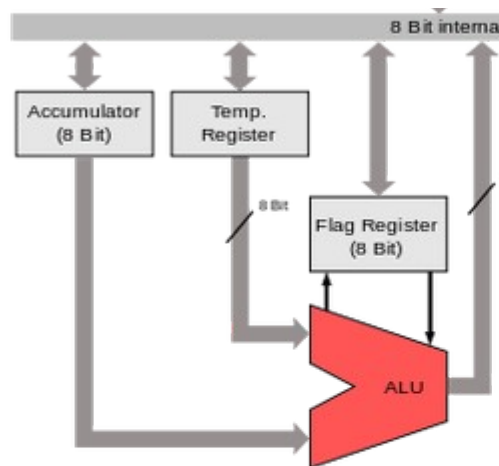


ULA

A Unidade Lógica e Aritmética (ULA, UAL, ALU) é o conjunto de circuitos no μP onde as operações aritméticas e lógicas são feitas. Para exemplificar, será mostrada a ULA do μP 8085 da Intel.

ULA do 8085

Abaixo temos o diagrama de blocos da ULA do 8085 além de mostrar o barramento interno de dados. Esta ULA pode efetuar operações aritméticas como adição e subtração e lógicas como AND, OR, NOT.

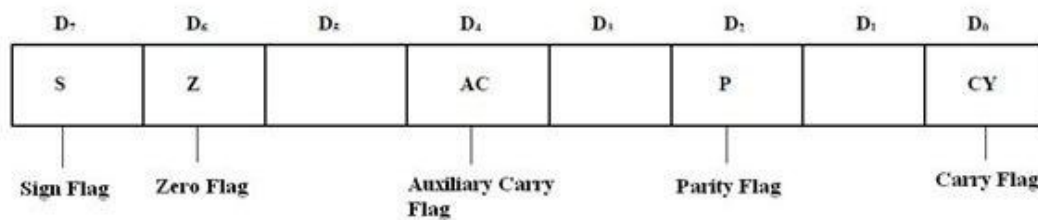


ULA do 8085 com o barramento interno

Esta ULA trabalha com 8 bits de uma vez só. Suas entradas são o Acumulador (A) e o Registrador Temporário (RT). Este último tem este nome porque o registrador funciona obtendo uma cópia dos outros registradores. Não há instruções de máquina que façam acesso a este registrador.

A saída da ULA é direcionada ao próprio A. Por isso ele tem esse nome de acumulador: é possível acumular uma sequência de operações como um somatório por exemplo.

O registrador de **Flags** contém informações importantes sobre a última operação efetuada na ULA. Este registrador tem 8 bits (cada bit vem de um FFD) e 3 deles são valores fixos (servem para que o conjunto tenha 8 bits). Os outros 5 podem ser alterados com uma operação na ULA. A figura abaixo mostra a disposição do registrador de *Flags*.



Registrador de flags do 8085. D1, D3 e D5 são fixos

Carry – C (transporte, vai-um)

Este flag é setado sempre que houver um transporte no bit mais significativo (MSB) de um resultado e resetado se não houver.

1 – Transporte ou vai-um no MSB em uma adição

0 – Não houve transporte no MSB

Parity – P (paridade)

Este flag é setado sempre que o resultado tem um número ímpar de bits 1 e resetado quando é par.

1 – O resultado tem um número par de bits 1

0 – O resultado tem um número ímpar de bits 1

Auxiliary carry – AC (carry auxiliar ou transporte auxiliar)

Este flag é setado sempre que houver um transporte do bit 3 para o bit 4 em uma adição. E resetado se não houver. Em outras palavras, se dividir a soma de 2 bytes no meio (soma de 4 bits ou nibbles), este flag indica se houve um transporte do nibble menos significativo direita para o mais significativo.

1 – Transporte do bit 3 para o bit4 do byte.

0 – Não houve transporte.

Zero – Z

O flag é setado quando o resultado da operação é zero (todos os bits do resultado em 0) e resetado quando isto não ocorrer.

1 – Resultado da operação 0

0 – Resultado da operação diferente de zero

Sign – S (sinal)

Este flag é setado quando o MSB do resultado é 1 e resetado quando o MSB for 0. Já que números negativos em binário são representados no 8085 em complemento de 2, o flag de sinal indica o sinal do resultado

1 – MSB é 1 (negativo)

0 – MSB é 0 (positivo)

Exemplo

a)

```
  111
10111010
+00011001
11010011
```

Neste exemplo temos a soma de dois números de 8 bits. Os transportes são indicados em cima da operação.

C = 0 porque não houve transporte no MSB

P = 0 pois temos 5 1s no resultado, o que dá um número ímpar

AC = 1 pois vemos que existe um transporte do bit3 para o bit4 (em negrito)

Z = 0 pois o resultado foi diferente de zero

S = 1 pois é o mesmo valor do MSB do resultado

b)

```
11111111
 11101010
+00010110
00000000
```

C = 1 porque houve transporte no MSB (destacado em negrito)

P = 1 pois não temos bits 1 no resultado, o que dá um número par de 1s

AC = 1 pois vemos que existe um transporte do bit3 para o bit4 como no exemplo a

Z = 1 pois o resultado foi zero

S = 0 pois é o mesmo valor do MSB do resultado