

KISS = Keep it simple, stupid!



\$ bit def.h

\$ io AT89C52.h

```

LED rojo      EQU    P1.0
LED verde     EQU    P1.1
barrera       EQU    P1.2

n             EQU    40

VECTOR _INTXT0 EQU    0x03 ; ROM
VECTOR _INTXT1 EQU    0x13 ; ROM

LONG_STACK   EQU    10 ; RAM
INICIO_STACK EQU    0x70 ; RAM
Lugares      EQU    60h ; RAM

ASEG
ORG 0x00
LJMP main

ASEG
ORG VECTOR _INTXT0
LJMP intExtE

ASEG
ORG VECTOR _INTXT1
LJMP intExtS
  
```

```

SEG
ORG INICIO_STACK
stack-init DS LONG_STACK
  
```

```

NAME main
PUBLIC main

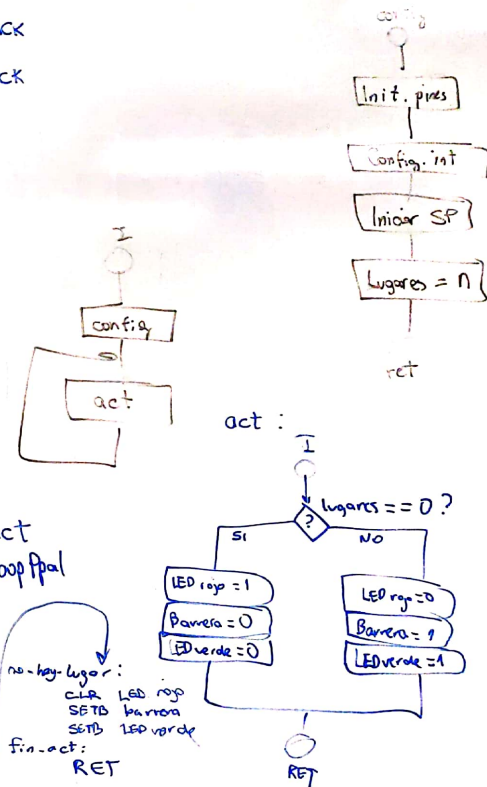
PUBLIC intExtE
PUBLIC intExtS
PUBLIC config
PUBLIC act
  
```

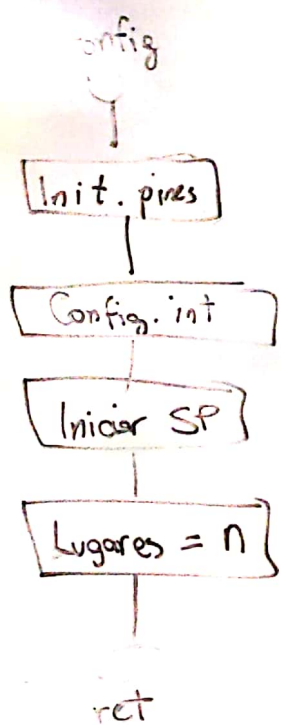
```

main:
    CALL config
loopPpal:
    CALL act
    JMP loopPpal
  
```

```

act:
    MOV A, Lugares
    JZ no-hay-lugar
; hay-lugar:
    SETB LED rojo
    CLR barrera
    CLR LED verde
    JMP fin-act
no-hay-lugar:
    CLR LED rojo
    SETB barrera
    SETB LED verde
fin-act:
    RET
END main
  
```





Config:

```

SETB LEDverde
CLR LEDrojo
SETB barrera
  
```

```
MOV IE, #10000101b
```

```
MOV IP, #0
```

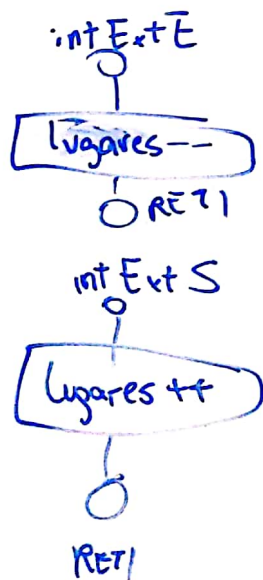
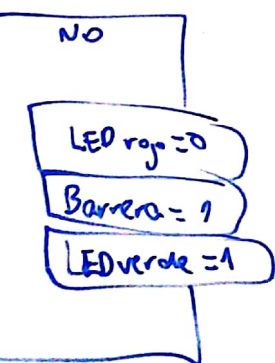
```
MOV TCON, #00000101b
```

```
MOV SP, #(stack_init - 1)
```

```

mov Lugares, #N ; Lugares = N
; mov Lugares, N ; Lugares ← N
RET
  
```

Lugares == 0?



int Ext E:

```
DEC Lugares
```

```
RET
```

int Ext S:

```
INC Lugares
```

```
RET
```