

A. P. que

Julio 2019

1º RTI no puede centrarse en solicitar si el buffer de INT esta lleno, tiene que ser solo de lectura

2º ¿RTI?

Fetch : Si ($s_i \cdot \overline{TF} \wedge RE.MI = 0$) entonces

$[SP-1] \leftarrow PC$

$[SP-1] \leftarrow RE$

$RE.MI \leftarrow 1$ (inhibir interrupciones)

$RE.S \leftarrow 1$ (superuser)

in a fetch

Si no

in a fetch

3º ¿Síndrome de bus de control P CPU con DMA notificar

DMA {
- BR (bus Request) : solicita los datos de cada memoria
- MHRFG (Memory Request)
- READ y WRITE

c) CPU \rightarrow 64 bits

2000 MIPS

SRI = 5 ms

$V_T = 100 \cdot 10^6$ B/s

$T_{max} = 5$ ms

\circ 2 Reg 64 bit

\circ sector 512 B

\circ RI, RF (50I y 200I)

a) c) Máximo nº de instrucciones de la rutina trat de Int?

$$2000 \text{ MIPS} = \frac{1}{2000 \cdot 10^6} = 0,5 \times 10^{-10} = \frac{n}{\mu} = \frac{10^{-9}}{10^{-8}} = 0,5 \text{ ns}$$

① $T_{int} = \frac{\text{Tamaño de Buffet Int}}{2 \text{ Reg 64 bits}} = \frac{128}{100 \cdot 10^6 \text{ B/s} \cdot 8 \text{ byte/B}} = 1,6 \cdot 10^{-7} = 1,6 \text{ ns} \cdot 10^{-1}$

Velocidad a la que se trae

② $SRI + RTI = T_{int}$

$$5 \text{ ms} + x = 160 \text{ ms}$$

$$x = 160 \text{ ms} \xrightarrow{aI} \frac{160}{0,5} = 320 I$$

b) c) Tiempo CPU sector si RTI son 60I?

$$T_{int} = (60 \times 0,5) + 5 \text{ ms} = 35 \text{ ms}$$

$$T_{cpu} = T_{inicio} + T_{int} \times N_{int} + T_{gim}$$

$$= (60 \times 0,5) + 35 \text{ ms} \times N_{INT} + (35 \text{ ms} + 200 \times 0,5)$$

$$= 1280 \text{ ns}$$

$$N_{INT} = \frac{\text{Tam Sector}}{\text{Tam buffer Int}} = \frac{512 \text{ byte}}{2 \times 64 \text{ bit}} = 16 = 32$$

5

Dicir cuantos : $V_f = 10^9 \text{ bit/s}$
 $\text{Pkg bytes} = 64 \text{ bits}$

a) ¿Prioridades?

$$T_{int} = \frac{64 \text{ bits}}{10^9 \text{ bit/s}} = 6,4 \times 10^{-9} = 64 \text{ ns}$$

64 < 160
Eth
Dicir

menos tiempo, por tanto mas prioridad

$$t_{tray} = 100, 150 \quad t_{ac} = 2 \text{ ms} \quad Concesión = 4 \text{ ms}$$

b) % CPU durante la transmisión de 1024 B a Ethernet?

DMA

$$\overline{T_{tray}} = T_{Robo} = \text{Concesión} + \text{ciclo de acceso}$$

$$= 6 \text{ ms}$$

$$T_{operación} = T_{inicio} + T_{tray} + T_{ultimo \ robo \ ciclo} + (T_{fin} + T_{SRI}) + (T_{RTI})$$

$$= (150 \times 0,5) + \frac{1024 \text{ B}}{V_f (10^9 \text{ bit/s})} + 6 \text{ ms} + ((100 \times 0,5) + 5)$$

$$= 8328 \text{ ms}$$

cuenta los de fin con RTI

$$T_{cpu} = T_{inicio} + N_{Robos \ ciclo} \times T_{Robo \ ciclo} + T_{int}$$

$$(150 \times 0,5) + \left(\frac{1024}{8} \right) \times 6 \text{ ms} + (5 + 60) = 898 \text{ ms}$$

$$\% \text{ CPU} = \frac{T_{cpu}}{T_{op}} \times 100 = \frac{898}{8328} \times 100 = 10,78\%$$

C) 5 120 B
 10 24 B

$$T_{Op_{diode}} = \frac{\text{Initial + Precess} + \text{Tracing} + \text{alline hit}}{(50 \times 0.5) + 5ms + \frac{512}{100 \cdot 10^6}} + 3 + 30 + 80$$

$$= 50061280 \text{ ns}$$

$$\frac{5120}{512} = 10 \text{ Vmax}$$

$$T_{TotalOp} = 10 \times T_{Op_{diode}} + 1 T_{Op_{ETH}}$$

$$= 60.061128 \text{ ns}$$

$$T_{Total CPU} = 10 \times T_{Op_{CPU}} + 5 T_{Op_{ETH}} = 16.910 \text{ ns}$$

$$\% \text{ CPU time} = \frac{T_{Total} - T_{Op_{CPU}}}{T_{Total}} \times 100 = \frac{60.061.128 - 16.910}{60.061.128} \times 100 = 91,91\%$$

Julio 2018

1) Buffering: almacena los datos que transmiten entre los programas
mientras avanza para tratar las interrupciones cuando este se plantea

2)

CPU 32 bits $V_F = 40 \times 10^6 \text{ bytes/s}$ Reg. datos, 32 bits

2000 MIPS $= 0,5 \text{ ms}^{-1}$ S. datos = 2048 B IN y FIN = 100 I y 150 I

SRI = 4 I Tacc = 4 ms RTI = 56 I

a) Cuantos discos pueden operar simultáneamente?

①

$$F_{\text{acc de Int}} = \frac{V_F (40 \times 10^6 \text{ B/s})}{\text{Reg. datos (32 bits)}} = \frac{4 \times 10^6}{4 \text{ B}} = 10^7 \text{ Hz}$$

② Cada Int supone ejecutar $SRI + RTI = 60 I \rightarrow 30 \text{ ms}$

$$\begin{aligned} \text{③} \text{ por tanto Cips que necesitar} &= F_{\text{acc Int}} \times \text{Instancia Int} \\ &= 10^7 \text{ Hz} \times 60 \text{ ms} \\ &= 600 \text{ MIPS} \end{aligned}$$

$$2000 / 600 = 3,33 = 3 \text{ simultaneos}$$

b) T_{op} y T_{cpu} para un vector

$$\begin{aligned} \text{① } T_{op} &= IN (100I)_{\text{rest}} + \frac{T_{\text{transf}}}{40 \times 10^6} + T_{\text{acc}} + FIN + RTI + SRI \\ &= 4.051.355 \text{ ns} \end{aligned}$$

$$\begin{aligned} \text{② } T_{cpu} &= (100I \times 0,5) + \frac{2048 \text{ B}}{(32 \times 32 \text{ B})} \times (2+28) + (160 \times 0,5) = \\ &= 50 + 512 \times 30 + 75 = 15485 \end{aligned}$$

Eth (DMA)

Permutación = 2 ms

$$V_T = 10^7 \text{ bits/s} \quad \text{Acc} = 2 \text{ ms}$$

4 Regs 32 bits

IN (100) FIN (100)

c) Cuantos discos pueden trabajar con un Eth?

$$\text{Frec Int} = \frac{V_T}{4 \times 32 \text{ bit}} = \frac{10^7}{128} = 7812500 \text{ Hz}$$

Ojo

Los bloques de datos duran 2 ms y se repiten en grupos de 4

$$\boxed{\frac{N^{\circ} \text{ Registros} \times \frac{T_{\text{acc}}}{T_{\text{DMA}}}}{4} + T_{\text{liberación}}} = 10 \text{ ms} / 0,5 = 20 \text{ I}$$

$$7812500 \times 20 \text{ I} = 156250000 = 156,25 \text{ MIPS}$$

$$2000 - 156,25 = 1843,75$$

$$\frac{1843,75}{600 \text{ MIPS}} = \boxed{3} \text{ Discos}$$

d) Top Eth bloque 1024 bit

$$\begin{aligned} \text{Top} &= (150 \times 0,5) + \frac{10240}{10^7 \text{ bits}} + (4 \times 2 + 2) + ((100 \times 0,5) + (45 \times 0,5)) \\ &= 8329 \text{ ms} \end{aligned}$$

e) Instante de fin fichero 10.240 B - Eth bloque 1024

$$10.240 / 2048 = 5 \text{ Discos}$$

$$10.240 \times 1024 = 10 \text{ Ethernet}$$

$T_{disco} > T_{cpu}$ lo solucionaremos en el último bloque

$$2048/1024 = \boxed{2}$$

$$T_{op} = 5 \times T_{operador} + 2 T_{operador} = 20.273.433 \text{ ns}$$

$$T_{cpu} = 5 T_{cpu \text{ disco}} + 10 T_{cpu \text{ Eth}} = 5 \times 15485 + 10 \times 767 = \\ = 85095 \text{ ns}$$

$$\rightarrow (150 \times 0,5) + \frac{1024B}{(4 \times 32)/8} \times 10 + (1041 \times 0,5) = 767 \text{ ns}$$

↑
1000 ns = 1 μs

$$\% CPU_{libre} = \frac{20.273.433 \text{ ns} - 85095 \text{ ns}}{20.273.433} \times 100 = 99,5\%$$

Junio 2019

$$1000 \text{ MIPS} = 1^9 \quad V_t = 10^9 \text{ bits}$$

32 bits $= 1 \text{ ms}$ Eth (INT)

$$SRI = 6 \text{ ms} = 6I$$

Byte $\times 4 = 1536 \text{ Bytes}$

1 Ray 32 bits

$$Int = 100I$$

$$RTI = 24I$$

hay 1 proceso I previa a
Transmisión

$$Fin = 200I$$

BTI

Disk

$$V_r = 10^8 \text{ B/S}$$

$$Tacc = 4 \text{ ms}$$

$$Ttransf = 512B$$

$$2 \text{ Ray 32 bits}$$

$$Int \text{ y Fin } 80I \text{ y } 400I$$

DMA Concesión de bus 2 ms

$$Cacc = Tacc = 1 \text{ ms}$$

a) Modulos Eth simultaneos

$$\text{Frec Int} = \frac{10^9}{32} = 31250000 \text{ bits}$$

$$I_{int} = 6I + 24I = 30I$$

$$CP_{eth} = 30 \times 31250000$$

$$= 937,5 \text{ MIPS}$$

$$1000 / 937,5 = \boxed{1}$$

b) Discos máximos con Eth simultaneos

$$\text{Frec}_{Int} = \frac{10^8 \text{ B}}{64 \text{ bit}} = 12500000 \text{ Hz}$$

"
8B

$$I_{int} = Ciclos de \text{Vray} + \text{Concesión de buses}$$

$$= 1 \text{ ms} \times 2 + 2 \text{ ms} = 4 \text{ ms}$$

$$CP_{Disk} = 4 \text{ ms} \times 1250000 \text{ Hz} = 50 \text{ MIPS}$$

$$937 + 50 = 987$$

c) 1,5K 6th Top y Cpu?

$$1,5 \text{ MB} = 1500 \cdot$$

$$\begin{aligned} T_{op} &= 100 \text{ ns} + \cancel{1} + 30 \text{ ns} + \frac{1536 \times 8 \text{ bit}}{10^9 \text{ bit/s}} + (200 + 4 \text{ ms}) \\ &= 384 \text{ ns} 12622 \text{ ms} \end{aligned}$$

"
1 bit/ms

$$T_{cpu} = 100 \text{ ns} + 30 \text{ ns} \times \left(\frac{1536 \times 8}{32} \right) + 284 = 11824 \text{ ns}$$

$$\% CPU = \frac{12622 - 11824}{12622} \times 100 = 6,3\%$$

d) Disco duro

$$\begin{aligned} T_{op} &= 80 \text{ ns} + 4 \text{ ms} + \frac{512 \text{ B}}{18^8 \text{ B/s}} + \cancel{406 \text{ ms}} \\ &\quad + 4 \text{ ms} (400 + 6 \text{ ms}) \end{aligned}$$

$$= 80 + 4 + 5120 + 4 \text{ ms} + 406 \text{ ms}$$

$$= 5614 \text{ ms}$$

$$\begin{aligned} T_{cpu} &= 80 + 4 \text{ ms} \times \frac{512 \times 8}{2 \times 32} + 406 \\ &= 742 \text{ ms} \end{aligned}$$

c) $D_1 = 20\%$

$D_2 = 50\%$

$$\text{Max op. discs} = \frac{V_t}{T_{op} - HD} = \frac{10^*}{4005610} = 249,65 \text{ op.-HD/s}$$

$$249,65 \times 0,2 = 50 \text{ op} \\ \times 0,5 = 125 \text{ op} = 175 \text{ ope}$$

F) 15KB

15KB fichero

$$15KB / 1,5KB = 10$$

$$15 / 0,5 = 30$$

$$T_{rotar} = 30 \times T_{op discs} + T_{op ETH}$$

$$T_{cpu} = 30 T_{op discs} + 10 T_{op ETH}$$

Marzo 2018

muestreo 82 bits

recuadros int = 44.100 Hz

suelen 6 muestras 6 Bytes

$$V_{\text{real}} = 441000 \text{ Hz} \times 4 \text{ B/muestra} \times (24+7)/2^4 = 242.55 \text{ DB/S}$$

D₁

bus datos 8 bit , dir 16 bits 20 MIPS

$$SRI = 8 I$$

Buffer 33 B

$$\text{Ini } 30 I \quad RTT = 264 I$$

fin 20 I Tras 33 Bytes del buffer

a)

c acceso a buses durante SRI ?

SRI → guardamos en pila el PC y el SR, el V_{int}
16 Bit 8 Bit 16 Bit

$$2 + 1 + 2 = 5 \text{ accesos}$$

b) T_{acc} = 1 Seg

T_{transf} = 30 Seg c T_{op}? 20 MIPS $\Rightarrow 50 \text{ ms}$

$$T_{op} = 30I \cdot 50 \text{ ns} + 1 \cdot 10^9 \text{ ns} + 30 \cdot 10^9 \text{ ns} + 259 \times 50 + 8 \times 50 + 20 \times 50 \\ = 316000,61 \cdot 10^{10} \text{ ns} = 315$$

a)

DMA

Rafeges 33 Bytes

RTI = 80 I

Transf = 40 ns

Protocol y lib = 40 ns

Int = 40 I

b)

$$F_{int} = \frac{Vt \cdot 242.550 \text{ B/S}}{33 \text{ Bytes}} \\ = 7350 \text{ Int/S}$$

$$CP_{DMA} = 7350 \text{ INT} \times 272T_{int} = 2 \text{ MIPS}$$

$$T_{DMA} = \cancel{\text{Protocolos} \times \text{Transfers}} +$$

$$T_{DMA} = \frac{\text{Part}_{cyc}}{40 \text{ ns}} + \frac{8 \text{ Rafiges}}{33 \text{ B}} \times \frac{T_{acc}}{40 \text{ ns}}$$

Cap disp

① $\text{Him} + \text{rafiges}$

$$T_{DMA} = 40 \text{ ns} + (33 \times 40 \text{ ns}) \\ = 1360 \text{ ns} \\ = 27,2 \text{ I}$$

$$CP_{DMA} = 27,2 \text{ I} \times 7350 \\ = 199920 \\ = 0,19992 \text{ MIPS}$$

c)

Rd = 1 Byte

RTI = 12 F

Fm 33 Byte = 50 I

$$T_{int} = (8 + 50) * 7350$$

$$(12 + 8) \times (242.550 - 7350) \\ = 5,93 \text{ MIPS}$$

$$CP = 14,87 \text{ MIPS}$$

- 1.- Ciclos de bus, SRI con vectorización al comienzo de RTI obtenido con dir. relativo a reg. base de la tabla de vectores. (Enero 2014)
- Para salvaguardar Reg. Estado y PC : 2
 - Leer tabla y obtener el específico: 2.
- 2.- Desde el punto de vista de la duración total, cuantifique el impacto de las técnicas para la realización de las op. E/S.
- Afecta más a la CPU que al periférico. (Enero 2014).
- 3.- Sobrecoste durante 1 op de E/S gestionada por int. en lugar de programada. (Junio 2013)
- SRI: Permite saltar a RTI salvando el estado del programa inter. (PC, RE, BHI).
- Salvar Registros: Salvar del estado de ejecución del prog. int. modificado por RTI.
- Pasa de parámetros: Lect. y Esc. de las var globales que contienen los param. de la op E/S.
- RTI: Recuperación del estado modificado por la SRI para reanudar la ejec. del prog. int.
- En muchos casos, el sobrecoste es muy inferior al beneficio obtenido al ceder al módulo de E/S la labor de sincronización.
- 4.- Técnicas de identificación del módulo E/S que solicita una int. (Junio 2013)
- Por muestreo: Una RTI lee sucesivamente los RE de los módulos para identificar cual de ellos solicita la int.
- Por vectorización: Durante SRI, CPU → inicia ciclo de reconocimiento de int para solicitar al módulo que coloque en el bus de datos su identificación.
- 5.- Enumere las señales del bus de control y con qué finalidad debe gobernar un módulo E/S para hacer ~~un~~ DMA. (MARZO 2013).
- BR: Solicitud rebos de ciclo y accede dir. a mem.
- HmRO: Hace la temporización de los ciclos de acc. a mem.
- Read & Write: Indica si el acceso a mem. es para lect. o esc.

6. Criterios para asignar prioridades. (Marzo 2013).

Explicar int no enmascrables

Mayor prioridad al disp. que pide int con mayor frecuencia. Esta asignación es óptima, algunos disp. no siguen esta regla

Int. no enmascrables: Notifican sucesos que no admiran demora y suponen fallos de funcionamiento (falto energía).

(Octubre 2012).

10.- Módulos E/S en un elem Hw que facilita que cualquier pfif. pueda ser

- Manejado por cualquier CPU: ✓

• Tiempo de respuesta a los sucesos es menor

- También buffer de los módulos no agota el tiempo de CPU

• La E/S prog. Solo necesita CPU que tenga instrucciones para escritura de los reg. de los mod. E/S: ✓

- Trop. de E/S depende sustancialmente de la técnica empleada: ✓

• Del paralelo, punto a punto, serial, multiplex

7.- (OCTUBRE 2013).

Robos de ciclo → CPU se limita a ceder los buses. Como no se altera el estado interno estas cesiones se hacen casi inmediatamente.