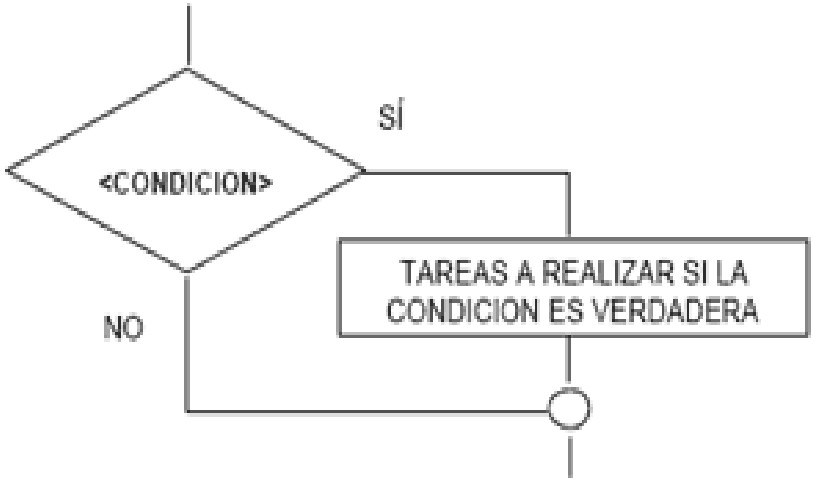
Codificación a partir de pseudocódigo/ordinogramas

Si <condicion>
entonces <instrucciones>
finsi

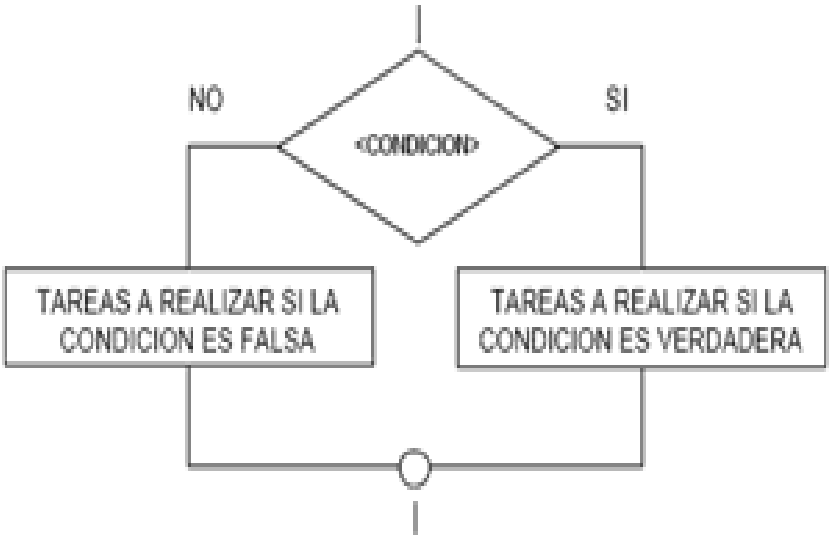



```
Si:  movlw  <valor>
     subwf  <registro>,W
     btfss STATUS,Z
     goto finsi

     <instrucciones>

finsi:
```

La condición siempre será una comparación entre un valor y el contenido de un registro.

Condicionales dobles

```
Si  <condicion>
    <instrucciones1>
sino
   <instrucciones2>
finsi
```

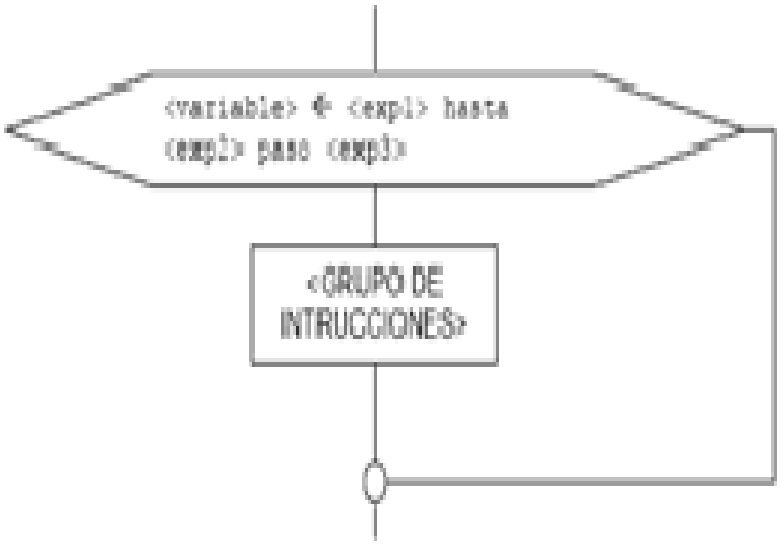



```
Si:  movlw  <valor>
     subwf  <registro>,W
     btfss STATUS,Z
     goto sino
        <instrucciones1>
     goto finsi
sino:
        <instrucciones2>
finsi:
```

La condición siempre será una comparación entre un valor y el contenido de un registro.

Bucles PARA

Para <variable> desde <valor1> hasta <valor2>
    <instrucciones>
finpara



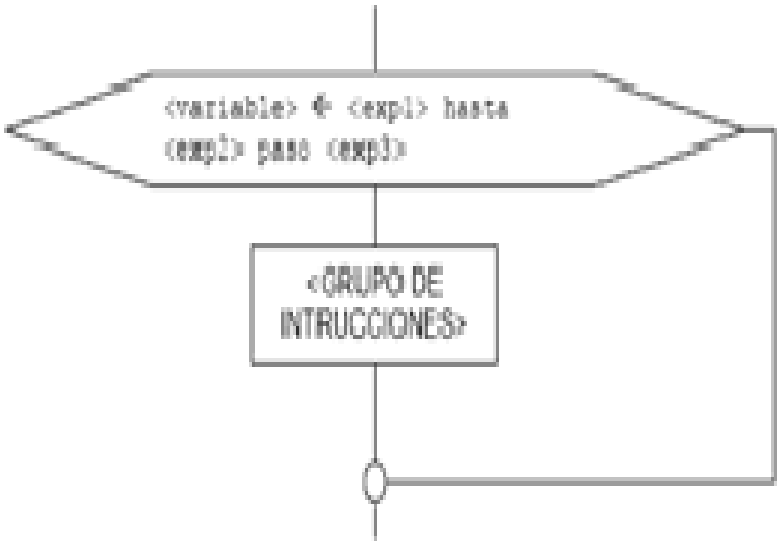

```
movlw <valor1>
movwf <registro>
para:
<instrucciones>

incf <registro>
movlw  <valor2>
subwf  <registro>,W
btfss STATUS,Z
goto para
```

Siendo <valor2> mayor que <valor1>, si es al revés, en vez de incf se usará decf. Además se supone que el valor se incrementa de uno en uno

Bucles PARA

Para <variable> desde <valor1> hasta 0
    <instrucciones>
finpara

```
        movlw <valor1>
        movwf <registro>
para:
        <instrucciones>

        decfsz <registro>
        goto para
```

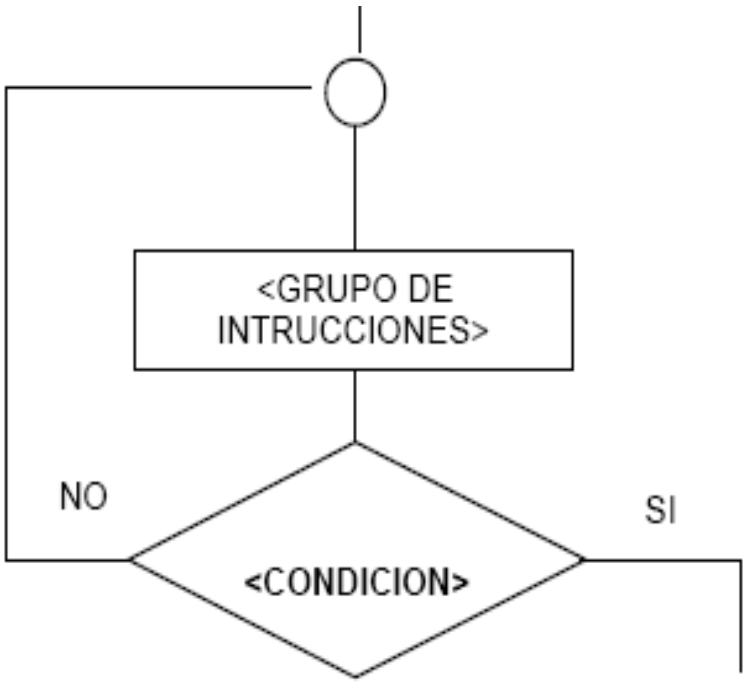


En este caso se irá decrementando desde el valor hasta 0.

Es un caso especial que permite simplificar el código ensamblador

Bucles repetir-hasta

Repetir
    <instrucciones>
hasta <condición>

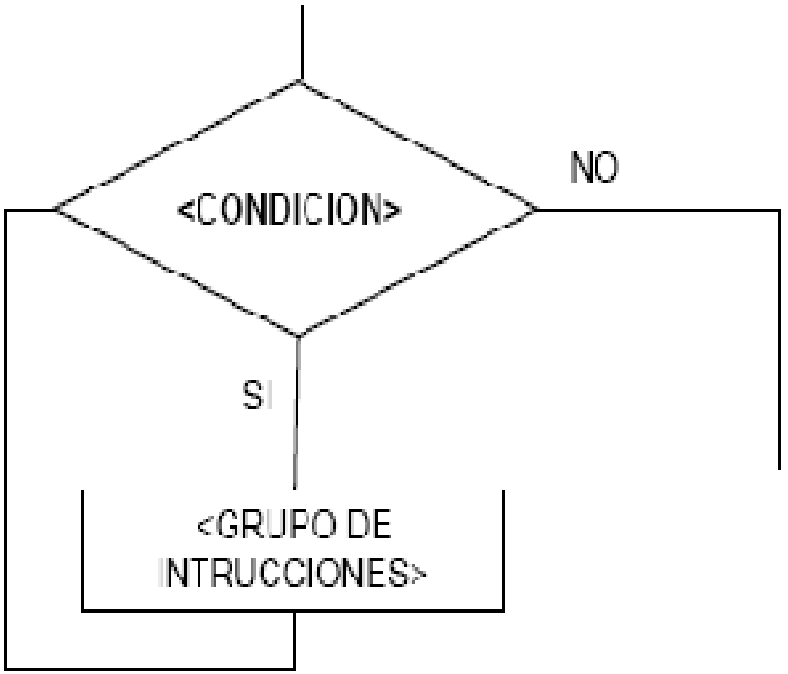



```
repetir:
        <instrucciones>
        movlw  <valor>
        subwf  <registro>,W
        btfss STATUS,Z
        goto bucle
```

La condición siempre será una comparación entre un valor y el contenido de un registro.

Bucles mientras-que

mientras que <condicion>
    <instrucciones>
finmientras




```
mientras:
        movlw  <valor>
           subwf  <registro>,W
           btfsc STATUS,Z
           goto finmientras

           <instrucciones>

           goto mientras

finmientras:
```

La condición siempre será una comparación entre un valor y el contenido de un registro.