Revisão para P2

1 - Explique Set de Instruções, com seus elementos.

Resposta: É a coleção de diferentes instruções que o processador pode executar. Seus elementos são:

-Op code (código de operação): Especifica a operação à ser realizado;

-Referência à operando fonte: A operação pode envolver um ou mais operandos fontes, ou seja, operandos que são entradas para operação;

-Referência à operando destino: A operação pode produzir um resultado;

-Referência à próxima instrução: Diz ao processador onde buscar a próxima instrução após o término da instrução em execução.

2 – Explique a forma que uma instrução pode ser representada, de acordo com os tipos possíveis de processador.

Resposta: No computador cada instrução é representada por uma sequência de bits. A instrução é dividida em campos correspondentes aos elementos constituintes de instrução. Os opcodes são representados por abreviações, chamadas mnemônicos, que indicam a operação. Ex.: Add: adiciona, sub: subtrai, mul: multiplica, div: divide, load: carrega e stor: armazena.

Exemplos de instruções para processadores com 3 endereços: Operando 1, operando 2, resultado: s ub y,a,b

Exemplos de instruções para processadores com 2 endereços: Neste caso, um endereço realiza trabalho duplo, como operando e como resultado: a=a+b (move y,a // add y,b)

Exemplos de instruções para processadores com 1 endereço: Este é o caso mais simples e para que funcione, um segundo endereço precisa ser implícito, sendo este endereço, um registrador do processador, conhecido como acumulador (AC). O acumulador contem um dos operandos e é usado para armazenar o resultado: Load d, mpy e, add c (ac=d, ac=ac\*e, ac=ac+c).

Existem também os casos dos processadores com zero endereços para algumas instruções, que se aplicam a uma organização de memória especial, chamada de pilha.

3 – Explique tipo de dados e qual sua relevância.

-Geral: locais de byte, word (16 bi ts), palavras duplas (32bits), quatro palavras (64 bits) e quatro palavras duplas (128bits) com conteúdo binário arbitrário.

-Números inteiros: Um valor binário com sinal, contido em um byte, palavra ou palavras duplas, usando a representação de complemento a dois. Interpretado como campo de bit ou inteiro;

-Números ordinais: Um inteiro sem sinal contido em um byte, palavra ou palavras duplas;

-Números em BCD (Binary coded decimal) (BCD) n ão agrupado: uma representação de um dígito BCD no intervalo de 0 a 9, com um dígito em cada byte;

-Agrupado BCD: Representação de byte agrupado de dois dígitos BCD; valor no intervalo de 0 a 99;

-Ponteiro near: Um endereço efetivo de 16, 32 ou 64 bits, que representa o deslocamento dentro de um segmento. Usado para todos os ponteiros em uma memória não segmentada e para referências dentro de um segmento em uma memória segmentada.

-Ponteiro far: são usados para referência à memória em um modelo de memória segmentado, onde a identidade de um segmento sendo acessado precisa ser especificada explicitamente;

-Campo de bits: Uma sequência contigua de bits, contendo de zero a 2^32 – 1 bits.

-Cadeia de bytes: Uma sequência contigua de bytes, palavras ou doublewords, contendo de zero a 2^32– 1 bytes.

-Ponto flutuante: ((Expoente)(significado) – precisão simples: 32 bits, sendo 22 bits para o significado, 8 bits para o expoente e um para o sinal.

-Precisão dupla: 64 bits, sendo 52 bits para o significado, 11 bits para o expoente e um para o sinal.

-Ponto flutuante de precisão dupla estendida: 80 bits, sendo 63 bits para o significado, 1 bit indicando bit de inteiro, 16 bits indicando o expoente e 1 bit de sinal.

-SIMD agrupada (do inglês single instruction , multiple data – única instrução, múltiplos dados): Tipos de dados de 64 a 128bits.

Os tipos de dados SIMD foram introduzidos à arquitetura x86 como parte das extensões do conjunto de instruções para otimizar o desempenho de aplicações de multimídia. O conceito básico é que múltiplos operandos são agrupados em um único item de memória referenciado e que esses múltiplos operandos são operados em paralelo. Os tipos são os seguintes:

-Byte agrupado e inteiro de byte agrupado: bytes agrupados em uma quadword de 64bits ou double 64bits ou double quadword de 128bits, interpretada como um campo de bit ou como um bit inteiro;

-Palavra agrupada e inteiro de palavra agrupada: palavras de 16bits agrupados em uma quadword de 64bits ou double quadword de 128bits, interpretada como um campo de bit ou como um inteiro;

-Doubleword agrupado e inteiro de doubleword agru pado: dou blewords de 32bits agrupados em uma quad word de 64 bits ou double quad word de 128bits, interpretada como um campo de bit ou como um inteiro

-Quadword agrupado e inteiro de quadword agrupado: duas quadwords de 64bits agrupadas em uma double quadword de 128bits, interpretada como um campo de bit ou como um inteiro;

-Ponto flutuante de precisão simples agrupado e ponto flutuante de precisão dupla agrupado: quatro valores de ponto flutuante de 32bits ou dois valores de ponto flutuante de 64bits agrupados em uma double quadword de 128bits.

4 – Explique os tipos de Operações possíveis em um computador.

-Transferência de dados: Especifica o local dos operandos de origem e destino, a extensão dos da dos e o modo de endereçamento para cada operando;

-Aritmética: A maioria das máquinas oferece operações aritméticas básicas de adição, sub tração, multiplicação e divisão. São invariavelmente fornecidas para n úmeros inteiros com sinal. Normalmente, elas também são fornecidas para números d e ponto flutuante e decimal agrupado. Outras operações possíveis incluem uma série de instruções de único operando, por exemplo: Absolute: apanha o valor absoluto do operando, negate: inverte o sinal do operando, incremente: soma 1 ao operando e decrement: subtrai 1 do operando. A execução de uma instrução aritmética pode envolver operações de transferência de dados para posicionar operandos para entrada na ALU e entregar na saída da ALU.

-Lógicas: Operações para manipular bits individuais de uma palavra ou outras unidades endereçáveis, normalmente conhecidas como “bit twiddli ng”. São baseadas em operações booleanas. Algumas das operações lógicas básicas são: NOT(inverte um b it), AND,OR e Exclusive OR (XOR) são as funções lógicas mais comuns com dois operandos. E qual é um teste binário útil;

-Conversão: Instruções de conversão são aquelas que mudam o formato ou operam sobre o formato dos dados. Ex. conversão de decimal para binário. Um exemplo de uma instrução de edição mais complexa é a instrução Translate (TR) do EAS/3 90. Essa instrução pode ser usada para converter um código de 8bits para outro e utiliza 3 operandos: TR R1(L), R2;

-Entrada e saída: As instruções de entrada e saída, possuem diversas técnicas que podem ser usadas, incluindo E/S programada, independente, E/S programada mapeada na memória, DMA e o uso de um processador de E/S. Muitas implementações oferecem a penas algumas instruções de E/S, com ações específicas ditadas por parâmetros, códigos ou palavras de comando.

-Controle de sistema: São aquelas que podem ser executadas apenas enquanto o processador está em um certo estado privilegiado ou está executando um programa em uma área privilegiada especial da memória. Normalmente, essas instruções são res ervadas para o uso do sistema operacional. Alguns exemplos de operações de controle são: Uma instrução de controle do sistema pode ler ou alterar um registrador de controle; outro exemplo é uma instrução para ler ou modificar uma chave de proteção de armazenamento, como a que é usada no sistema de memória do EAS/390. Outro exemplo é o acesso para processar blocos de controle em um sistema multiprogramado;

-Transferência de controle: Uma fração significativa da s instruções em um programa tem como função mudar a sequência de execução de instruções. Para essas instruções, a operação realizada pelo processador é atualizar o contador de programa para conter o endereço d e alguma instrução na memória.

Os mais importantes motivos das operações de transferência de controle são:

1-Loop no programa para processar uma sequência de instruções repetidamente, para que todos os dados sejam processados, ao invés de escrever cada instrução separadamente;

2-Para uma tomada de decisão;

3-Mecanismos para dividir a tarefa em pedaços menores para que possam ser trabalhadas uma de cada vez.

5 – Explique as características dos processadores CISC.

Resposta: CISC (Complex Instruction Set Computer, ou, em um a tradução literal, “Computador com um Conjunto Complexo de Instruções”): é um processador capaz de executar centenas de instruções complexas diferentes sendo, assim, extremamente versátil. Exemplos de processadores CISC são os 386 e os 486 da Intel. Os processadores baseados na computação de conjunto de instruções complexas contêm uma micro programação, ou seja, um conjunto d e 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. Esta arquitetura suporta operações do tipo “a=a+b”, podendo 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.

6 – Explique as características dos processadores RISC.

Resposta: Reduced Instruction Set Computer ou Computador com um Conjunto Reduzido de Instruções (RISC), é uma linha de arquitetura de computadores que favorece um conjunto simples e pequeno de instruções que levam aproximadamente a mesma quantidade de tempo para serem executadas. Como exemplo de aplicação dessa arquitetura são DEC Alpha, SPARC, MIPS, e PowerPC. O tipo de microprocessador mais largamente usado em desktops, o x86, é mais CISC do que RISC, embora chips mais novos traduzam instruções x86 baseadas em arquitetura CISC em formas baseadas em arquitetura RISC mais simples, utilizando prioridade de execução. Os processadores baseados na computação de conjunto de instruções reduzido não tem microprogramação, as instruções são executadas diretamente pelo hardware. Como característica, esta arquitetura, além de não ter microcódigo, tem o conjunto de instruções reduzido, bem como baixo nível de complexidade.

7 - Explique as características dos processadores Híbridos CISC- RISC.

Resposta: As características dos processadores são hibridas por sim ples questão de performance. Como na 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. Internamente, o processador processa apenas instruções simples. Estas instruções internas, variam de processador para processador, são como uma luva, que se adapta ao projeto do chip. Sobre estas instruções internas, temos um circuito decodificador, que converte as instruções complexas utilizadas pelos programas em várias instruções simples que podem ser entendidas pelo processador. Estas instruções complexas sim, são iguais em todos os processadores usados em micros PC. O conjunto básico d e instruções usadas em micros PC é chamado de conjunto x86. Este conjunto é composto por um total de 187 instruções, que são as utilizadas por todos os programas. Além deste conjunto principal, alguns processadores trazem também instruções alternativas, que permitem aos programas executar algumas tarefas mais rapidamente do que seria possível usando as instruções x86 padrão

8 – Explique o banco de registradores dos processadores RISC e, o motivo de sua adoção.

Resposta: Os registradores são organizados em janela para reduzir o acesso à memória. Cada janela é dedicada a uma chamada de procedimento.

Registradores de parâmetro: Armazenam os parâmetros passados para o procedimento;

Registradores locais: contém as variáveis locais ao procedimento;

Registradores temporários: São usados para trocar resultados / parâmetros com o procedimento seguinte.

Os bancos de registradores foram adotados para m inimizar o acesso a memória, para garantir a rapidez e eficiência do sistema e por ser o dispositivo mais rápido disponível, mais rápido que a memória principal e que a cache.

9 – Explique Registradores de uso geral e de uso específico, dizendo, pro caso dos de uso específico, quais são e para que servem.

Resposta: Os registradores de uso geral podem ser atribuídos p ara uma variedade de funções pelo programador. Algumas vezes, seu uso dentro do conjunto de instruções é ortogonal para a operação, ou seja, qualquer registrador de uso geral pode conter um operando para qualquer opcode. Permite o verdadeiro uso dos registradores de uso geral. Podem ser usados p ara dados (acumulador) ou endereços (uso geral). O que fazem os registradores para uso geral é:

-Aumento da flexibilidade e opções de programação;

-Aumento do tamanho das instruções e flexibilidade.

Registradores de uso específico: Menores (mais rápidas) instruções e menos flexibilidade.

-Dados: Contém um dado à ser lido ou escrito na memória. Deveriam ser capazes de guardar valores da maioria de tipos de dados, em algumas máquinas permitem que dois registradores contínuos sejam usados em conjunto para guardar valores de tamanho duplo.

-Endereços: Armazenam endereços da posição da memória à ser li da ou escrita. Precisam ter pelo menos o tamanho suficiente para guardar o maior endereço possível.

-Códigos condicionais: Também chamadas de flags, são bits definidos pelo hardware do processador como resultado das operações, por exemplo, uma operação aritmética pode produzir um resultado positivo, negativo (sinal), zero (zero) ou fora da capacidade (overflow). Além do resultado que é guardado no registrador ou na memória, um código condicional também é definido. O código pode ser testado na sequência como parte de uma operação de desvio condicional (branch). Os bits de código condicional são coletados em um ou mais registradores. Normalmente fazem parte do registrador de controle.

Geralmente, as instruções de máquina permitem que esses bits sejam lidos por referência implícita, mas o programador não pode alterá-los.

-Registradores de controle e estado:

Contador de programa (PC): contém o endereço de uma instrução a ser lida;

Registrador de instrução (IR): contém a instrução lida mais recentemente;

Registrador de endereço de memória (MAR): contem o endereço de uma posição de memória;

Registrador de buffer de memória (MBR): contém uma palavra de dados para ser escrita na memória ou a palavra lida mais recentemente.

PSW (program status word):

Sinal: contém o bit de sinal do resultado da última operação aritmética;

Zero: marcado quando o resultado é 0.

Carry: m arcado se uma operação resultou em transportar (adição) para empréstimo (subtração) de um bit de ordem maior. Usado para operações aritméticas de múltiplas palavras.

Igual: marcado se uma comparação lógica resultou em igualdade.

Overflow: Usado para indicar sobrecarga aritmética.

Habilitar/desabilitar interrupção: usado para habilitar ou desabilitar interrupções.

Supervisor: indica se o processador está executando no modo supervisor ou usuário

10 – Explique, utilizando as características do ciclo de execução de instrução, Pipeline.

Resposta: A pipeline é uma forma de organizar o funcionamento da CPU criada com o intuito de melhorar o desempenho da mesma. Nela, ao invés de uma instrução esperar que a instrução anterior seja executada para que então ela seja executada, a instrução é dividida em:

Buscar instrução(FI, do inglês Fetch Instruction): Ler a próxima instrução esperada em um buffer;

Decodidficar instrução (DI): determinar o opcode e os especificadores dos operandos;

Calcular operandos (CO): calcular o endereço efetivo de cada operando de origem, que pode envolver endereçamento por deslocamento, registrador indireto ou outras formas de cálculo de endereço;

Obter operandos (FO: Fetch operands): obter cada operando da memória. Operandos que se encontram nos registradores, não precisam ser lidos pela memória;

Executar instrução (EI): efetuar a operação indicada e armazenar o resultado, se houver na posição operando de destino especificada;

Escrever operando (W rite operands): Armazenar o resultado na memória.

11 – Explique os tipos de tratamento de saltos em Pipelines.

Resposta: Múltiplos fluxos: Tem dois pipelines, fazendo uso de dois fluxos. Com múltiplos pipelines, existem atrasos no acesso aos registradores e à memória. Instruções de desvio adicionais podem entrar no pipeline (ou e m qualquer dos fluxos) antes que a decisão de desvio original seja resolvida. Cada uma dessas instruções precisa de um fluxo adicional.

-Busca antecipada do alvo do desvio: Quando um desvio condicional é reconhecido, o alvo do desvio é lido antecipadamente, além da instrução que segue o desvio. Esse alvo é então salvo até que a instrução de desvio seja executada. Se o desvio for tomado, o alvo já foi obtido.

-Buffer de laço de repetição: Um buffer de laço d e repetição é uma memória pequena e extremamente rápida mantida pelo estágio pipeline de busca da instrução e que contém n instruções mais recentemente lidas na sequência. Se um desvio está para ser tomado, o hardware primeiro verifica se o alvo do desvio já está no buffer. Se estiver, a próxima instrução é obtida do buffer. O buffer de laço de repetição possui 3 benefícios:

-Com o uso de busca antecipada, o buffer conterá algumas instruções em sequência na frente do endereço da instrução atual, assim as instruções obtidas na sequência estarão disponíveis sem o tempo usual de acesso à memória.

Se um desvio para um alvo estiver apenas algumas posições à frente do endereço da instrução de desvio, o alvo já estará no buffer.

-É particularmente bem adaptada para lidar com lações ou interações.

-Previsão de desvio: Várias técnicas podem ser usadas para prever se um desvio será tomado. Entre as mais comuns estão as seguintes:

-Previsão nunca tomada: Nunca vai ter salto;

-Previsão sempre tomada: Sempre atrasa 1 tempo pipeline;

-Previsão por opcode (mais utilizada): O processador assume que o desvio será feito para determinados opcodes de desvio e não para outros. Taxas de sucesso superiores a 75% com esta estratégia.

-Chave tomada/não tomada: As estratégias dinâmicas tentam melhorar a precisão da previsão armazenando um histórico de instruções de desvios condicionais de programa . Um ou mais bits podem ser associados com cada instrução de desvio condicional que reflete o histórico recente da instrução.

Estes bits são conhecidos como uma chave tomada/não tomada que direciona o processador a tomar uma determinada decisão na próxima vez que a instrução for encontrada.

-Tabela de histórico de desvio: é uma pequena memória cache associada com o estágio de leitura da instrução do pipeline. Cada entrada da tabela consiste de três elementos: o endereço da instrução de desvio, algum número de bits de histórico que guardam o estado de uso dessa instrução e informação sobre a instrução alvo. Outra possibilidade é que o terceiro campo contenha a instrução alvo em si. A negociação é clara: armazenar o endereço do alvo necessita de uma tabela menor, porém, um

tempo maior para obter a instrução se comparado com armazenar a instrução alvo.

12 – Explique Superpipeline.

Resposta: O superpipeline explora o fato de que muitos estágios de pipeline executam tarefas que requerem menos do que a metade de um ciclo de clock. Assim, a velocidade interna de clock dobrada possibilita o desempenho de duas tarefas de um ciclo de clock externo. O controle assíncrono do processador é uma dificuldade do superpipeline.

13 – Explique Superescalar.

Resposta: Superescalar é aquele em que múltiplos e independentes pipelines de instruções são usados. Cada pipeline consiste de múltiplos estágios, de tal forma que cada pipeline possa lidar com múltiplas instruções ao mesmo tempo. Os pipelines múltiplos introduzem um nível de paralelismo, possibilitando que múltiplos fluxos de instruções sejam processados ao mesmo tempo.

14 – Explique as limitações de um sistema Superescalar.

Resposta: A abordagem superescalar depende da habilidade de executar várias instruções em paralelo.

O termo paralelismo em nível de instruções diz respeito ao nível no qual as instruções de um programa podem ser executadas de forma paralela (em média). Aumento da complexidade e problemas de dependências e desvios.

-Dependência de dados verdadeira: A segunda instrução pode ser obtida e decodificada, mas não pode ser executada até que a primeira execute. O motivo é que a segunda instrução precisa de dados produzidas pela primeira.

-Dependências procedurais: As instruções que vem depois de um desvio (tomado ou não) possuem uma dependencia procedural com o desvio e não podem ser executadas até que o desvio seja executado. Uma oportunidade de magnitude maior é perdida com cada atraso.

-Conflito de recursos: Um conflito de recursos é uma competição de duas ou mais instruções pelo mesmo recurso e ao mesmo tempo.

-Dependência de saída: dependência de escrita após escrita (W AW , write after write)

-Antidependência: dependência de ler após escrever (RAW , read after write)

15 – Explique as possibilidades para Multiprocessadores Simétricos.

Resposta: Multiprocessadores simétricos o u SMP, do inglês symmetric multiprocessor, têm vantagens potenciais sobre processadores de arquitetura uniprocessador, como desempenho, disponibilidade, crescimento incremental e personalização do fornecimento. Um SMP consiste de vários processadores semelhantes dentro de um mesmo computador, interconectados por um barramento ou algum tipo de arranjo de comutação. O problema mais crítico à ser resolvido em um SMP é a coerência de cache. Cada processador possui a sua própria cache e assim é possível que uma determinada informação esteja presente em mais de uma delas. Se tal informação for alterada em uma cache, então a memória principal e a outra cache possuem uma versão inválida dessa informação. Os protocolos de coerência de cache são projetados para lidar com esse problema. Quando mais de um processador é implementado em um chip único, a configuração é conhecida como chip de multiprocessameto. Um esquema de projeto relacionado é replicar alguns dos componentes de um único processador para que o processador possa executar várias threads de forma concorrente; isto é conhecido como um processador mutithread . Um cluster é um grupo de computadores completo conectados trabalhando juntos como um recurso computacional unificado que pode criar a ilusão de ser apenas uma máquina. O termo computador completo significa um sistema que pode funcionar por conta própria, separado do cluster. Um sistema NUMA é um multiprocessador de memória com partilhada em que o tempo de acesso para determinado processador a uma palavra na memória varia de acordo com a posição da palavra n a memória. A classificação de sua organização é: tempo compartilhado ou barramento comum, memórias com múltiplas portas e unidade de controle central.

16 – Explique a Lei de Pollack.

Resposta: O aumento de desempenho do microprocessador, devido aos avanços da microarquitetura é aproximadamente proporcional à raiz quadrada do aumento de sua complexidade (lógica do processador).

17 – Explique a organização Multicore para os Principais Processadores.

Resposta: No nível mais alto da descrição, as principais variáveis um uma organização multicore são as seguintes:

-Números de núcleos processadores no chip.

-Números de níveis da memória cache

-Quantidade de memória cache que é compartilhada

Intel Core Duo: Foi introduzido em 2006, implementa dois processadores x86 superescalares com uma cache L2 compartilhada. Como é comum em sistemas multicore cada núcleo tem a sua cache L1 dedicada. Neste caso, cada núcleo tem um a cache de instruções de 32KB e uma cache de dados de 32 KB. Cada núcleo tem uma unidade de controle térmica. Com a densidade maior de chips atuais, o gerenciamento térmico é uma capacidade fundamental, especialmente para laptops e sistemas móveis.

A unidade de controle térmico de CORE DUO é projetada para gerenciar a dissipação de calor do chip para maximizar o desempenho dentro das restrições térmicas, o que melhora também a ergonomia para o sistema de esfriamento e menor barulho acústico do ventilador. Ele monitora sensores digitais de alta precisão para m edição de temperatura de alta precisão. Cada núcleo é uma zona térmica independente.

Se a temperatura exceder o limite em um núcleo, a unidade térmica reduz a taxa de clock para diminuir a geração de calor. O próximo elemento deste processador é o controlador programável avançado de interrupção (APIC – advanced programmable interrupt controler, que desempenha uma série de funções, como o provimento de interrupções entre processadores, que permite que qualquer processador interrompa qualquer outro processador ou conjunto de processadores, aceita interrupções de E/S e encaminha-as para o núcleo adequado, cada APIC inclui um temporizador, o qual pode ser ajustado pelo SO para gerar uma interrupção no núcleo local.

A lógica de gerenciamento de energia é responsável por reduzir o consumo de energia quando possível.

Monitora as condições térmicas e atividade da CPU e ajusta os níveis de voltagem e consumo de energia de acordo. O Chip CORE DUO inclui uma cache L2 de 2MB compartilhada, onde sua lógica permite alocação dinâmica do espaço da cache com base nas necessidades atuais do núcleo de tal forma que um núcleo pode ser atribuído até 100 porcento da cache L2. A cache L2 possui lógica para suportar o protocolo MESI para caches L1 anexadas.. O principal ponto a ser considerado é quando um a escrita é feita em nível L1.

A interface de barramento conecta-se com o barramento externo, conhecido como barramento frontal, qual se conecta com a memória principal, controladores de E/S e outros chips processadores.

Intel CORE I7

Foi introduzido em novembro de 200 8, implementa 4 processador es x86 SMT, cada um com uma cache L2 dedicada e com uma cache L3 compartilhada.

Cada núcleo tem sua cache L2 dedicada e quatro núcleos compartilham a cache L3 de 8MB. Um mecanismo que a Intel usa para tornar suas caches mais eficientes é a pré -busca, onde o hardware analisa padrões de acesso a memória e tenta preencher as caches de forma especulativa com dados que provavelmente serão requisitados logo. O chip CORE I7 suporta duas formas de comunicação externa com outros chips:

O controlador de memória DDR3: traz o controlador de memória para memória principal DDR para o chip.

A interface suporta três canais com tamanho de 8 bytes para um barramento total de 192 bits, para uma taxa de dados agregada de 32GB/s. Com o controlador de memória no chip, o barramento frontal é eliminado.

O caminho de interterconexão rápida (QPI – quick-path interconnect) é uma especificação de interconexão elétrica ponto a ponto, com coerência de cache pa ra processadores e chipsets da Intel. Ele possibilita comunicação de alta velocidade entre chips de processadores conectados. A ligação QPI opera a 6,4 GT/s (transferências por segundo). A 16 bits por transferência, isso atinge até 12,8 GB/s e como ligações QPI envolvem pares bidirecionais dedicados, a largura de banda total é 25,6GB/s.