

G:\SYLVIELLY_MICROCONTROLADORES\TRABALHOS FINALIZADOS\sylvially_microcontroladores_esta

```
1: /*MICROCONTROLADORES - S6 ENGENHARIA DE CONTROLE E AUTOMACAO*/
2: /*TRABALHO 03 - CONTROLE DE ESTACIONAMENTO 10 VAGAS EM C*/
3: //ALUNA: SYLVIELLY S. SOUSA
4:
5: #include<18f4550.h> //PIC18F4550
6: #fuses hs, nowdt, CPUDIV1, NOMCLR, noprotect, nolvp //habilitacao dos :
7: //hs = high speed (alta vel.
8: //noprotect = codigo nao pr
9: //NOMCLR = desabilita maste
10: //nolvp = desabilita low vo.
11: //nowtd = desabilita watchd
12:
13:
14: //DEFINICOES DE PINOS PARA CANCELA DE ENTRADA
15: #define SENSOR_0 pin_b0 //sensor de abertura da cancela
16: #define SENSOR_1 pin_b1 //sensor em que indica que o carro
17: #define CANCELA_ENT pin_d0
18: #define CHEIO pin_d7
19: //DEFINICOES DE PINOS PARA CANCELA DE SAIDA
20: #define SENSOR_2 pin_b2
21: #define SENSOR_3 pin_b3
22: #define CANCELA_SAIDA pin_d1
23:
24: #use delay(clock=2000000) //CLOCK = 4MHz
25:
26: void main()
27: {
28: {
29:
30: //DECLARACAO DE VARIAVEIS
31: int VAGAS_OCUPADAS=0; //estacionamento vazio
32: int ESTADO_CANCELA_ENTRADA=0;
33: int ESTADO_CANCELA_SAIDA=0;
34: int AUX_CANCELA_ENT=0;
35: int AUX_CANCELA_SAIDA=0;
36:
37: port_b_pullups(TRUE);
38: //INICIO WHILE
39: while(TRUE) {
40:
41:
42: //INICIO LOGICA CANCELA DE ENTRADA
43: if( (input(SENSOR_0)==0) && (VAGAS_OCUPADAS<=9) ) //test.
44: {
45: output_high(CANCELA_ENT);
46: AUX_CANCELA_ENT=1; //varia
47: }
48:
49: if( (AUX_CANCELA_ENT==1) && (input(SENSOR_1)==0) ) //captura est.
50: {
51: ESTADO_CANCELA_ENTRADA=1;
52: }
53:
54: if( (ESTADO_CANCELA_ENTRADA==1) && (input(SENSOR_1)==1) )
55: {
56: delay_ms(500);
57: if( (ESTADO_CANCELA_ENTRADA==1) && (input(SENSOR_1)==1) )
58: {
59: output_low(CANCELA_ENT); //desce cancela
60: VAGAS_OCUPADAS++; //incrementa vagas
61: ESTADO_CANCELA_ENTRADA=0;
62: AUX_CANCELA_ENT=0;
63: }
64: }
65:
66:
67: if (VAGAS_OCUPADAS==5)
68: {
```

```

G:\SYLVIELLY_MICROCONTROLADORES\TRABALHOS FINALIZADOS\sylvielly_microcontroladores_esta
69:         output_high(CHEIO);           //liga led indicando que nao :
70:     }
71:
72:
73: //FIM LOGICA CANCELA DE ENTRADA
74:
75: //INICIO LOGICA CANCELA DE SAIDA
76:     if((input(SENSOR_2)==0)&&(VAGAS_OCUPADAS>0))
77:     {
78:         output_high(CANCELA_SAIDA);
79:         AUX_CANCELA_SAIDA=1;
80:     }
81:
82:     if((AUX_CANCELA_SAIDA==1)&&(input(SENSOR_3)==0))           //
83:     {
84:         ESTADO_CANCELA_SAIDA=1;
85:     }
86:
87:     if((ESTADO_CANCELA_SAIDA==1)&&(input(SENSOR_3)==1))
88:     {
89:         delay_ms(500);
90:         if((ESTADO_CANCELA_SAIDA==1)&&(input(SENSOR_3)==1))
91:         {
92:             output_low(CANCELA_SAIDA);           //desce cancela
93:             VAGAS_OCUPADAS--;                     //decrementa vagas
94:             ESTADO_CANCELA_SAIDA=0;
95:             AUX_CANCELA_SAIDA=0;
96:         }
97:     }
98:
99:
100:     if (VAGAS_OCUPADAS==0)
101:     {
102:         output_low(CHEIO);           //desliga led indicando que ha
103:     }
104:
105: //FIM LOGICA CANCELA DE SAIDA
106: } //FIM WHILE
107:
108: } //FIM PROGRAMA
109:
110:
111:

```