

V.17 – O conversor A/D

Neste tópico estudaremos o conversor de analógico para digital, mais conhecido por conversor A/D ou apenas A/D.

O objetivo do conversor A/D é converter um sinal analógico, em nosso exemplo, de 0 a 5 V, em equivalentes digitais.

Como o conversor pode ser ajustado para conversão de 8 ou 10 bits, vejamos os dois casos:

A) Conversão com resolução de 8 bits

Neste caso, com 8 bits temos $2^8 = 256$ possibilidades, o que dá aproximadamente 20 mV por conversão ($V_{ref} = 5V$). Assim sendo, poderíamos fazer a seguinte tabela:

Valor da Entrada	Valor Binário de 8 bits
0,000 V	00000000
0,020 V	00000001
0,040 V	00000010
...	...
4,980 V	11111110
5,000 V	11111111

Para usarmos o conversor A/D com resolução de 8 bits devemos colocar o resultado em uma variável de 8 bits.

B) Conversão com resolução de 10 bits

Neste caso, com 10 bits temos $2^{10} = 1.024$ possibilidades, o que dá aproximadamente 5 mV por conversão ($V_{ref} = 5V$). Assim sendo, poderíamos fazer a seguinte tabela:

Valor da Entrada	Valor Binário de 10 bits (usar variável de 16 bits)
0,000 V	0000000000
0,005 V	0000000001
0,010 V	0000000010
...	...
4,995 V	1111111110
5,000 V	1111111111

Para usarmos o conversor A/D com resolução de 10 bits devemos colocar o resultado em uma variável de 16 bits (onde os 6 bits mais significativos serão zerados).

V.17.1 – Funções para controle do conversor analógico/digital

```
Setup_Adc_Ports ( selecao_dos_pinos_analogicos );  
Setup_Adc ( modo_de_funcionamento );  
Set_Adc_channel ( qual_canal_vai_converter );  
X = read_adc ( );
```

Conforme a resolução desejada usar as diretivas: #device adc=8 ou #device adc=10 que serão estudadas no capítulo VI.