

```

001 ;*****
002 ;* CINTA TRANSPORTADORA 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 5000; permet programacio segura durant 1 segon despres de connectar
030 ;*****
031 ;*****
032 ;CONFIGURACIONS DEL HARDWARE
033 setint %00000000,%00000000 ; Reactivar interrupcions per C.0 a nivell baix.
034 let dirsC=%00010100 ; Definicions del port_C (1=sortida 0=entrada).
035 let dirsB=%00110011 ; Definicions del port_B (1=sortida 0=entrada).
036 symbol bridge_M1a=C.2 ; SORTIDA.Semi-bridge_a del motor_M1.
037 symbol bridge_M1b=C.4 ; SORTIDA.Semi-bridge_b del motor_M1.
038 symbol bridge_M2a=B.0 ; SORTIDA.Semi-bridge_a del motor_M2.
039 symbol bridge_M2b=B.1 ; SORTIDA.Semi-bridge_b del motor_M2.
040 symbol PUJAR=C.3 ; ENTRADA. Boto de pujar.
041 symbol pujant=pinc.3 ; ENTRADA. Boto de pujar.
042 symbol BAIXAR=C.1 ; ENTRADA. Boto de baixar.
043 symbol baixant=pinc.1 ; ENTRADA. Boto de baixar.
044 symbol led_inferior=B.5 ; SORTIDA. Led emisor1.
045 symbol led_superior=B.4 ; SORTIDA. Led emisor2.
046 symbol fiCursa_inferior=B.2 ; ENTRADA. Fototransistor_1
047 symbol fiCursa_superior=B.3 ; ENTRADA. Fototransistor_2
048 symbol sensor=C.0 ; ENTRADA. Sensor infrarroig TSOP4838.
049 ;DEFINICIO DE CONSTANTS.
050 ;DEFINICIO DE VARIABLES
051 symbol codi_IR=b0
052 symbol codi_IR_=b3
053 symbol temporitzant=b1 ; variable 0 ? 1. estat del programa
054 symbol tim=b2 ; variable del temporitzador.Comptador.
055 ;PROGRAMA INICIAL
056 low bridge_M1a, bridge_M1b, bridge_M2a, bridge_M2b; inicialment motors aturats.
057 let temporitzant=0
058 ;PROGRAMA PRINCIPAL O REITERATIU
059 ;PROGRAMA INICIAL
060 low bridge_M1a, bridge_M1b, bridge_M2a, bridge_M2b ;inicialment motors aturats.
061 setint %00000000,%00000001 ; Permetre interrupcions per C.0 a nivell baix.
062 ;PROGRAMA PRINCIPAL O REITERATIU
063 main2:
064 ; l'avaluacio de select case es fa amb variable que només s'actualitza a l'inici.
065 if codi_IR <> codi_IR_ then ; Hi ha algun canvi? S'ha rebut un codi diferent?
066 let codi_IR=codi_IR_ ; (SI) Actalitza la variable codi_IR

```

```

067 else
068 goto main2 ; (NO) Tornar a preguntar fins que hi hagi un
canvi.
069 endif
070 select case codi_IR
071 case 13 ; codi_IR_=13?
072 call baixar1 ; (SI) Pujar_1 o encendre_verd.
073 case 22 ; codi_IR_=22?
074 call pujar1 ; (SI) Baixar_1 o encendre_vermell.
075 else ; codi_IR_=altres?
076 call aturar1 ; Aturar_1 o apagar_1
077 endselect
078 let codi_IR=0
079 goto main2
080 ;SUBROUTINES
081 interrupt:
082 irin [1000,timeout_irin],sensor,codi_IR_ ; llegir TSOP4838 i desar en codi_IR.
083 timeout_irin:
084 setint %00000000,%00000001 ; Reactivar interrupcions per C.0 a nivell baix.
085 return
086 ;SUBROUTINA.MOVIMENT DELS MOTORS.
087 pujar1:
088 ;call aturar1
089 ;pause 100
090 low bridge_M1a
091 let b0=b0
092 high bridge_M1b
093 return
094
095 baixar1:
096 ;call aturar1
097 ;pause 100
098 high bridge_M1a
099 let b0=b0
100 low bridge_M1b
101 return
102
103 aturar1:
104 low bridge_M1b
105 low bridge_M1a
106 return
107
108 pujar2:
109 ;call aturar2
110 ;pause 100
111 high bridge_M2b
112 let b0=b0
113 low bridge_M2a
114 return
115
116 baixar2:
117 ;call aturar2
118 ;pause 100
119 high bridge_M2a
120 let b0=b0
121 low bridge_M2b
122 return
123
124 aturar2:
125 low bridge_M2b
126 low bridge_M2a
127 return
128

```