

Hardware Base

O hardware utilizado no curso é a Placa de Testes EDT-044 que utiliza o microcontrolador AT89S52.

Este microcontrolador possui 32 pinos de I/O divididos em 4 ports de 8 bits Estes ports são bit/byte endereçáveis, ou seja, podemos acessar o port (P0, P1, P2, P3) ou cada bit individualmente (P0_0, P0_1, P0_2, P0_3, P0_4, P0_5, P0_6, P0_7, P1_0 ... P1_7, P2_0 ... P2_7, P3_0 ... P3_7).

Na Placa de Testes são utilizadas 9 entradas e 9 saídas, a saber:

Entradas:

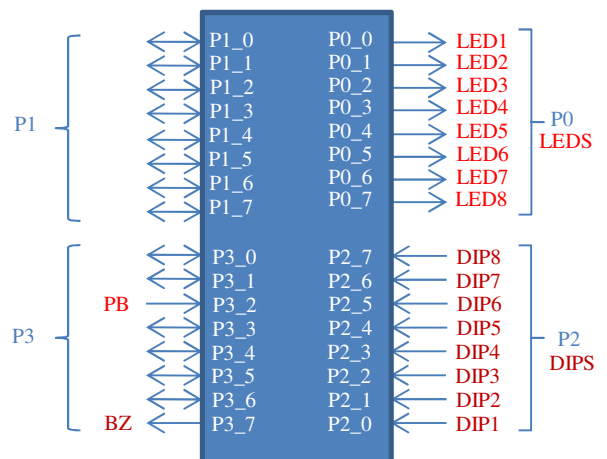
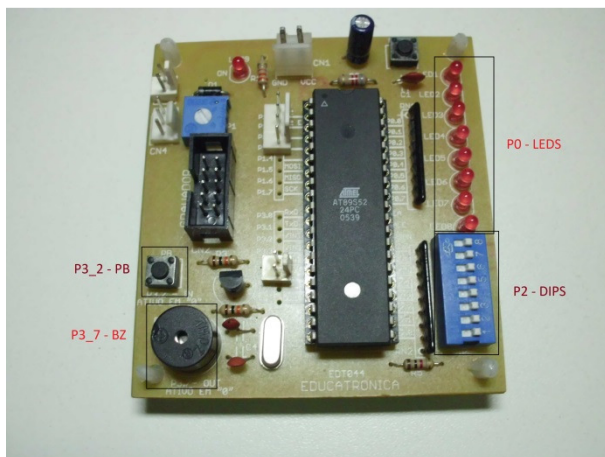
P2 – Dip-switch com 8 chaves (ON = nível lógico “0”)

P3_2 – Push-Button (acionada = nível lógico “0”)

Saídas:

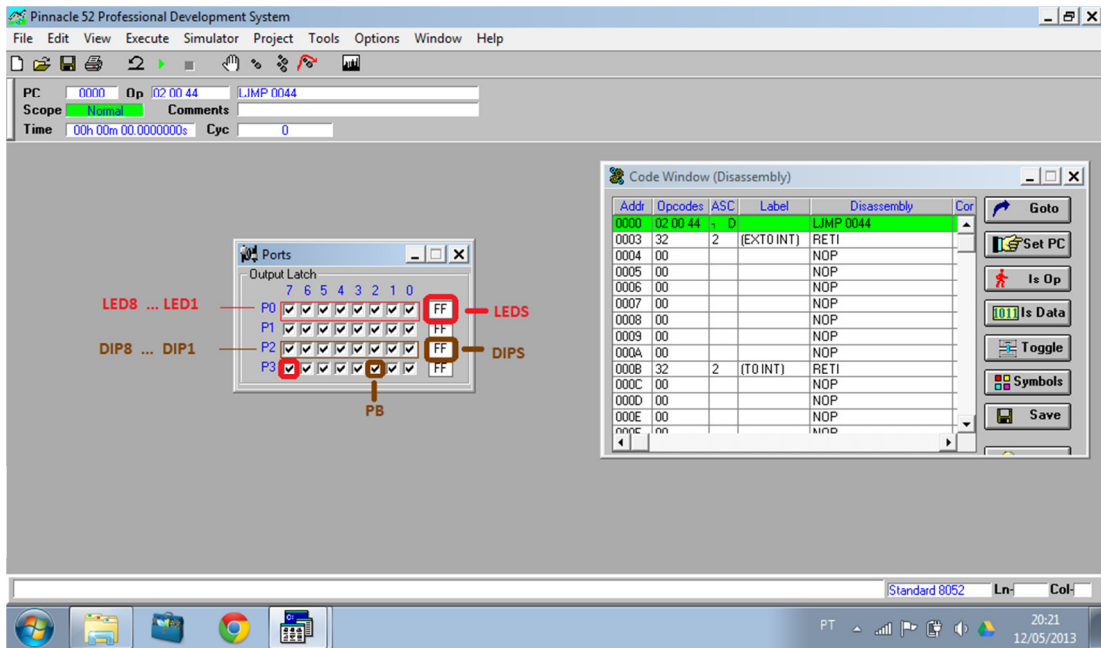
P0 – 8 Leds (ativos em nível lógico “1”)

P3_7 – Buzzer (ativo em nível lógico “0”)



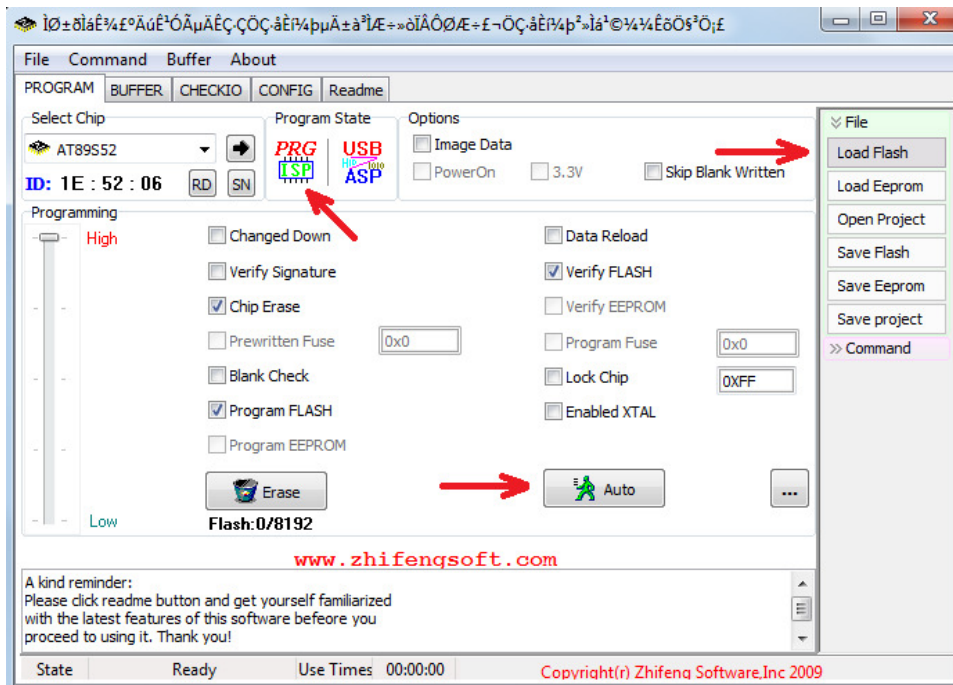
Simulação

- Abrir o programa Pinnacle52 (Iniciar – Programas – Pinnacle – Pinnacle52 – register later)
- Abrir Code Memory (View - Code Memory) – otimizar janela.
- Abrir Ports (View – Ports) – otimizar janela.
- Abrir o Programa – (File open – Arquivos do tipo: INTEL HEX – PROG1.HEX)
- Executar programa (seta verde - run)
- Simular entradas (DIPS e PB) e verificar saídas (LEDS e BZ)



Programação na Placa de Testes

- Abrir o Gravador USB (C:\8051\Gravadores\Gravador USB)
- Conectar o cabo USB do Kit Programador
- Carregar o Arquivo .HEX (LOAD FLASH)
- Gravar o programa (AUTO)
- Simular o programa alterando o estado da Dips-Switch e PB e verificando os LEDS e BZ.



Revisão

Função OR
S = A | B

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

Função AND
S = A & B

A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

Função XOR
S = A ^ B

A	B	S
0	0	0
0	1	1
1	0	0
1	1	1

Função NOT
S = !A

A	S
0	1
1	0

Tabela de Conversão

Binário				Hexa	Decimal
0	0	0	0	0	0
0	0	0	1	1	1
0	0	1	0	2	2
0	0	1	1	3	3
0	1	0	0	4	4
0	1	0	1	5	5
0	1	1	0	6	6
0	1	1	1	7	7
1	0	0	0	8	8
1	0	0	1	9	9
1	0	1	0	A	10
1	0	1	1	B	11
1	1	0	0	C	12
1	1	0	1	D	13
1	1	1	0	E	14
1	1	1	1	F	15

at89x52.h

```

sfr at 0x80 P0 ;
sfr at 0x80 LEDS ;
sfr at 0x90 P1 ;
sfr at 0xA0 P2 ;
sfr at 0xA0 DIPS ;
sfr at 0xB0 P3 ;

sbit at 0x80 P0_0 ;
sbit at 0x81 P0_1 ;
sbit at 0x82 P0_2 ;
sbit at 0x83 P0_3 ;
sbit at 0x84 P0_4 ;
sbit at 0x85 P0_5 ;
sbit at 0x86 P0_6 ;
sbit at 0x87 P0_7 ;

sbit at 0x80 LED1 ;
sbit at 0x81 LED2 ;
sbit at 0x82 LED3 ;
sbit at 0x83 LED4 ;
sbit at 0x84 LED5 ;
sbit at 0x85 LED6 ;
sbit at 0x86 LED7 ;
sbit at 0x87 LED8 ;

sbit at 0xA0 DIP1 ;
sbit at 0xA1 DIP2 ;
sbit at 0xA2 DIP3 ;
sbit at 0xA3 DIP4 ;
sbit at 0xA4 DIP5 ;
sbit at 0xA5 DIP6 ;
sbit at 0xA6 DIP7 ;
sbit at 0xA7 DIP8 ;

sbit at 0xB2 P3_2 ;
sbit at 0xB2 PB ;
sbit at 0xB7 P3_7 ;
sbit at 0xB7 BZ ;
    
```