CISC

Examinando de um ponto de vista um pouco mais prático, a vantagem de uma arquitetura CISC é que já temos muitas das instruções guardadas no próprio processador, o que facilita o trabalho dos programadores, que já dispõe de praticamente todas as instruções que serão usadas em seus programas. Os processadores CISC têm a vantagem de reduzir o tamanho do código executável por já possuírem muito do código comum em vários programas, em forma de uma única instrução.

Os processadores baseados na computação de conjunto de instruções complexas contêm uma micro programação, ou seja, um conjunto de códigos de instruções que são gravados no processador, permitindo-lhe receber as instruções dos programas e executá-las, utilizando as instruções contidas na sua micro programação. Seria como quebrar estas instruções, já em baixo nível, em diversas instruções mais próximas do hardware (as instruções contidas no micro código do processador). Como característica marcante esta arquitetura contém um conjunto grande de instruções, a maioria deles em um elevado grau de complexidade.

A CISC é implementada e guardada em micro-código no processador, sendo difícil modificar a lógica de tratamento de instruções. Esta arquitetura suporta operações do tipo “a=a+b” descrito por “add a, b”, ou seja, podem simplesmente utilizar dois operandos para uma única instrução, sendo um deles fonte e destino (acumulador) e permite um ou mais operadores em memória para a realização das instruções. Com isto se comprova a necessidade de abranger um elevado leque de modelos de endereçamento, com acesso direto à memória e com apontadores para as variáveis em memória, armazenados eles próprios (ponteiros) em células de memória.

Porém, do ponto de vista do desempenho, os CISC’s têm algumas desvantagens em relação aos RISC’s, entre elas a impossibilidade de se alterar alguma instrução composta para se melhorar a performance. O código equivalente às instruções compostas do CISC pode ser escrito nos RISC’s da forma desejada, usando um conjunto de instruções simples, da maneira que mais se adequar. Sendo assim, existe uma disputa entre tamanho do código X desempenho.

RISC

No caso de um chip estritamente RISC, o programador já teria um pouco mais de trabalho, pois como disporia apenas de instruções simples, teria sempre que combinar várias instruções sempre que precisasse executar alguma tarefa mais complexa.

Os processadores baseados na computação de conjunto de instruções reduzido não têm micro-programação, as instruções são executadas diretamente pelo hardware. Como característica, esta arquitetura, além de não ter micro código, tem o conjunto de instruções reduzido, bem como baixo nível de complexidade.

A idéia foi inspirada pela descoberta de que muitas das características incluídas na arquitetura tradicional de processadores para ganho de desempenho foram ignoradas pelos programas que foram executados neles. Mas o desempenho do processador em relação à memória que ele acessava era crescente. Isto resultou num número de técnicas para otimização do processo dentro do processador, enquanto ao mesmo tempo tentando reduzir o número total de acessos à memória.

RISC é também a arquitetura adotada para os processadores dos videogames modernos, que proporcionam um hardware extremamente dedicado somente à execução do jogo, tornando-o muito mais rápido em relação a micro computadores com mais recursos, embora com processador x86.

Pode-se concluir que os projetistas de arquiteturas CISC consideram três aspectos básicos: – uso de micro código; – construção de conjuntos com instruções completas e eficientes (completeza no conjunto); – criação de instruções de máquina de “alto nível”, ou seja, com complexidade semelhante à dos comandos de alto nível.