

```

001 ;*****
002 ;* TEST SENSORS INFRAROIG_IR(per interrupcions) versio 3 Lleonard Garcia i Llop*
003 ;*****
004 ; 1.-Per estalviar energia es pot col·locar un led bicolor en lloc del motor.
005 ; 2.-BRIDGE MOTORS (4 semibridges amb SN754410NE)
006 ;   Bridge= pont   Es una connexio en que de forma de H (el trem vertical
007 ;   de la dreta, el tram mig horitzontal i el tram vertical de l'esquerra).
008 ;   El Semibridge_a, en funcio d'un 0 o 1, connecta a + o - el born_a del motor.
009 ;   El Semibridge_b, en funcio d'un 0 o 1, connecta a + o - el born_b del motor.
010 ;   Semi_bridge_a Semi_bridge_M1b Motor          Leds
011 ;       0 (-)      0 (-)      ATURAT            APAGAT
012 ;       1 (+)      0 (-)      CCWS (gira en sentit invers) Verd
013 ;       0 (-)      1 (+)      CWS (gira envant)    Vermell
014 ;       1 (+)      1 (+)      ATURAT            APAGAT
015 ; 3.-El sistema te un defecte. Si gira envant i immediatament es dona instruccio
016 ;   de girar enrera, te el mateix efecte que un curtcircuit. Es recomana afegir
017 ;   un breu retard i fer que el temps amortitzi l'energia acumulada en el motor
018 ;   en moviment.
019 ; 4.-CIRCUIT IMPRES QUE UTILITZA PICAXE 14M2
020 ;           *****
021 ;           +V * 1           14 * GND
022 ;           SI /C.5 * 2       13 * B.0 /SO bridge M2 (canviar jumper)
023 ;   bridge M1   C.4 * 3       12 * B.1   bridge M2
024 ;   boto_PUJAR C.3 * 4       11 * B.2   SENSOR_INFERIOR
025 ;   bridge M1   C.2 * 5       10 * B.3   SENSOR_SUPERIOR / I2C_SCL
026 ;   boto_BAIXAR C.1 * 6       9 * B.4   LED_SUPERIOR / I2C_SDA
027 ;   IR_TSOP4838 C.0 * 7       8 * B.5   LED_INFERIOR
028 ;           *****
029 pause 1000; permet programacio segura durant 1 segon despres de connectar
030 ;*****
031 ;*****
032 ;CONFIGURACIONS DEL HARDWARE
033 let dirsC=%00010100 ; Definicions del port_C (1=sortida 0=entrada).
034 let dirsB=%00110011 ; Definicions del port_B (1=sortida 0=entrada).
035 symbol bridge_M1a=C.2 ; SORTIDA.Semi-bridge_a del motor_M1.
036 symbol bridge_M1b=C.4 ; SORTIDA.Semi-bridge_b del motor_M1.
037 symbol bridge_M2a=B.0 ; SORTIDA.Semi-bridge_a del motor_M2.
038 symbol bridge_M2b=B.1 ; SORTIDA.Semi-bridge_b del motor_M2.
039 symbol PUJAR=C.3 ; ENTRADA. Boto de pujar.
040 symbol pujant=pinc.3 ; ENTRADA. Boto de pujar.
041 symbol BAIXAR=C.1 ; ENTRADA. Boto de baixar.
042 symbol baixant=pinc.1 ; ENTRADA. Boto de baixar.
043 symbol led_inferior=B.5 ; SORTIDA. Led emisor1.
044 symbol led_superior=B.4 ; SORTIDA. Led emisor2.
045 symbol fiCursa_inferior=B.2 ; ENTRADA. Fototransistor_1
046 symbol fiCursa_superior=B.3 ; ENTRADA. Fototransistor_2
047 symbol sensor=C.0 ; ENTRADA. Sensor infrarroig TSOP4838.
048 ;DEFINICIO DE CONSTANTS.
049 ;DEFINICIO DE VARIABLES
050 symbol codi_IR=b0
051 symbol codi_IR_=b1
052 ;PROGRAMA INICIAL
053 low bridge_M1a, bridge_M1b, bridge_M2a, bridge_M2b ;inicialment motors aturats.
054 setint %00000000,%00000001 ; Permetre interrupcions per C.0 a nivell baix.
055 ;PROGRAMA PRINCIPAL O REITERATIU
056 goto main2; ANAR A VERSIO DE PROGRAMA 2
057 main:
058 ; l'avaluacio de select case es fa amb variable que només s'actualitza a l'inici.
059 if codi_IR <> codi_IR_ then ; Hi ha algun canvi? S'ha rebut un codi diferent?
060 let codi_IR=codi_IR_ ; (SI) Actualitza la variable codi_IR
061 else
062 goto main ; (NO) Tornar a preguntar fins que hi hagi un
canvi.
063 endif
064 if codi_IR=13 then ; El codi_IR = 13?
065 call pujar1 ; (SI) Pujar_1 o encendre_verd.

```

```

066 let codi_IR=0 ; Assenyalar que l'ordre ja ha esta atesa.
067 else
068 endif
069 if codi_IR=22 then ; El codi_IR = 22?
070 call baixar1 ; (SI) Baixar_1 o encendre_vermell.
071 let codi_IR=0 ; Assenyalar que l'ordre ja ha esta atesa.
072 else
073 endif
074 if codi_IR=16 then ; El codi_IR = 16?
075 call pujar2 ; (SI) Pujar_2 o encendre_verd.
076 let codi_IR=0 ; Assenyalar que l'ordre ja ha esta atesa.
077 else
078 endif
079 if codi_IR=17 then ; El codi_IR = 17?
080 call baixar2 ; (SI) Baixar_2 o encendre_vermell.
081 let codi_IR=0 ; Assenyalar que l'ordre ja ha esta atesa.
082 else
083 endif
084 if codi_IR<>0 then ; S'ha premut un codi diferent als anteriors?
085 call aturar1 ; (SI) Aturar_1 o apagar_1
086 call aturar2 ; Aturar_2 o apagar_2
087 let codi_IR=0 ; Assenyalar que l'ordre ja ha esta atesa.
088 else
089 endif
090 goto main
091
092 main2:
093 ; l'avaluacio de select case es fa amb variable que nomes s'actualitza a l'inici.
094 if codi_IR <> codi_IR_ then ; Hi ha algun canvi? S'ha rebut un codi diferent?
095 let codi_IR=codi_IR_ ; (SI) Actalitza la variable codi_IR
096 else
097 goto main2 ; (NO) Tornar a preguntar fins que hi hagi un
canvi.
098 endif
099 select case codi_IR
100 case 13 ; codi_IR_=13?
101 call pujar1 ; (SI) Pujar_1 o encendre_verd.
102 case 22 ; codi_IR_=22?
103 call baixar1 ; (SI) Baixar_1 o encendre_vermell.
104 case 16 ; codi_IR_=16?
105 call pujar2 ; (SI) Pujar_2 o encendre_verd.
106 case 17 ; codi_IR_=17?
107 call baixar2 ; (SI) Baixar_1 o encendre_vermell.
108 else ; codi_IR_=altres?
109 call aturar1 ; Aturar_1 o apagar_1
110 call aturar2 ; Aturar_1 o apagar_1
111 endselect
112 let codi_IR=0
113 goto main2
114 ;SUBROUTINES
115 interrupt:
116 irin [1000,timeout_irin],sensor,codi_IR_ ; llegir TSOP4838 i desar en codi_IR.
117 timeout_irin:
118 setint %00000000,%00000001 ; Reactivar interrupcions per C.0 a nivell baix.
119 return
120 ;SUBROUTINA.MOVIMENT DELS MOTORS.
121 pujar1:
122 ;call aturar1
123 ;pause 100
124 low bridge_M1a
125 let b0=b0
126 high bridge_M1b
127 return
128
129 baixar1:
130 ;call aturar1

```

```
131 ;pause 100
132 high bridge_M1a
133 let b0=b0
134 low bridge_M1b
135 return
136
137 aturar1:
138 low bridge_M1b
139 low bridge_M1a
140 return
141
142 pujar2:
143 ;call aturar2
144 ;pause 100
145 high bridge_M2b
146 let b0=b0
147 low bridge_M2a
148 return
149
150 baixar2:
151 ;call aturar2
152 ;pause 100
153 high bridge_M2a
154 let b0=b0
155 low bridge_M2b
156 return
157
158 aturar2:
159 low bridge_M2b
160 low bridge_M2a
161 return
162
```