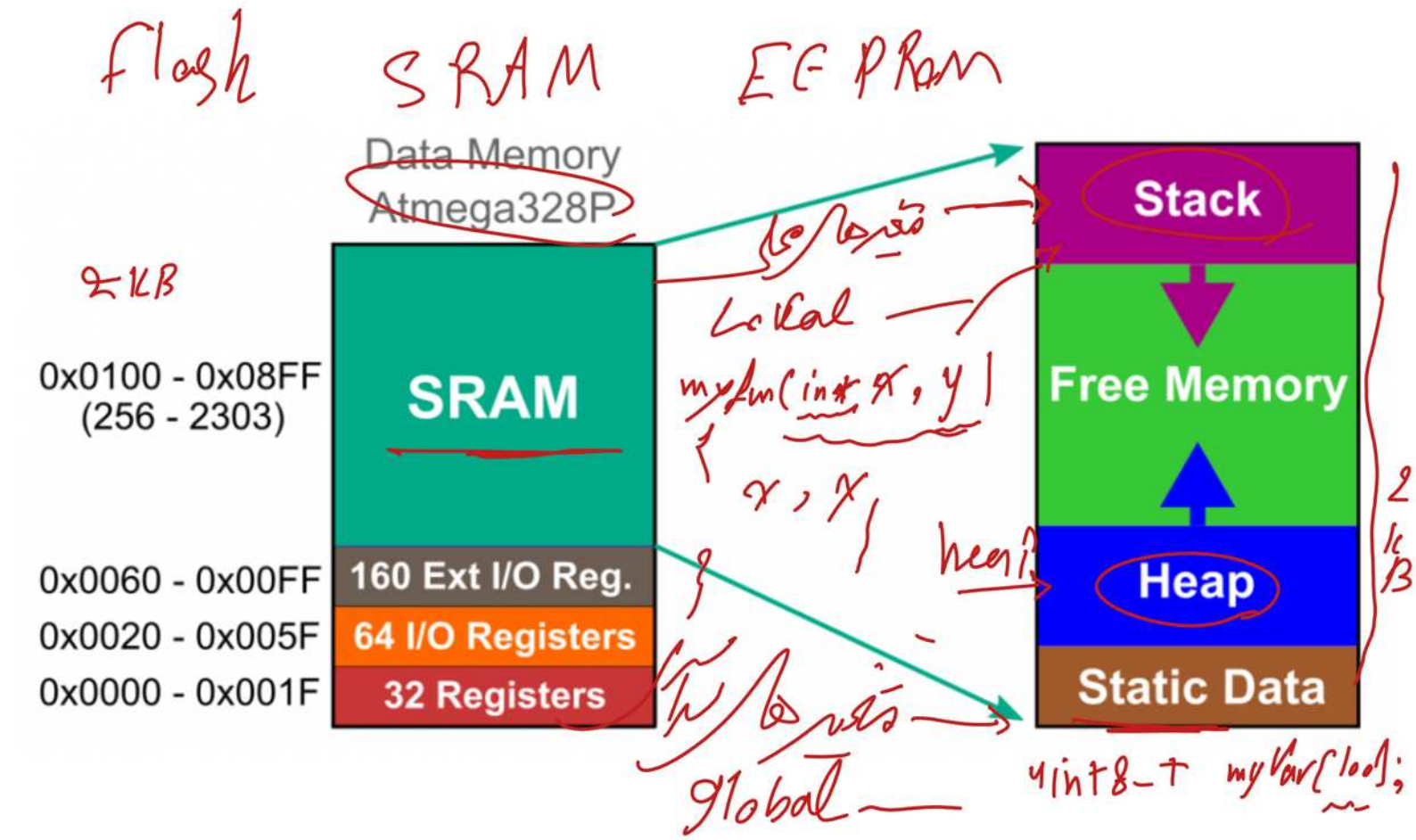
```
int main(void){
    while(1){
        // Stop interrupting me
    }
}

ISR(ADC_vect){
    // I like to interrupt
}

ISR(TIMER1_vect){
    // Me too
}
```

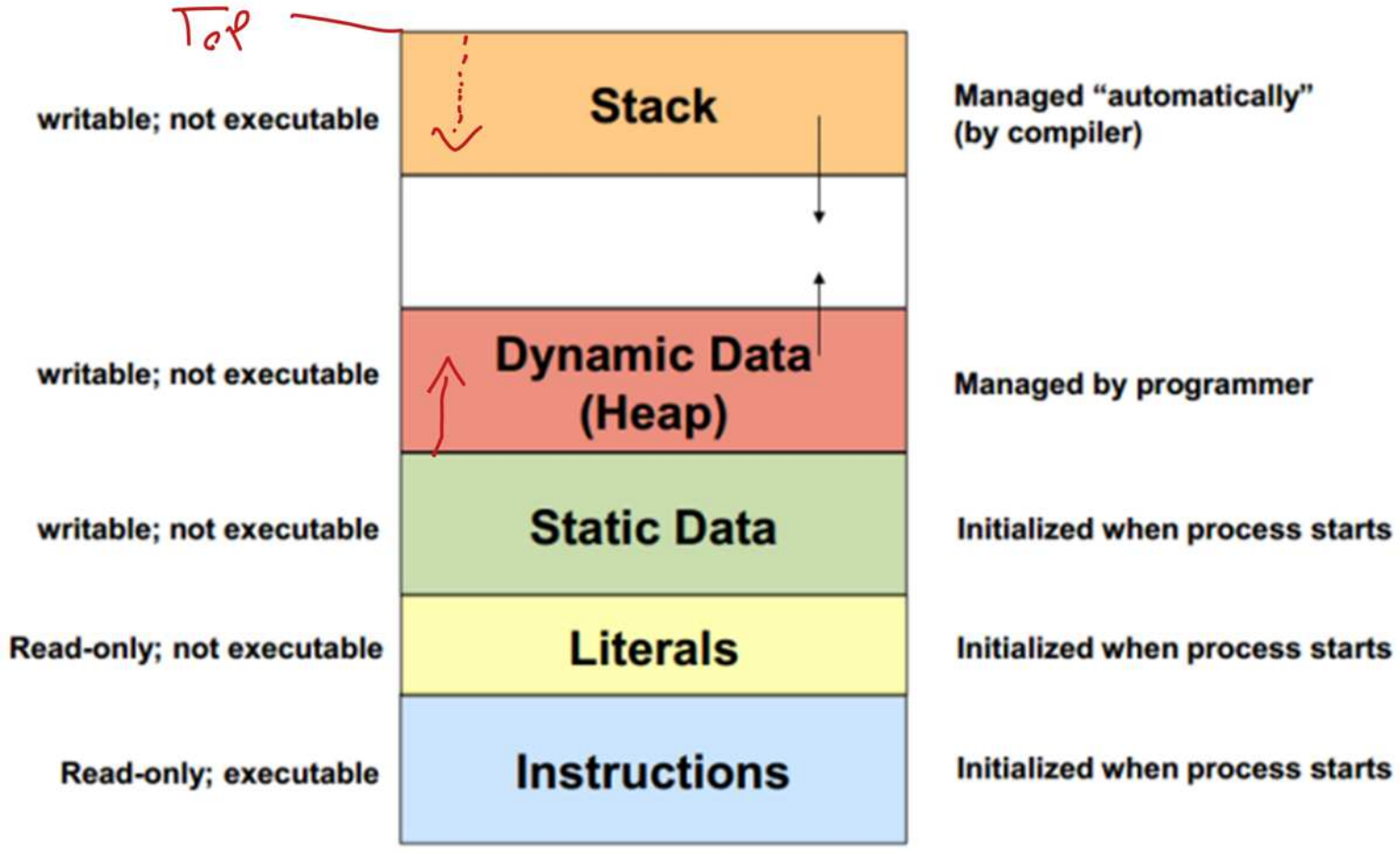
```
ISR(EXT_INT0)
{
    setBit(PORTB, 0);
};

ISR(EXT_INT1)
{
    clearBit(PORTB, 0);
};

int main(void)
{
    setBit(DDRB, 0);
    clearBit(DDRD, 0);
    clearBit(DDRD, 4);
    setBit(PORTD, 4);
    clearBit(MCUCR, PUD);
    while(1)
    {
    };
};
```

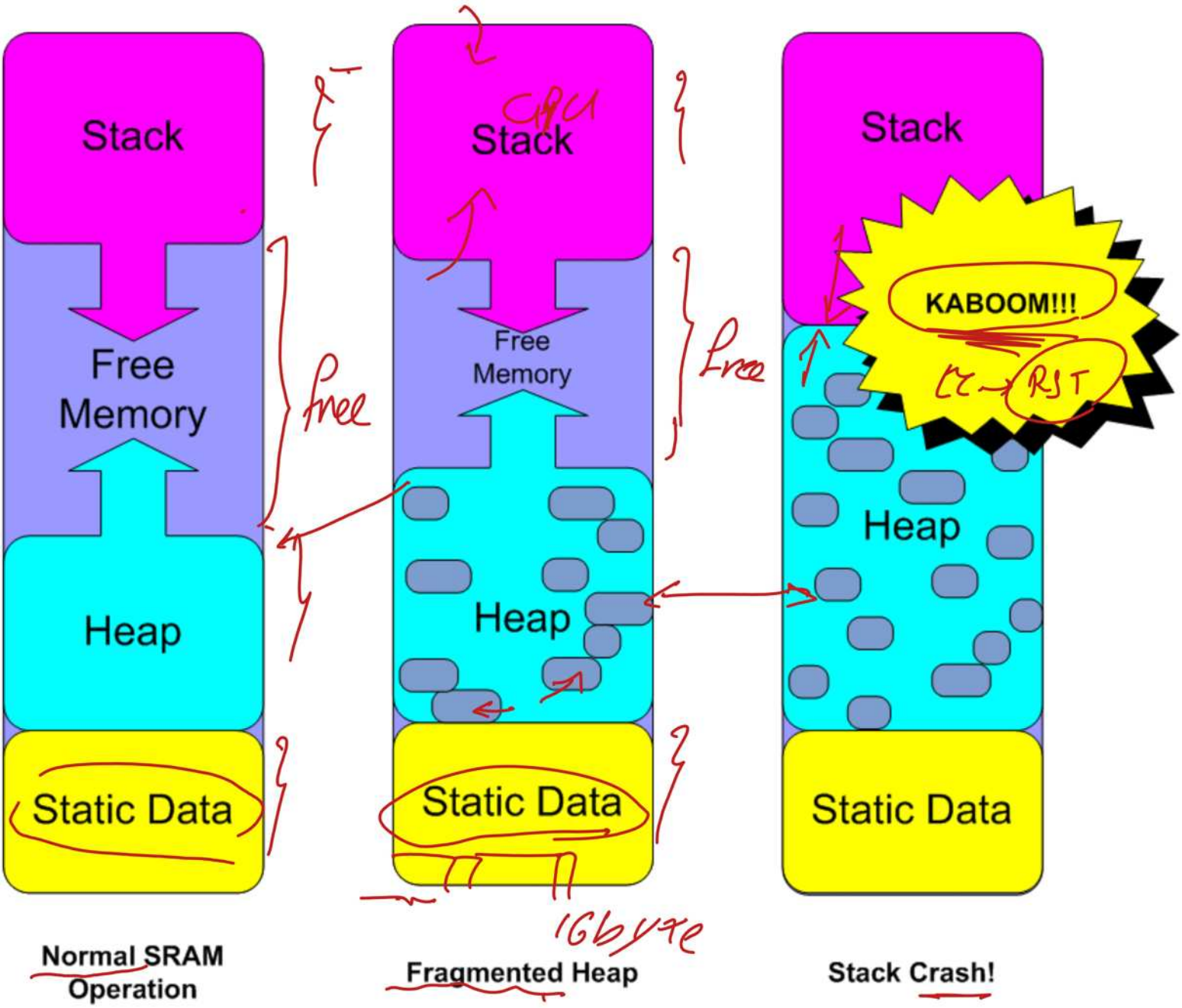



Initialization
Task 1
Task 2
Task 3
ISR Entry
Clear Flags
Execute Code
ISR Exit

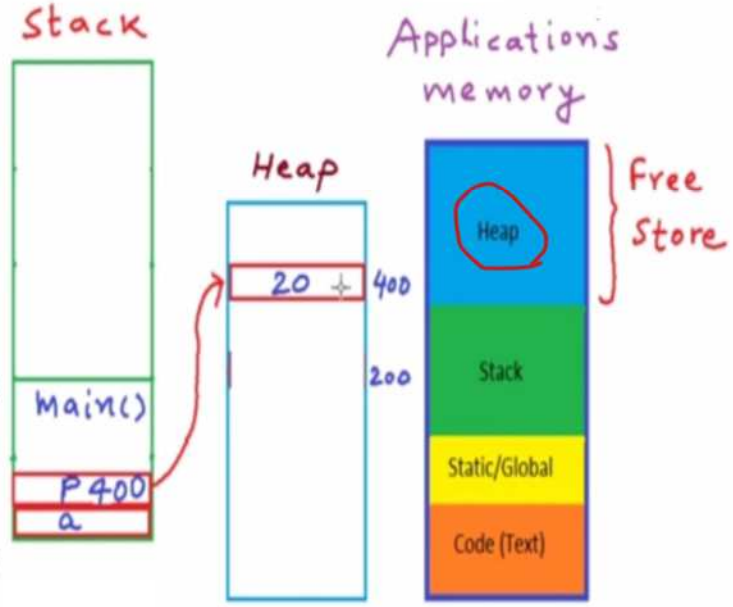

2
ISR
Vector
Table
Main
Program
Flow
Interrupt
Service
Routine
(ISR)
1
3
Interrupt

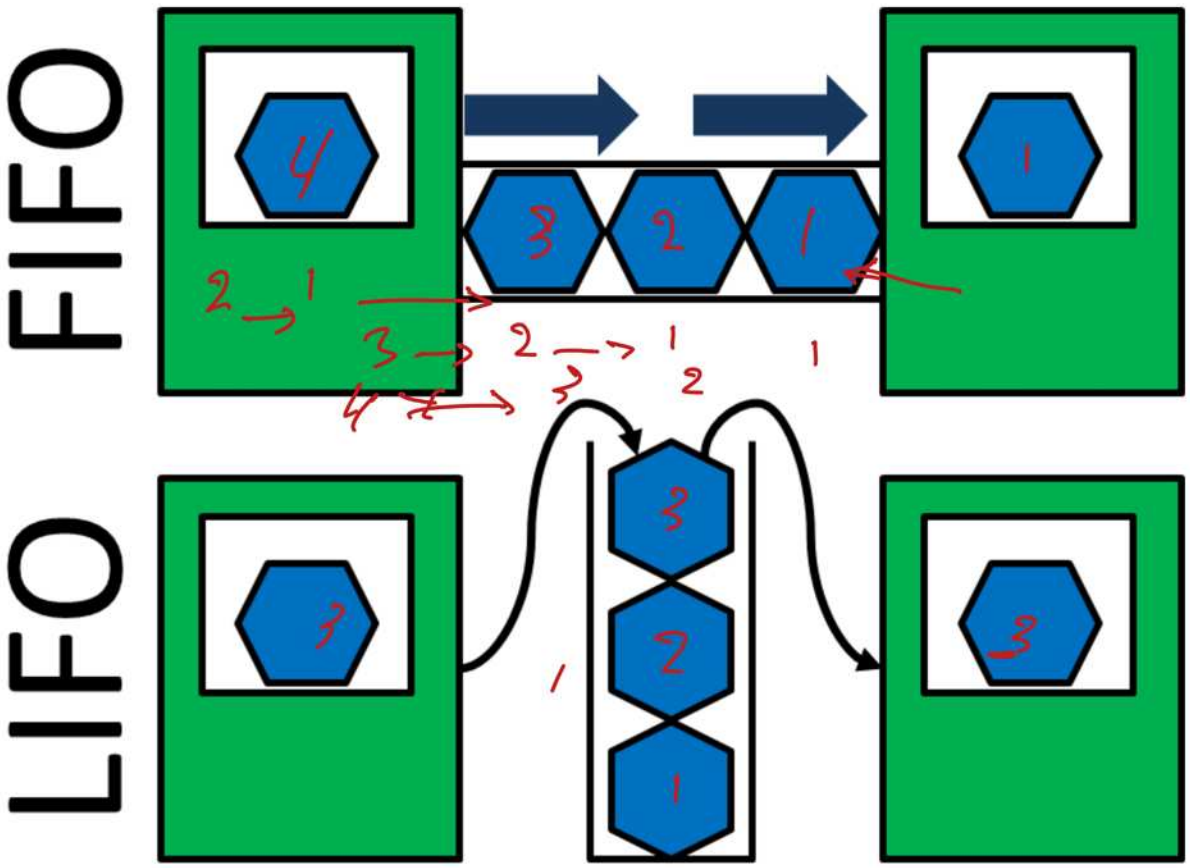

Flash
SRAM
EEPROM
Data Memory
Atmega328P
2KB
0x0100 - 0x08FF
(256 - 2303)
SRAM
0x0060 - 0x00FF
160 Ext I/O Reg.
0x0020 - 0x005F
64 I/O Registers
0x0000 - 0x001F
32 Registers
Stack
Free Memory
Heap
Static Data
Local
Global
uint8_t myVar[100];


Top
writable; not executable
Stack
Managed "automatically"
(by compiler)
writable; not executable
Dynamic Data
(Heap)
Managed by programmer
writable; not executable
Static Data
Initialized when process starts
Read-only; not executable
Literals
Initialized when process starts
Read-only; executable
Instructions
Initialized when process starts

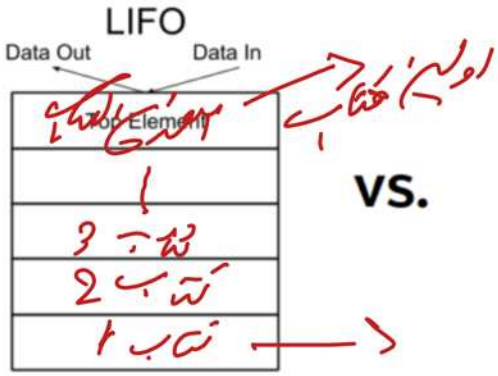



```c
#include<stdio.h>
#include<stdlib.h>
int main()
{
  int a; // goes on stack
  int *p;
  p = (int*)malloc(sizeof(int));
  *p = 10;
  free(p);
  p = (int*)malloc(sizeof(int));
  *p = 20;
```





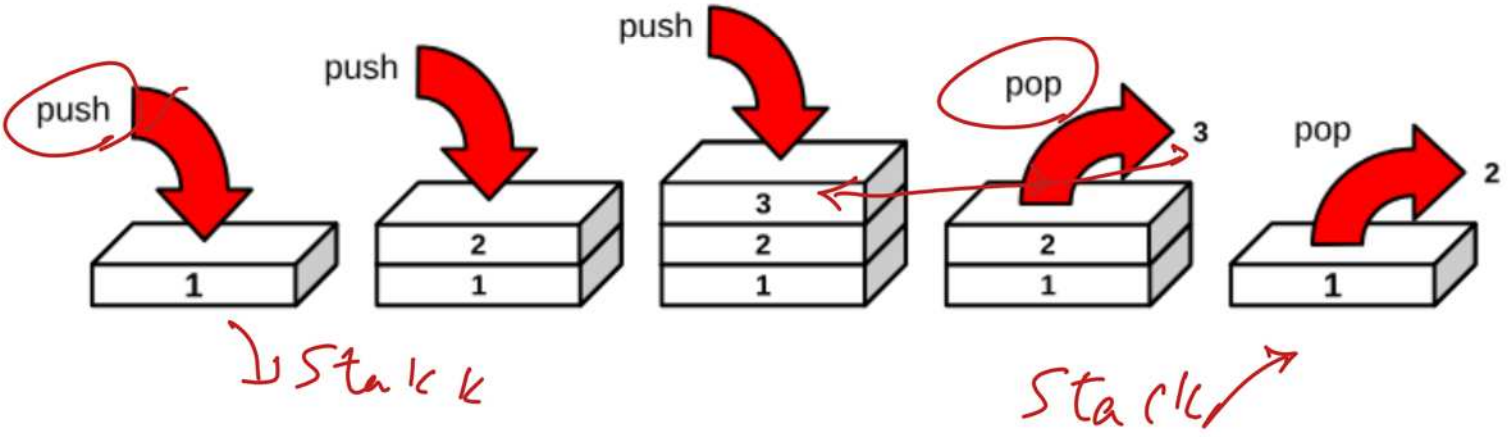
FIFO
LIFO



vs.

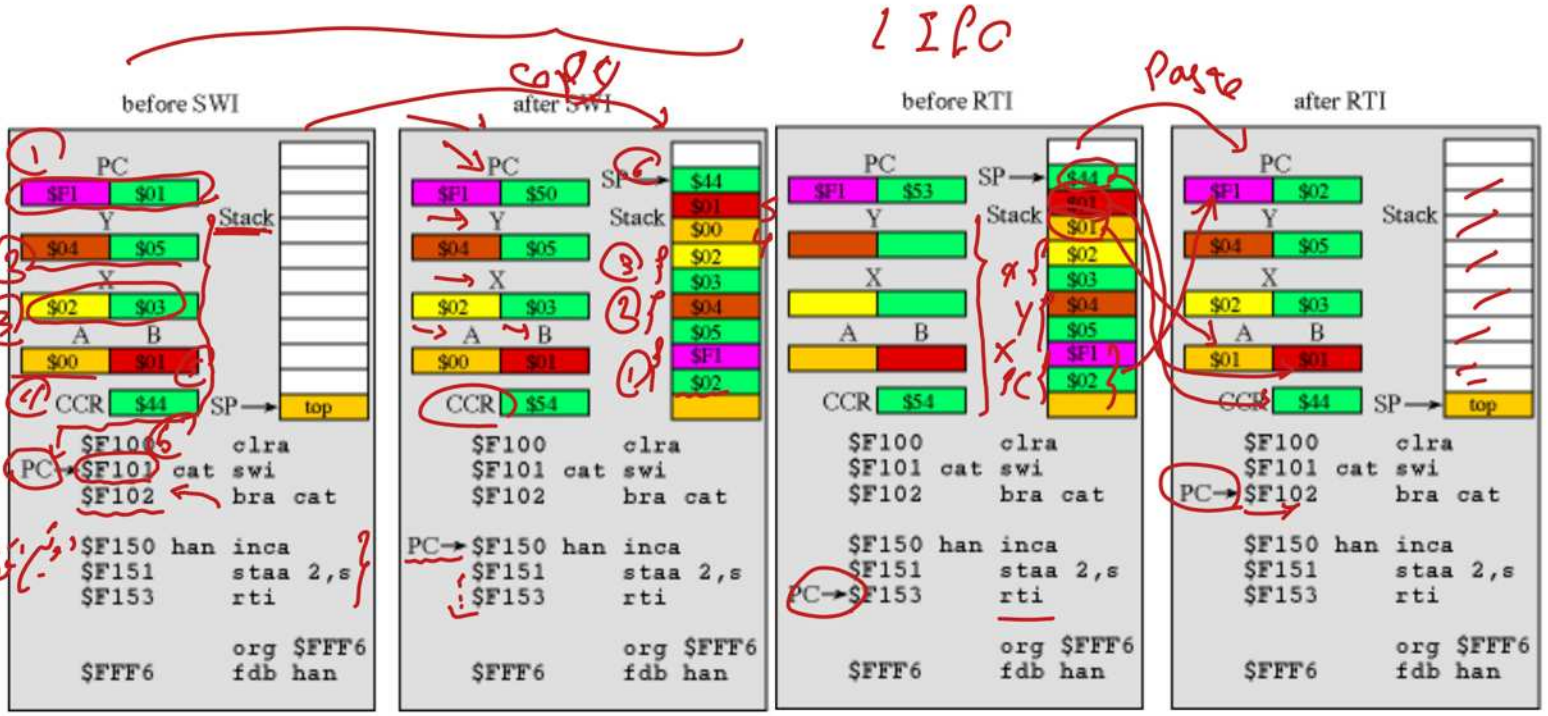
Shift Reg

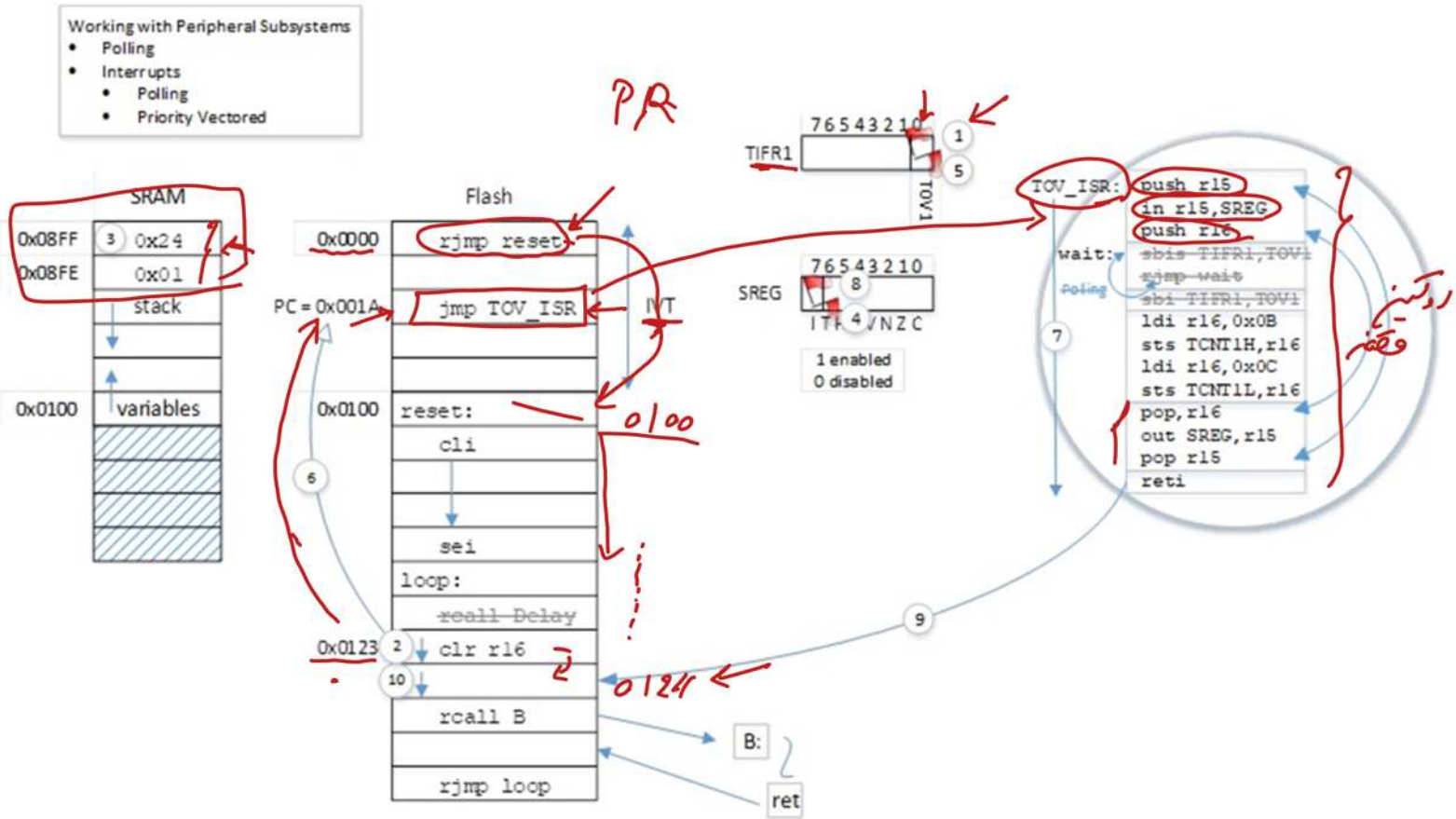
LIFO: Last In First Out



Stack

Stack


before SWI
after SWI
before RTI
after RTI
Stack
PC
Y
X
A
B
CCR
SP
top
$F100 clra
$F101 cat swi
$F102 bra cat
$F150 han inca
$F151 staa 2,s
$F153 rti
org $FFF6
$FFF6 fdb han


Working with Peripheral Subsystems
Polling
Interrupts
Polling
Priority Vectored
SRAM
Flash
0x08FF
0x08FE
0x24
0x01
stack
variables
0x0100
0x0000
rjmp reset
jmp TOV_ISR
IVT
PC = 0x001A
reset:
cli
sei
loop:
rcall Delay
clr r16
rcall B
rjmp loop
B:
ret
0x0123
TIFR1
TOV1
SREG
1 enabled
0 disabled
TOV_ISR:
push r15
in r15,SREG
push r16
wait:
sbis TIFR1,TOV1
rjmp wait
Poling
sbi TIFR1,TOV1
ldi r16,0x0B
sts TCNT1H,r16
ldi r16,0x0C
sts TCNT1L,r16
pop,r16
out SREG,r15
pop r15
reti

12.4 Interrupt Vectors in ATmega328 and ATmega328P

Table 12-6. Reset and Interrupt Vectors in ATmega328 and ATmega328P

VectorNo.	Program Address[2]	Source	Interrupt Definition
1	0x0000[1]	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 1
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	SPI_STC	SPI Serial Transfer Complete
19	0x0024	USART_RX	USART Rx Complete
20	0x0026	USART_UDRE	USART, Data Register Empty
21	0x0028	USART_TX	USART, Tx Complete
22	0x002A	ADC	ADC Conversion Complete
23	0x002C	EE_READY	EEPROM Ready
24	0x002E	ANALOG_COMP	Analog Comparator
25	0x0030	TWI	2-wire Serial Interface
26	0x0032	SPM_Ready	Store Program Memory Ready

NVIC Configuration

NVIC | Code generation

Priority Group: 4 bits for pre-emption priority 0 bits for subpriority

Sort by Premption Priority and Sub Prority

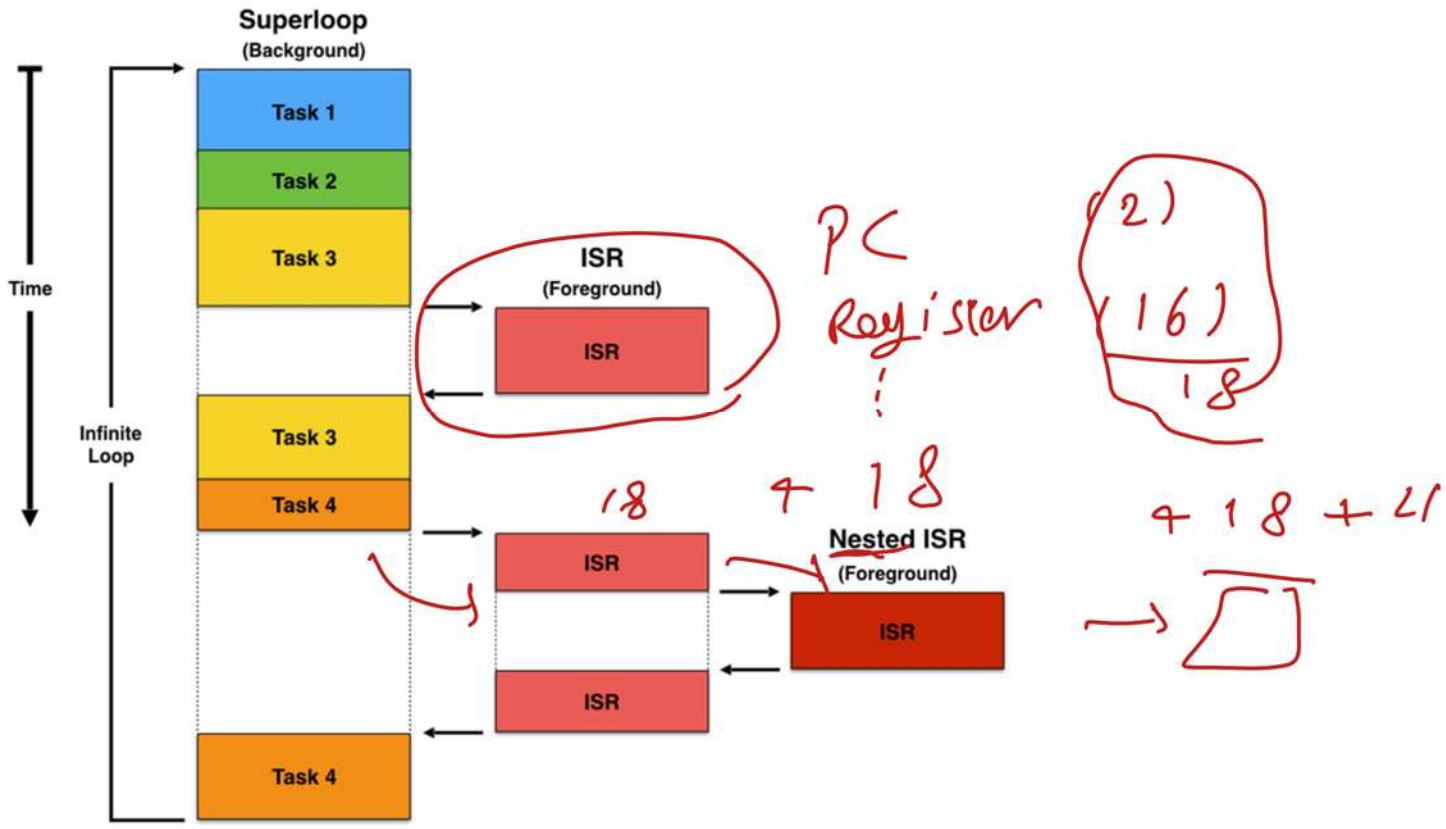
Search: Search (Crtl+F)

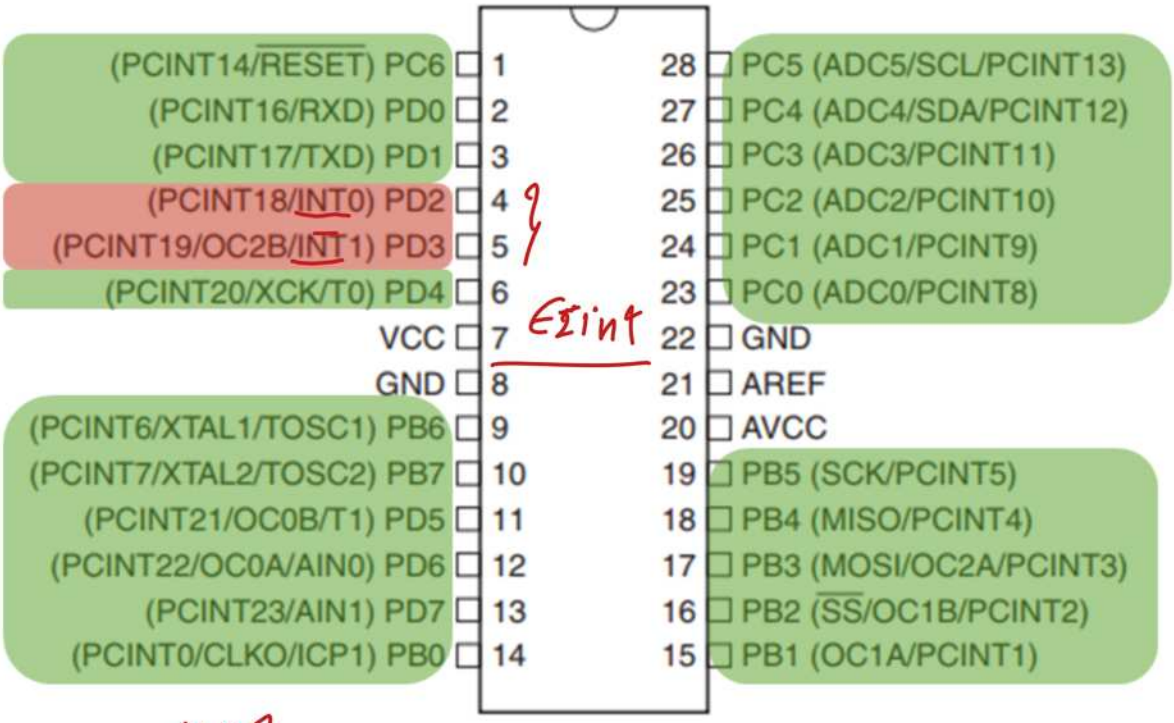
Show only enabled interrupts

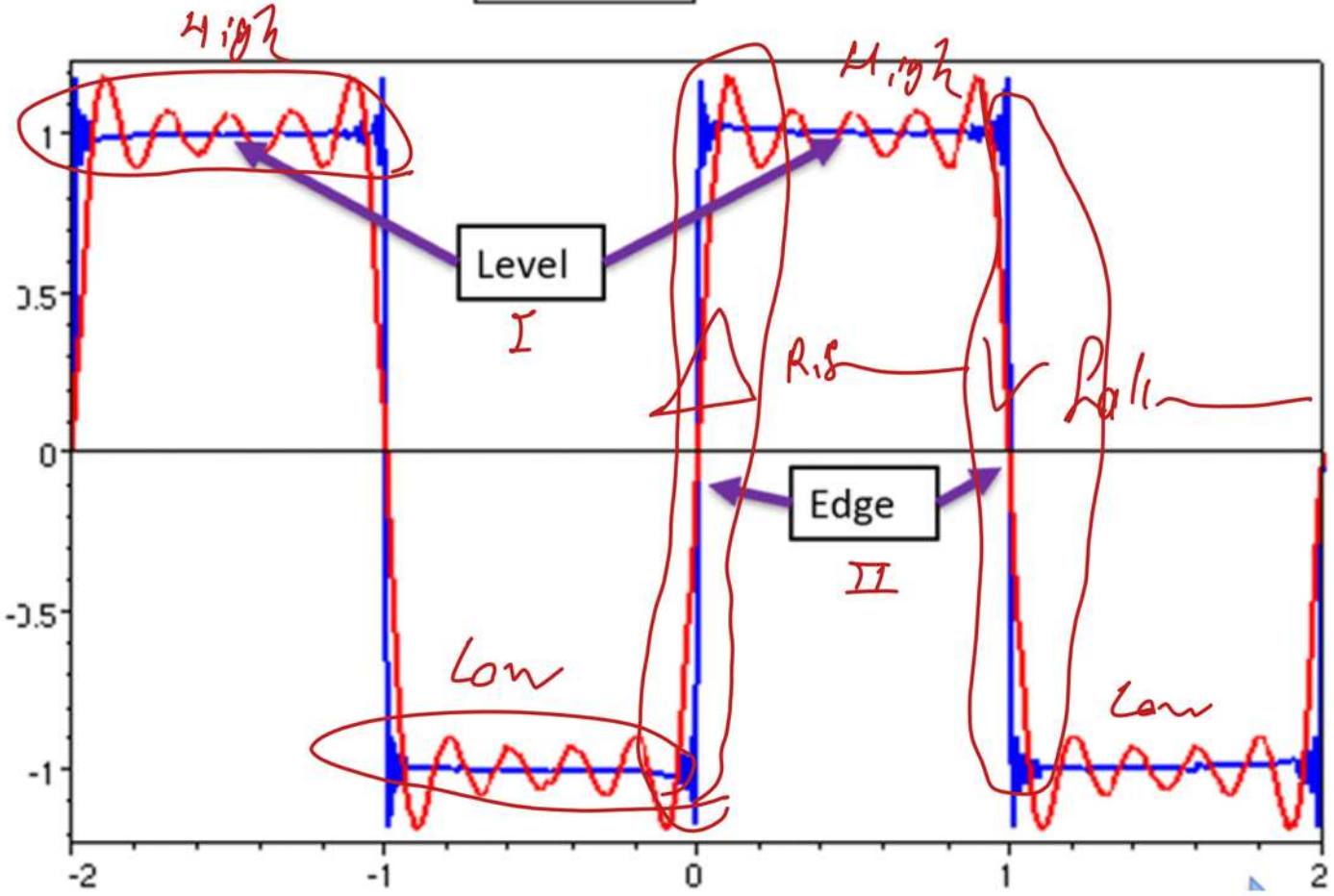
Interrupt Table	Enabled	Preemption Priority	Sub Priority
Non maskable interrupt	☑	0	0
Hard fault interrupt	☑	0	0
Memory management fault	☑	0	0
Prefetch fault, memory access fault	☑	0	0
Undefined instruction or illegal state	☑	0	0
System service call via SWI instruction	☑	0	0
Debug monitor	☑	0	0
Pendable request for system service	☑	0	0
Time base: System tick timer	☑	0	0
PVD interrupt through EXTI line 16	☐	0	0
Flash global interrupt	☐	0	0
RCC global interrupt	☐	0	0
SPI1 global interrupt	☐	0	0
USART1 global interrupt	☐	0	0

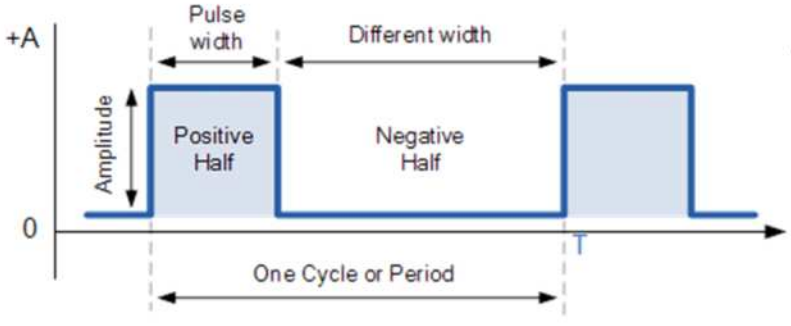
Enabled | Preemption Priority | Sub Priority

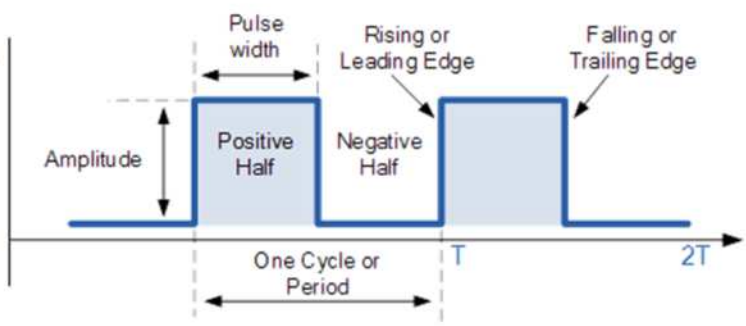
Apply | Ok | Cancel




(PCINT14/RESET) PC6 1
(PCINT16/RXD) PD0 2
(PCINT17/TXD) PD1 3
(PCINT18/INT0) PD2 4
(PCINT19/OC2B/INT1) PD3 5
(PCINT20/XCK/T0) PD4 6
VCC 7
GND 8
(PCINT6/XTAL1/TOSC1) PB6 9
(PCINT7/XTAL2/TOSC2) PB7 10
(PCINT21/OC0B/T1) PD5 11
(PCINT22/OC0A/AIN0) PD6 12
(PCINT23/AIN1) PD7 13
(PCINT0/CLKO/ICP1) PB0 14
28 PC5 (ADC5/SCL/PCINT13)
27 PC4 (ADC4/SDA/PCINT12)
26 PC3 (ADC3/PCINT11)
25 PC2 (ADC2/PCINT10)
24 PC1 (ADC1/PCINT9)
23 PC0 (ADC0/PCINT8)
22 GND
21 AREF
20 AVCC
19 PB5 (SCK/PCINT5)
18 PB4 (MISO/PCINT4)
17 PB3 (MOSI/OC2A/PCINT3)
16 PB2 (SS/OC1B/PCINT2)
15 PB1 (OC1A/PCINT1)


High
High
Level
I
Edge
II
Low
Low
1
0.5
0
-0.5
-1
-2
-1
0
1
2


+A
Pulse width
Different width
Amplitude
Positive Half
Negative Half
0
T
One Cycle or Period


+A
Pulse width
Rising or Leading Edge
Falling or Trailing Edge
Amplitude
Positive Half
Negative Half
0
T
2T
One Cycle or Period