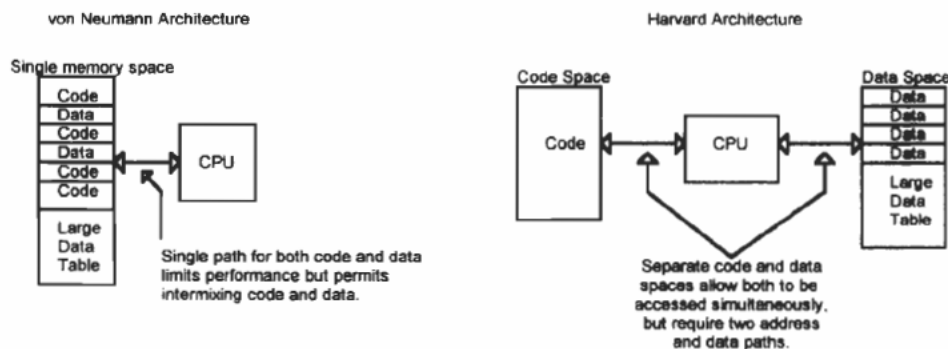


3.1 Introducción

La CPU es el núcleo de los $\mu P/\mu C$, en el caso de los μP la CPU se le suele considerar como un sinónimo.

La CPU accede a dos tipos de información a través del bus de datos: Datos y Programa. Existen dos arquitecturas distintas Von Neumann / Harvard dependiendo de si los datos y el programa comparten espacios. La arquitectura Von Neumann es la habitual en μP , mientras que la Harvard (modificada) se usa en algunos μC .



Una vez que arranca, la CPU ejecuta de manera secuencial el ciclo de captura y ejecución de instrucciones.

3.2 Ciclos de bus. Ciclo de Reset

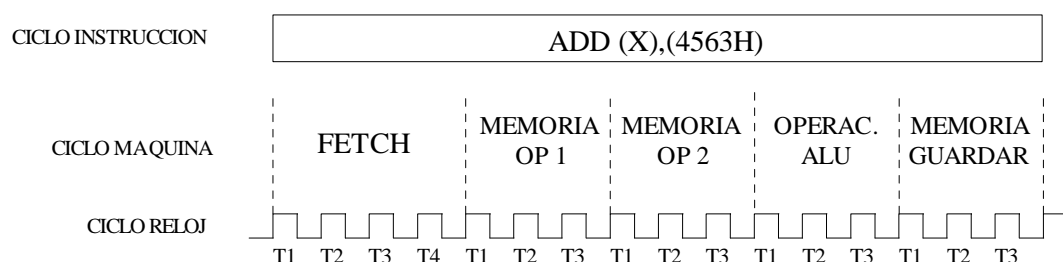
La operación temporal de la CPU se puede dividir en segmentos temporales con distinto nivel de detalle.

Ciclo Instrucción: Cada una de las instrucciones ejecutadas

Ciclo Máquina: Cada una de las operaciones que se realizan en una instrucción.

Ciclo Reloj: Cada uno de los ciclos de funcionamiento síncrono.

(No del todo porque puede haber sincronización en ambos flancos).



Ciclo de Reset

Todo sistema síncrono ha de arrancar en un sistema conocido para que su comportamiento pueda ser previsible. El arranque presupone normalmente el borrado de todos los registros y la ejecución de instrucciones desde una determinada posición de memoria o desde un lugar apuntado por un puntero que se halla en una posición prefijada.

Tipos de Reset:

a) Reset bajo demanda

Suele existir una línea RESET* que es monitorizado por la CPU, por ejemplo, en cada flanco de bajada de reloj. Cuando se detecta que está activa, se genera un ciclo interno de reset que pone en estado conocido:

- a) Algunos registros
- b) Líneas de I/O (normalmente entrada)

b) Power-On Reset

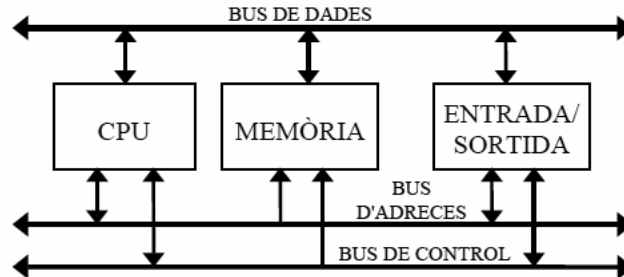
Reset generado automáticamente cuando arranca la alimentación.

Se suele usar un circuito RC para ello.

Si el $\mu P/\mu C$ no lo tiene internamente, se le ha de facilitar actuando sobre la línea de reset.

3.3 Organización de los buses

Los μP y no todos los μC tienen buses de datos, de direcciones y de control.



El **bus de datos** es un bus bidireccional XT XR que permite que los bytes de código y datos se muevan desde o hacia la CPU.

Tanto la CPU como el resto de subsistemas deben poder poner el estado Z en las líneas de este bus.

El bus de datos suele tener un ancho que acomoda bytes enteros. Son típicos los anchos de 8 bits, 16 bits y 32 bits.

El **bus de direcciones** es un bus, a priori unidireccional, 1T XR que permite a la CPU indicar al subsistema de memoria a que datos o código se desea acceder tanto para lectura como para escritura. Bajo ese punto de vista no es necesario que sus líneas se puedan hallar en estado Z.

Existe, habitualmente, la posibilidad de DMA. Un periférico puede solicitar el uso de los dos buses (datos y direcciones) a la CPU para hacer transferencias de datos más rápidas de lo que puede hacer a través de la CPU. Si la CPU lo permite, entonces pone el bus en estado Z mientras dure la cesión de los buses al periférico. Si existe la posibilidad de DMA entonces el bus se ha de considerar como XT XR igual que el caso del bus de datos.

El ancho del bus de direcciones ha de poder acomodar todo el espacio direccionable.

Si hay N líneas de direcciones (desde A0 hasta A(N-1)), la memoria direccionable es:

$$\text{Direcciones} = 2^N, \text{ esto es, desde } 0 \text{ hasta } 2^{N-1}$$

Las direcciones que maneja el bus de datos son, a priori, bytes (8 bits) con independencia de la anchura del bus de datos.

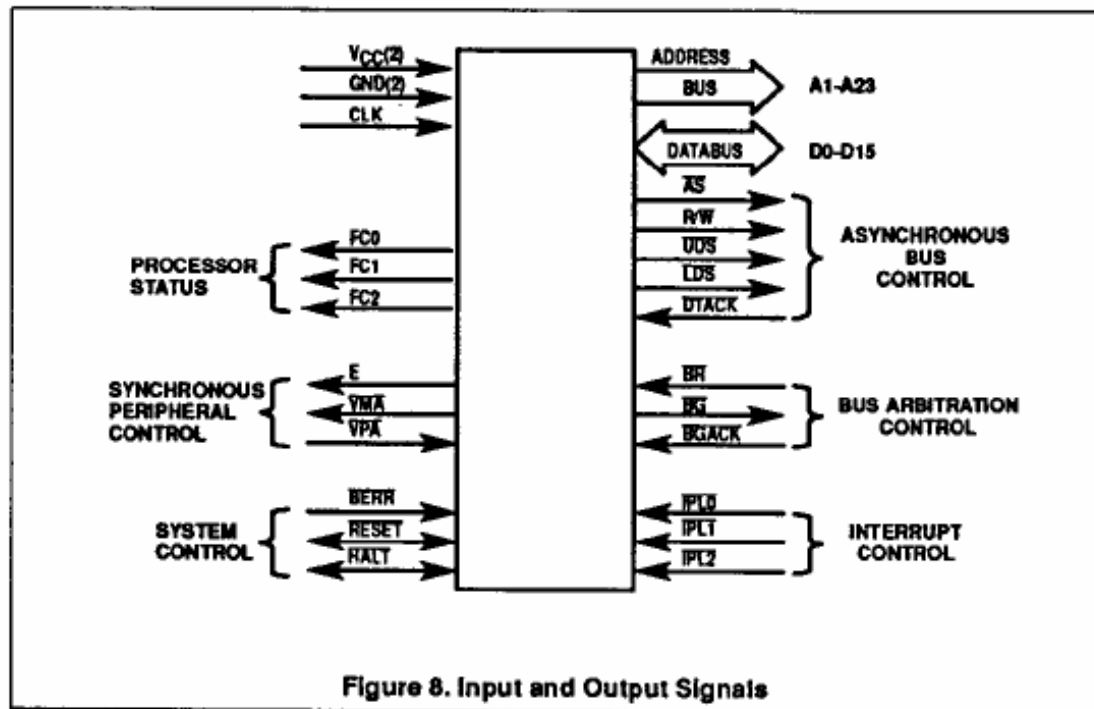
El **Bus de Control** es, por definición, el conjunto de señales de la CPU que no corresponden a los otros dos buses. Contiene señales que pueden ser unidireccionales o bidireccionales y que se agrupan en distintos conjuntos como por ejemplo:

- Información del estado del procesador.
- Control del $\mu P/\mu C$ (RESET* y HALT* por ejemplo).
- Gestión de interrupciones.
- Gestión de la cesión de buses.
- Control del acceso a la memoria (R/W*, Strobes...)

Dependiendo del procesador puede haber más grupos de líneas adicionales.

En el caso de los μC , las líneas de salida de los periféricos, pese a no pertenecer a los buses de datos o direcciones no son del bus de control al no salir de la CPU.

Ejemplo de buses en el μP 68000 de Motorola



Señales del μP 68000

Buses de datos de más de 8 bits

Si el bus de datos es de más de 8 bits, las líneas más bajas del bus de direcciones no son accesibles desde el exterior ya que no son necesarias. (Véase 68000).

El 68000 es internamente de 16/32 bits pero su bus de datos es de 16 bits. Dado que las direcciones del bus de direcciones son siempre referidas a bytes, todas las posiciones pares irán por las líneas D0 a D7 y las impares desde D8 a D15. No tiene sentido, por tanto, sacar al exterior la línea A0 del bus de direcciones.

El número de líneas del bus de direcciones es 24, eso corresponde a 3 bytes o 6 dígitos hexadecimales (nibbles). Direcciones desde 0H a FFFFFFFH. De ellas únicamente 23 líneas A1 a A23 salen al exterior.

| Posición | Valor |
|----------|-------|
| 086643H | 42H |
| 086644H | 33H |
| 086645H | 28H |
| 086646H | 56H |

Ejemplo de 4 posiciones de de memoria

Para las operaciones de lectura podríamos leer siempre los datos de 2bytes en 2 bytes.
Para leer el word (16 bits) 086644H pondríamos 043322H en (A23-A1) y leeríamos los 16 bits (D15-D0).

Para leer el byte 086644H pondríamos 043322H en (A23-A1) y leeríamos los bits de datos D7 a D0.

Para leer el byte 086645H pondríamos 043322H en (A23-A1) y leeríamos los bits de datos D15 a D8.

En la escritura de words no hay problema.

El problema es cuando escribimos bytes. Si deseamos escribir sobre un byte particular, no podemos modificar el que le acompaña como parte de la misma palabra.

Si deseamos escribir 4FH en la posición 086644H pondríamos poner 043322H en (A23-A1) y 4FH en (D7-D0), pero ¿que pondríamos en (D15-D8)?

Es por ello que necesitamos strobes de bits:

UDS* : Upper Data Strobe : Activo si accedemos a D15-D8

LDS* : Lower Data Strobe: Activo si accedemos a D7-D0

Ejemplo de operaciones:

| Acción | A23-A1 | D15-D8 | D7-D0 | R/W* | UDS* | LDS* |
|------------------------|---------|--------|-------|------|------|------|
| NOP | XXXXXX | XX | XX | X | H | H |
| MOVE.W (086644H),D0 | 043322H | 28H | 33H | H | L | L |
| MOVE.B (086644H),D1 | 043322H | XX | 33H | H | H | L |
| MOVE.B #\$21,(086645H) | 043322H | 21H | XX | L | L | H |

Cuando la CPU tiene un bus de 32 bits (4 bytes), sólo sale al exterior el bus de direcciones desde A2 en adelante. A0 y A1 codifican los 4 strobes de bytes:

| A1 | A0 | Data | Strobes |
|----|----|---------|---------|
| 0 | 0 | D7-D0 | BE0* |
| 0 | 1 | D15-D8 | BE1* |
| 1 | 0 | D23-D16 | BE2* |
| 1 | 1 | D31-D24 | BE3* |

Lectura de byte

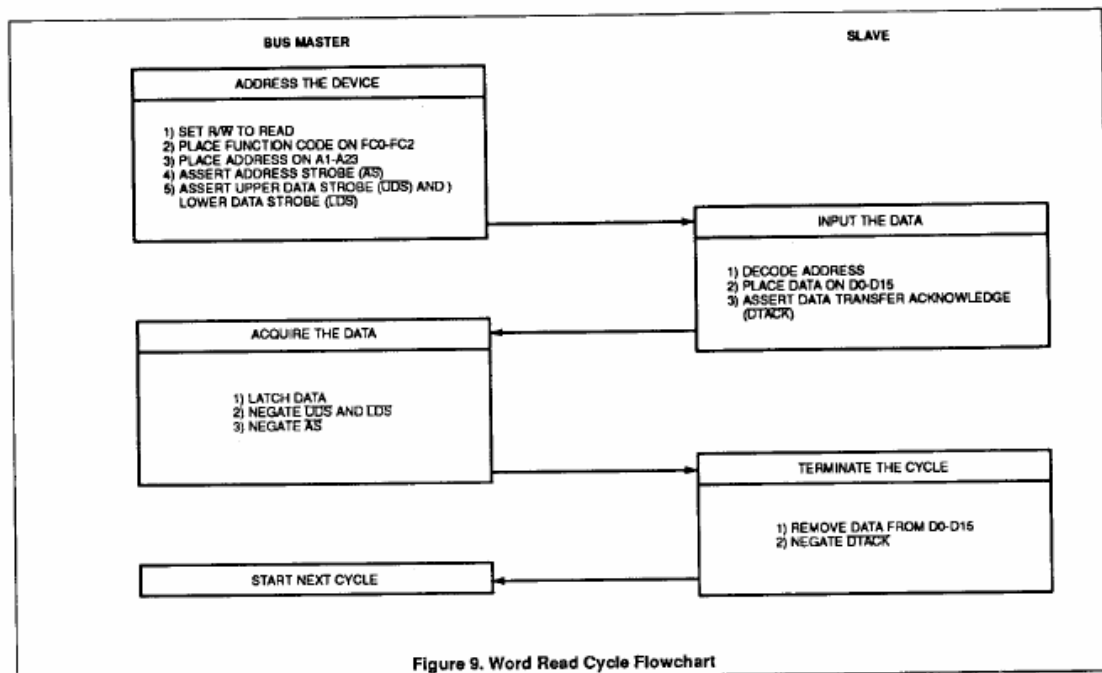
| A1 | A0 | Data | Strobes |
|----|----|---------|------------|
| 0 | 0 | D15-D0 | BE0*, BE1* |
| 1 | 0 | D31-D16 | BE2*, BE3* |

Lectura de Word

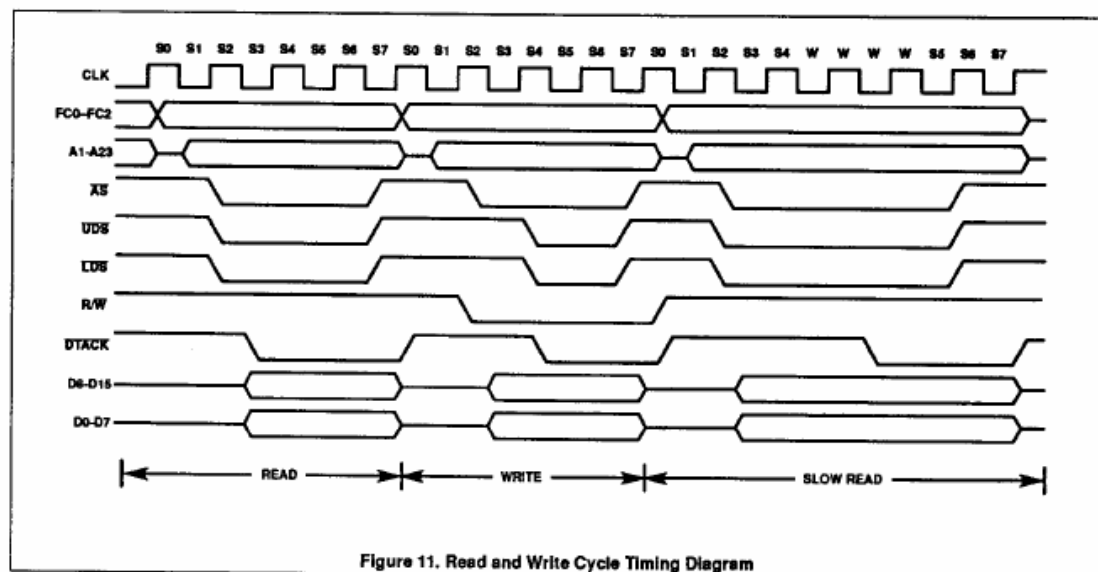
Dependiendo de la CPU acceder a Words con direcciones impares implica un error o la generación de dos ciclos de bus. En todo caso los compiladores intentan poner los Words en posiciones pares en el espacio de datos.

Ciclo de Lectura (68000)

El ciclo de lectura indica la secuencia de acciones a realizar para leer la memoria.



Secuencia de acciones en lectura

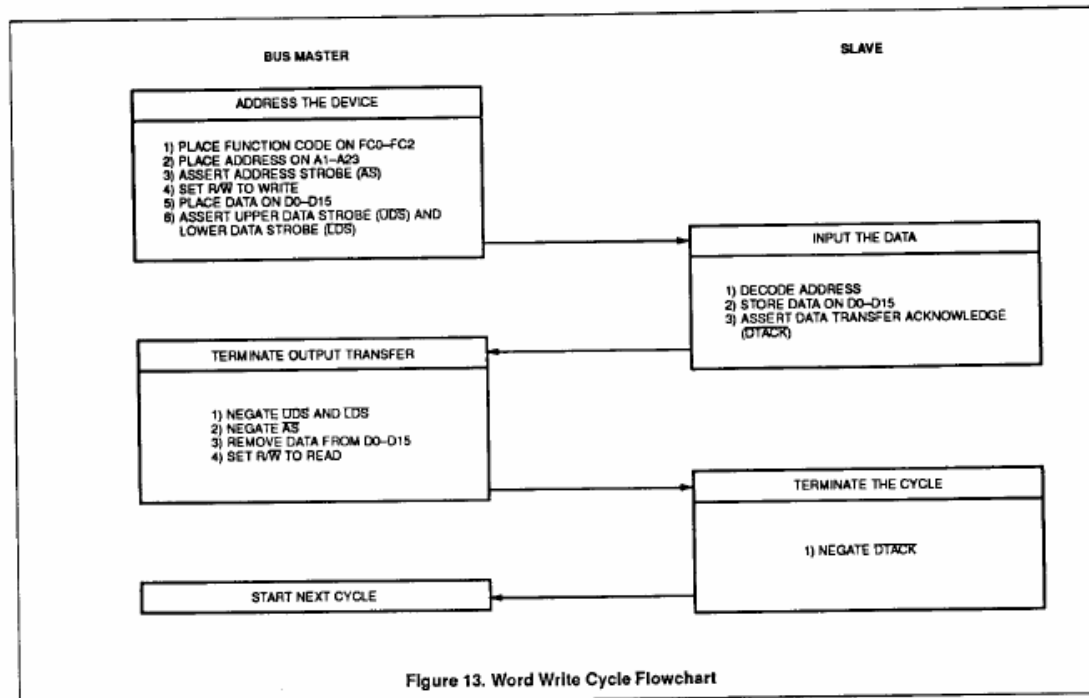


Cronograma de lectura y escritura

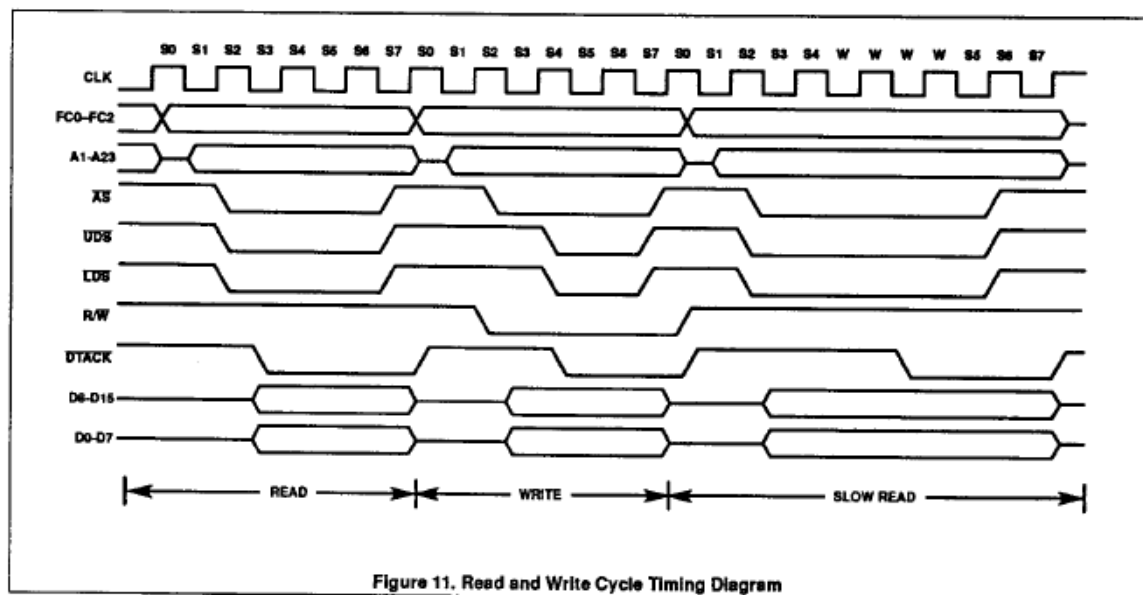
En el caso de los buses A1-A23, D8-D15 y D0-D7 una línea identifica el estado Z

Ciclo de Escritura (68000)

El ciclo de escritura indica la secuencia de acciones a realizar para escribir en la memoria.



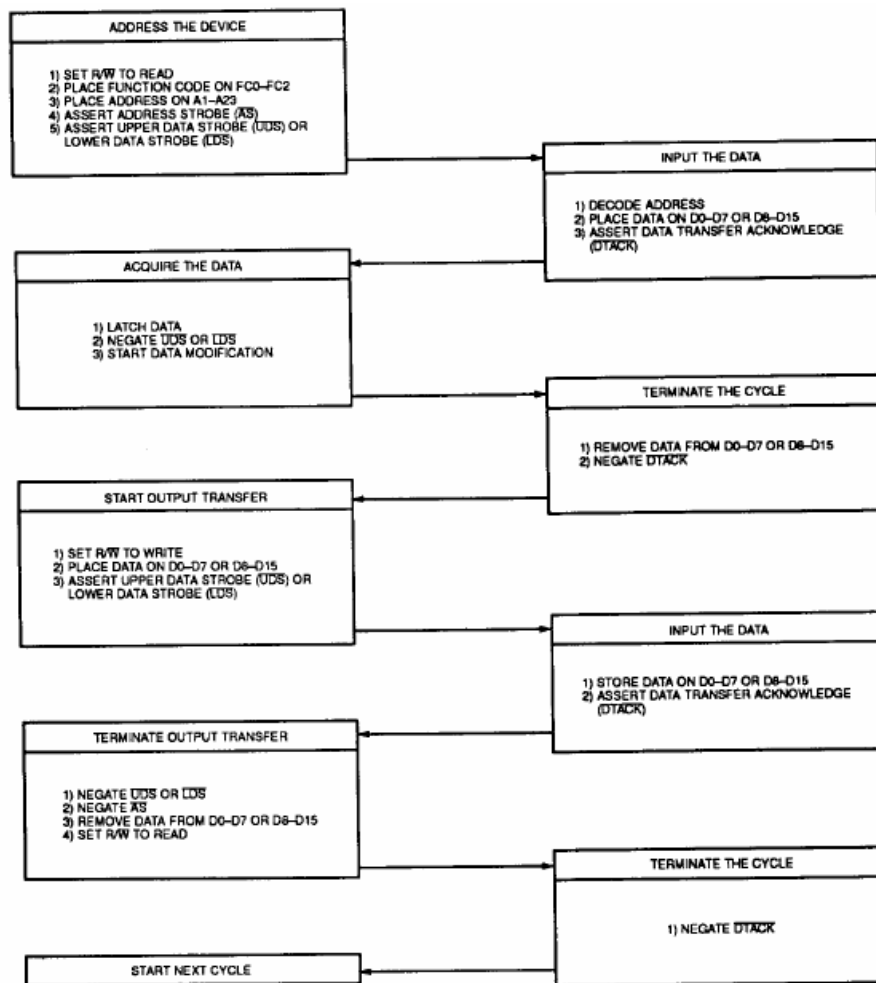
Secuencia de acciones en escritura



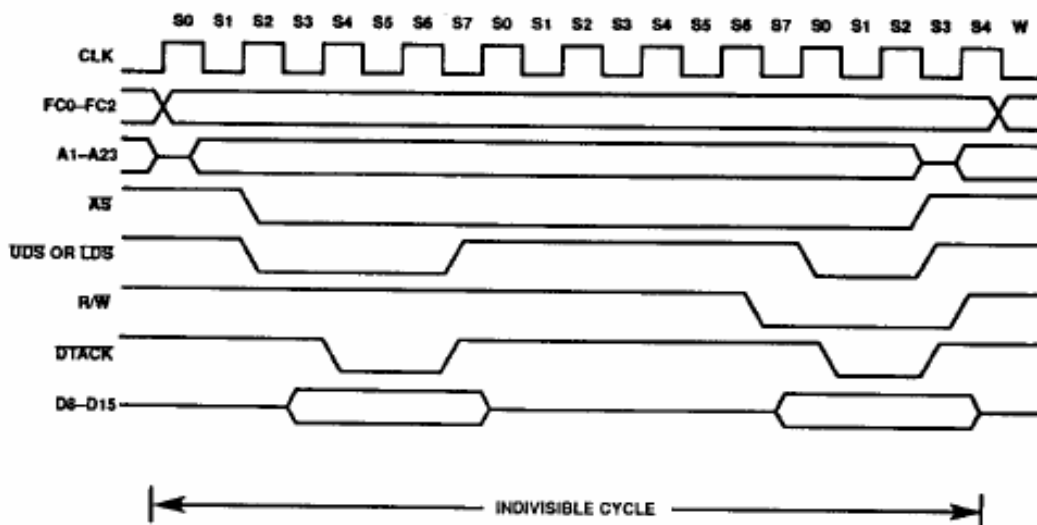
Cronograma de lectura y escritura

Ciclo combinado (Lectura-Modificación-Escritura) (68000)

Ciclo indivisible ideal para trabajar con semáforos.

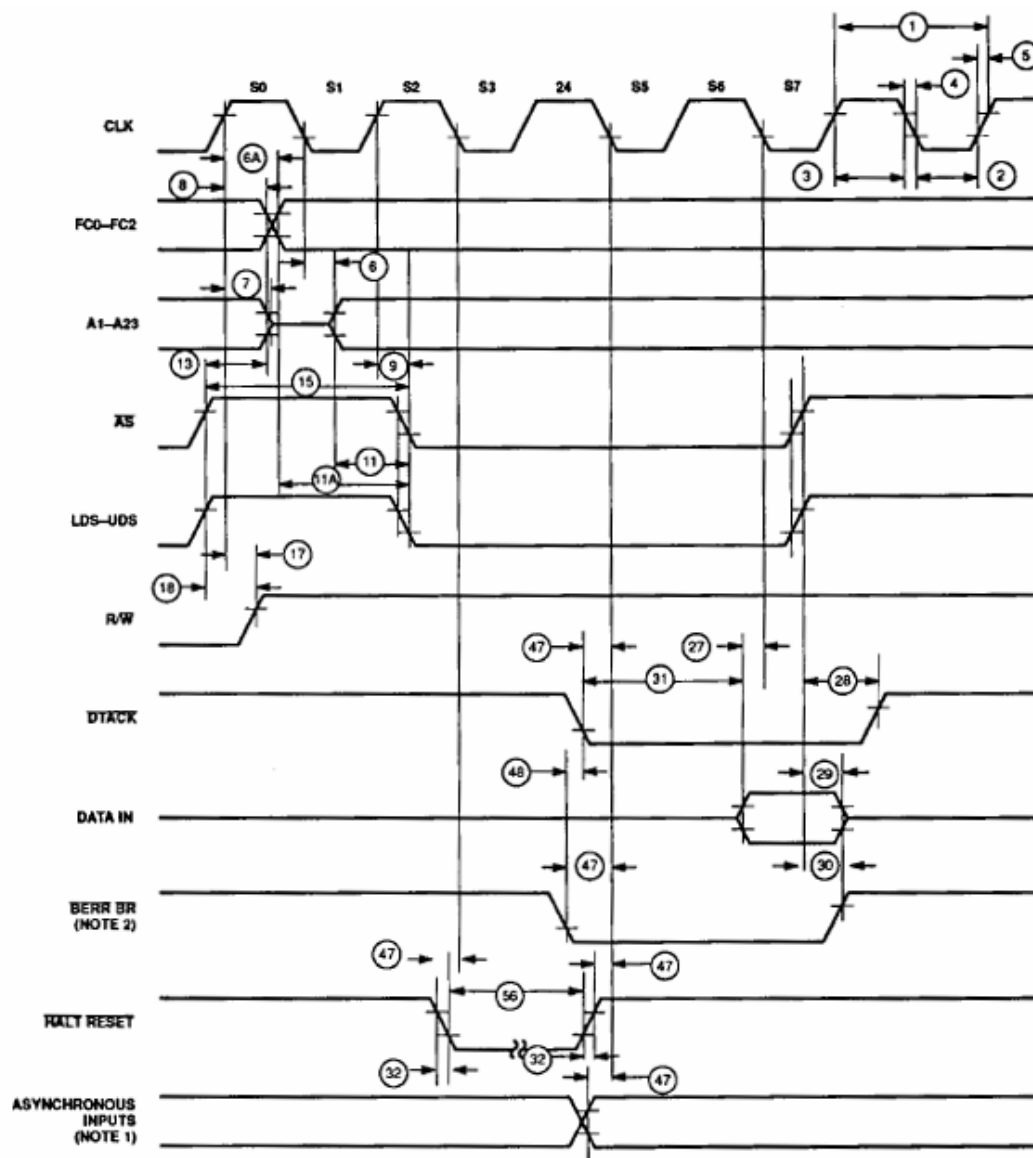


Secuencia de acciones del ciclo combinado



Cronograma para el ciclo combinado

Cronograma detallado en Lectura (68000)



A señalar:

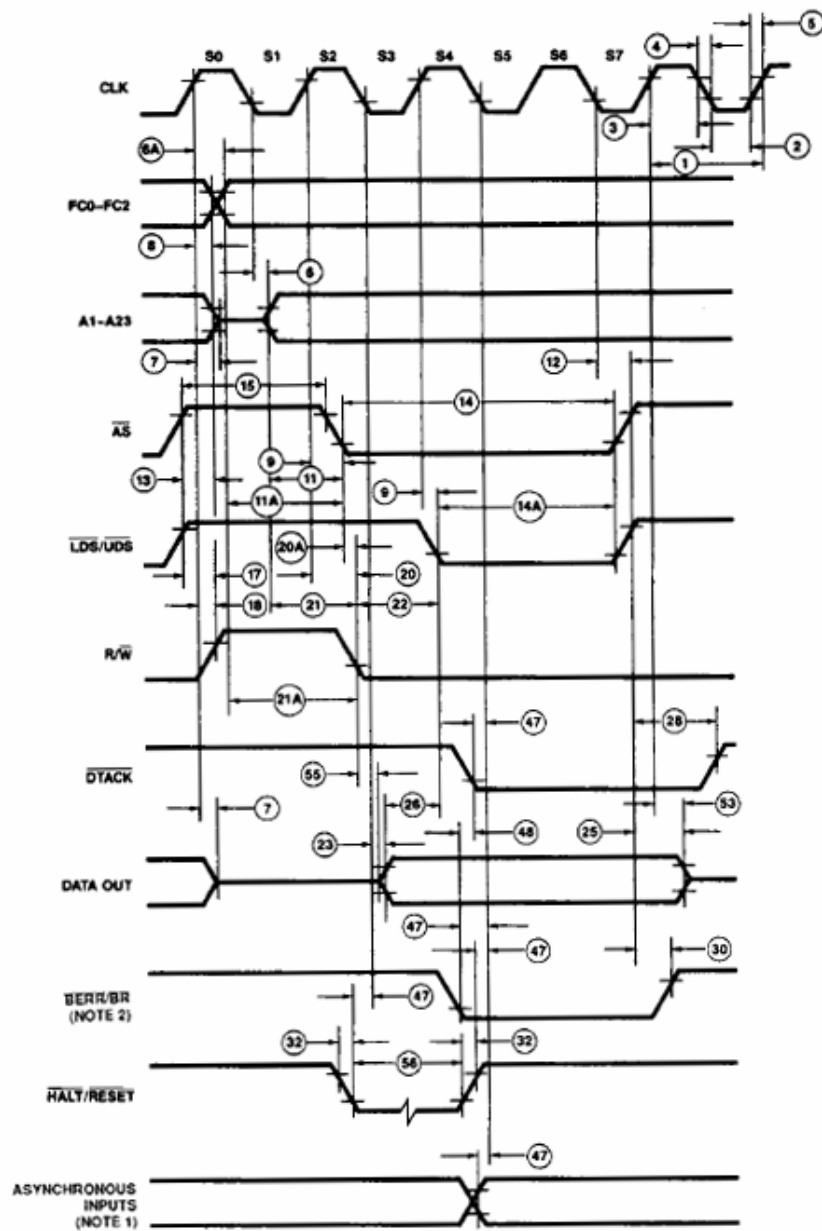
Las líneas de salida de la CPU implican retardos acotados por un valor máximo.

Las líneas que deben leerse por la CPU se leen idealmente en un instante dado pero han de tener unos tiempos de Setup y Hold acotados por valores mínimos.

Los tiempos referidos son de pines para fuera. Es por ello que, debido a los retardos internos al procesador, los tiempos de hold pueden llegar a ser 0 como es el caso de (29).

Todos los tiempos tienen, al final, como referencia la señal de reloj de la CPU.

Cronograma detallado en Escritura (68000)



AC ELECTRICAL CHARACTERISTICS¹—Read and Write Cycles –55°C ≤T_C≤+110°C (See Figures 47 and 48)

| NO. | SYMBOL | CHARACTERISTIC | 6MHz | | 8MHz | | 10MHz | | UNIT |
|-----------------------|--------------------|--|------|-----|------|-----|-------|-----|-----------|
| | | | Min | Max | Min | Max | Min | Max | |
| 1 | t _{CYC} | Clock Period | 167 | 250 | 125 | 250 | 100 | 250 | ns |
| 2 | t _{CL} | Clock width low | 75 | 125 | 55 | 125 | 45 | 125 | ns |
| 3 | t _{CH} | Clock width high | 75 | 125 | 55 | 125 | 45 | 125 | ns |
| 4 | t _{CF} | Clock fall time | — | 10 | — | 10 | — | 10 | ns |
| 5 | t _{CF} | Clock rise time | — | 10 | — | 10 | — | 10 | ns |
| 6 | t _{CLAV} | Clock low to address valid | — | 80 | — | 70 | — | 60 | ns |
| 6A | t _{CHFCV} | Clock high to FC valid | — | 80 | — | 70 | — | 60 | ns |
| 7 ⁷ | t _{CHAZx} | Clock high to address data high impedance (maximum) | — | 100 | — | 80 | — | 70 | ns |
| 8 ⁸ | t _{CHAZn} | Clock high to address/FC invalid (minimum) | 0 | — | 0 | — | 0 | — | ns |
| 9 ⁹ | t _{CHSLx} | Clock high to \overline{AS} , \overline{DS} low (maximum) | — | 70 | 0 | 60 | 0 | 55 | ns |
| 10 ¹⁰ | t _{CHSLn} | Clock high to \overline{AS} , \overline{DS} low (minimum) | — | — | — | — | — | — | ns |
| 11 ¹¹ | t _{AVSL} | Address valid to \overline{AS} , \overline{DS} (read) low/ \overline{AS} write | 35 | — | 30 | — | 20 | — | ns |
| 11A ^{7,11} | t _{FCVSL} | FC valid to \overline{AS} , \overline{DS} (read) low/ \overline{AS} (write) | 70 | — | 60 | — | 50 | — | ns |
| 12 ⁹ | t _{CLSH} | Clock low to \overline{AS} , \overline{DS} high | — | 80 | — | 70 | — | 55 | ns |
| 13 ^{7,11} | t _{SHAZ} | \overline{AS} , \overline{DS} high to address/FC invalid | 40 | — | 30 | — | 20 | — | ns |
| 14 ^{7,11} | t _{SL} | \overline{AS} , \overline{DS} width low (read/ \overline{AS} write) | 337 | — | 240 | — | 195 | — | ns |
| 14A ^{7,11} | t _{DMPW} | \overline{DS} width low (write) | 170 | — | 115 | — | 95 | — | ns |
| 15 ^{7,11} | t _{SH} | \overline{AS} , \overline{DS} width high | 180 | — | 150 | — | 105 | — | ns |
| 16 ⁷ | t _{CHSZ} | Clock high to \overline{AS} , \overline{DS} high impedance | — | 100 | — | 80 | — | 70 | ns |
| 17 ^{7,11} | t _{SHRH} | \overline{AS} , \overline{DS} high to R/W high | 50 | — | 40 | — | 20 | — | ns |
| 18 ⁹ | t _{CHRHx} | Clock high to R/W high (maximum) | — | 80 | — | 70 | — | 60 | ns |
| 19 ¹⁰ | t _{CHRHn} | Clock high to R/W high (minimum) | — | — | — | — | — | — | ns |
| 20 ⁹ | t _{CHRL} | Clock high to R/W low | — | 80 | — | 70 | — | 60 | ns |
| 20A ^{7,15} | t _{ASRV} | \overline{AS} low to R/W valid | — | 20 | — | 20 | — | 20 | ns |
| 21 ^{7,11} | t _{AVRL} | Address valid to R/W low | 25 | — | 20 | — | 0 | — | ns |
| 21A ^{7,11} | t _{FCVRL} | FC valid to R/W low | 70 | — | 60 | — | 50 | — | ns |
| 22 ^{7,11} | t _{RLSL} | R/W low to \overline{DS} low (write) | 140 | — | 80 | — | 50 | — | ns |
| 23 | t _{CLDO} | Clock low to data out valid | — | 80 | — | 70 | — | 55 | ns |
| 24 ¹³ | t _{CHRZ} | Clock high to R/W, \overline{VMA} high impedance | — | — | — | — | — | — | ns |
| 25 ^{7,11} | t _{SHDO} | \overline{DS} high to data out invalid | 40 | — | 30 | — | 20 | — | ns |
| 26 ¹¹ | t _{ODSL} | Data out valid to \overline{DS} low (write) | 35 | — | 30 | — | 20 | — | ns |
| 27 ^{7,17} | t _{DICL} | Data in to clock low (setup time) | 25 | — | 15 | — | 10 | — | ns |
| 28 ^{7,11} | t _{SHDAH} | \overline{AS} , \overline{DS} high to DTACK high | 0 | 167 | 0 | 125 | 0 | 100 | ns |
| 29 ⁷ | t _{SHDI} | \overline{DS} high to data invalid (hold time) | 0 | — | 0 | — | 0 | — | ns |
| 30 ⁷ | t _{SHBEH} | \overline{AS} , \overline{DS} high to BERR high | 0 | — | 0 | — | 0 | — | ns |
| 31 ^{7,11,17} | t _{DALDI} | DTACK low to data valid (Setup time) | — | 120 | — | 90 | — | 65 | ns |
| 32 ⁷ | t _{RLU} | HALT and RESET input transition time | 0 | 200 | 0 | 200 | 0 | 200 | ns |
| 33 | t _{CHGL} | Clock high to BG low | — | 80 | — | 70 | — | 60 | ns |
| 34 ⁷ | t _{CHGH} | Clock high to BG high | — | 80 | — | 70 | — | 60 | ns |
| 35 ⁷ | t _{BRLGL} | BR low to BG low | 1.5 | 3.5 | 1.5 | 3.5 | 1.5 | 3.5 | Clk. per. |

Tabla de tiempos para lectura y escritura

Dependiendo del tiempo considerado es normal que se especifique sólo un tiempo máximo (retardos) o mínimo (setup/hold).

AC ELECTRICAL CHARACTERISTICS¹—Read and Write Cycles (Continued)

| NO. | SYMBOL | CHARACTERISTIC | 6MHz | | 8MHz | | 10MHz | | UNIT |
|--------------------|-----------------------------|--|------|------|------|-----|-------|-----|-----------|
| | | | Min | Max | Min | Max | Min | Max | |
| 36 ^{7,18} | t _{BRHGH} | BR high to BG high | 1.5 | +100 | 1.5 | +90 | 1.5 | +80 | Cik. per. |
| 37 ⁷ | t _{BALGH} | BGACK low to BG low | 1.5 | NS | 1.5 | NS | 1.5 | NS | Cik. per. |
| 37A ⁷ | t _{BGBR} | BGACK low to BR high (to prevent re arbitration) | 20ns | 1.5 | 20ns | 1.5 | 20ns | 1.5 | Cik. per. |
| 38 ⁷ | t _{GLZ} | BG low to bus high impedance (with AS high) | — | 100 | — | 80 | — | 70 | ns |
| 39 ⁷ | t _{GH} | BG width high | 1.5 | — | 1.5 | — | 1.5 | — | Cik. per. |
| 40 | t _{CLVML} | Clock low to VMA low | — | 80 | — | 70 | — | 70 | ns |
| 41 | t _{CLC} | Clock low to E transition | — | 85 | — | 70 | — | 55 | ns |
| 42 ⁷ | t _{ED} | E output rise and fall time | — | 25 | — | 25 | — | 25 | ns |
| 43 ⁷ | t _{VMLEH} | VMA low to E high | 240 | — | 200 | — | 150 | — | ns |
| 44 ⁷ | t _{SHVPH} | AS, DS high to VPA high | 0 | 160 | 0 | 120 | 0 | 90 | ns |
| 45 ⁷ | t _{ELAI} | E low to address /VMA/FC invalid | 35 | — | 30 | — | 10 | — | ns |
| 46 ⁷ | t _{BGL} | BGACK width | 1.5 | — | 1.5 | — | 1.5 | — | Cik. per. |
| 47 ¹⁷ | t _{ASI} | Asynchronous input setup time | 25 | — | 20 | — | 20 | — | ns |
| 48 ^{7,14} | t _{BELDAL} | BERR low to DTACK low | 25 | — | 20 | — | 20 | — | ns |
| 49 ⁷ | t _{ELSI} | E low to AS, DS invalid | −80 | +80 | −70 | +70 | −55 | +55 | ns |
| 50 ⁷ | t _{EH} | E width high | 600 | — | 450 | — | 350 | — | ns |
| 51 ⁷ | E width low t _{EL} | | 900 | — | 700 | — | 550 | — | ns |
| 52 ^{7,18} | t _{CEHX} | E extended rise time | — | — | — | — | — | — | ns |
| 53 ⁷ | t _{CHDO} | Data hold from clock high | 0 | — | 0 | — | 0 | — | ns |
| 54 ^{5,7} | t _{ELDOZ} | Data hold from E low (write) | 40 | — | 30 | — | 20 | — | ns |
| 55 ⁷ | t _{RLO} | R/W to data bus impedance change | 35 | — | 30 | — | 20 | — | ns |
| 56 ^{7,12} | t _{HRPW} | HALT/RESET pulse width | 10 | — | 10 | — | 10 | — | Cik. per. |
| 57 | t _{GABD} | BGACK high to control bus driven | 1.5 | — | 1.5 | — | 1.5 | — | Cik. per. |
| 58 ¹⁸ | t _{GHBD} | BG high to control bus driven | 1.5 | — | 1.5 | — | 1.5 | — | Cik. per. |

NOTES:

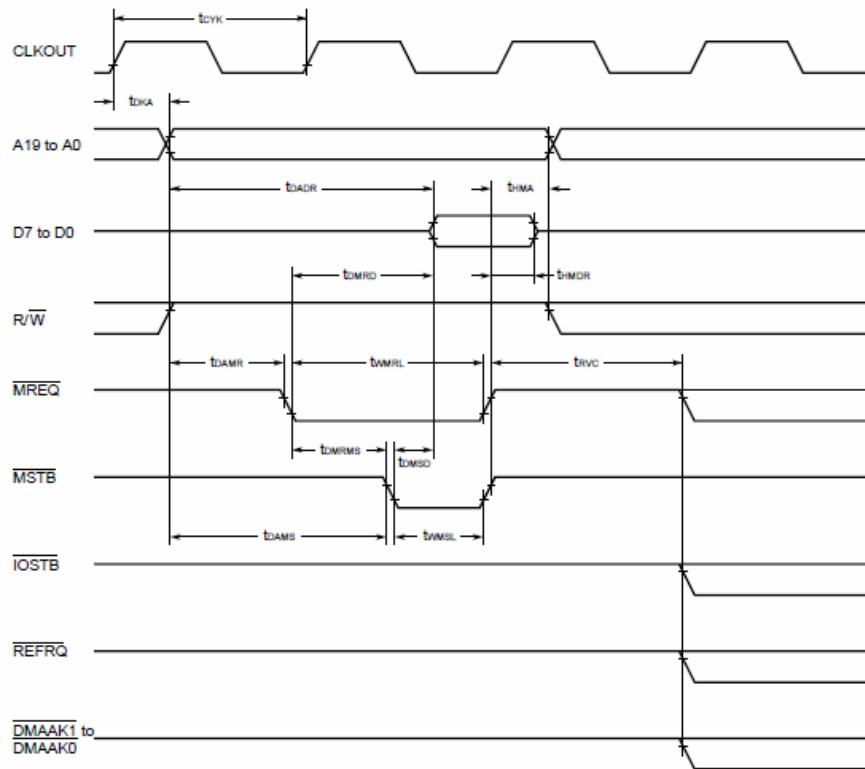
- After V_{CC} has been applied for 100 ms.
- All outputs unloaded except for load capacitance. Clock should be either 4MHz or F_{MAX}. Low; HALT, RST, (Part is held in reset). High; DTACK, BR, BGACK, IPLQ-2, VPA, BERR.
- Guaranteed but not tested.
- V_{CC} = 5V ± 5%
- After V_{CC} has been applied for 100ms.
- All outputs unloaded except for load capacitance. Clock should be either 4MHz or F_{MAX}. Low; HALT, RST, (Part is held in reset). High; DTACK, BR, BGACK, IPLQ-2, VPA, BERR.
- As a minimum, tested initially and for process or design changes only.
- For invalid, as a minimum, tested initially and for process and design changes only.
- For a loading capacitance of less than or equal to 50pF, subtract 5ns from the values given in the maximum column.
- Combined with the above parameter. Previous specification of 0ns was theoretical and not attainable.
- Actual value depends on clock period.
- For power up, the MPU must be held in RESET state for 100ms to all stabilization of on-chip circuitry. After the system is powered up, 56 refers to the minimum pulse width required to reset the system.
- Combined with 16, control bus specification
- If 47 is satisfied for both DTACK and BERR, 48 may be 0ns.
- When AS and R/W are equally loaded (±20%), subtract 10ns from the values given in these columns.
- Deleted, useful only if E clock used to drive clock input on MC6809 Microprocessor.
- If the asynchronous setup time (47) requirements are satisfied, the DTACK low to data setup time (#31) requirement can be ignored. the data must only satisfy the data in to clock-low setup time (#27) for the following cycle.
- The processor will negate BG and begin driving the bus again if external arbitration logic negates BR before asserting BTACK.

Cronogramas NEC V25

NEC

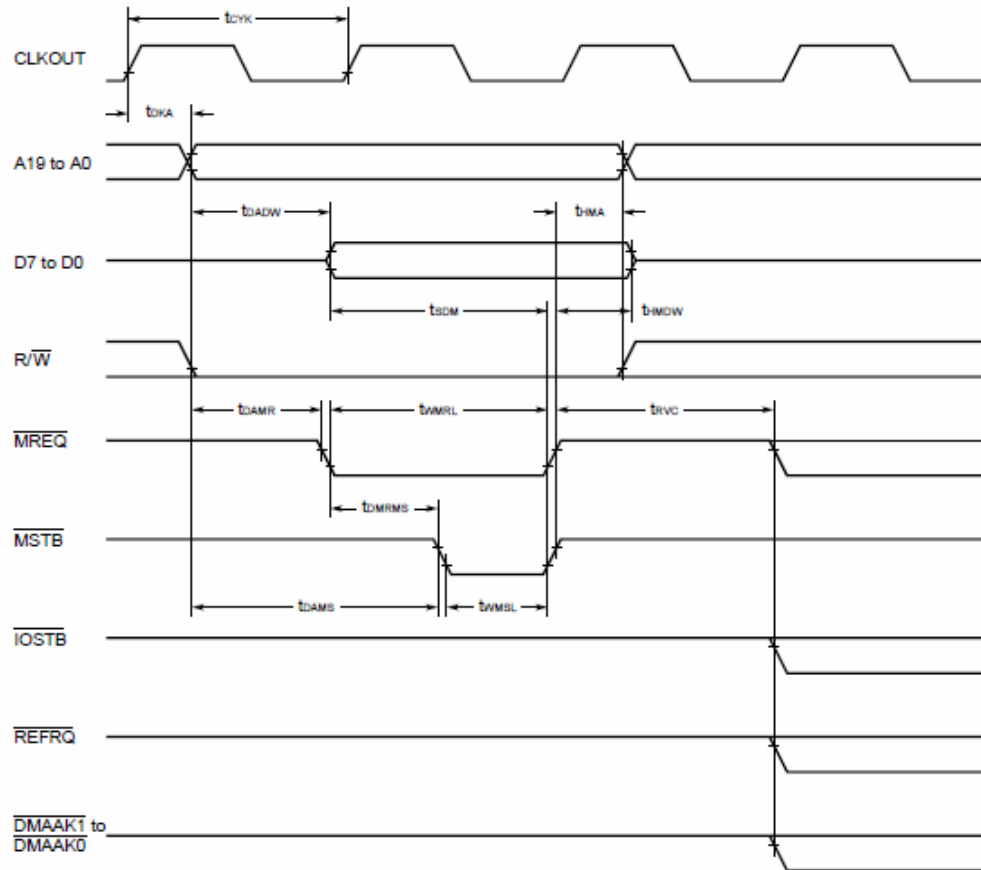
μ PD70320

READ OPERATION



A diferencia de los cronogramas del 68000, no se halla referido al reloj.

WRITE OPERATION



| Parameter | Symbol | Test Conditions | MIN. | MAX. | Unit |
|--|--------|--------------------------------------|-------------------|-------------------|------|
| Address Delay Time from CLKOUT | tDKA | | | 90 | ns |
| Data Input Delay Time from Address | tDADR | | | $(n + 1.5)T - 90$ | ns |
| Data Delay Time from $\overline{\text{MREQ}} \downarrow$ | tDMRD | | | $(n + 1)T - 75$ | ns |
| Data Delay Time from $\overline{\text{MSTB}} \downarrow$ | tDMSD | | | $(n + 0.5)T - 75$ | ns |
| $\overline{\text{MSTB}} \downarrow$ Delay Time from $\overline{\text{MREQ}} \downarrow$ | tDMRMS | | $0.5T - 35$ | $0.5T + 35$ | ns |
| $\overline{\text{MREQ}}$ Low-Level Width | tWMRL | | $(n + 1)T - 30$ | $(n + 1)T + 30$ | ns |
| Address Hold Time (from $\overline{\text{MREQ}} \uparrow$) | tHMA | | $0.5T - 30$ | | ns |
| Data Input Hold Time (from $\overline{\text{MREQ}} \uparrow$) | tHMDR | | 0 | | ns |
| Control Signal Recovery Time | tRVC | | $T - 25$ | | ns |
| Data Output Delay Time from Address | tDAOW | | | $0.5T + 50$ | ns |
| Address Setup Time (to $\overline{\text{MREQ}} \downarrow$) | tDAMR | | $0.5T - 30$ | | ns |
| Address Setup Time (to $\overline{\text{MSTB}} \downarrow$) | tDAMS | | $T - 30$ | | ns |
| $\overline{\text{MSTB}}$ Low-Level Width | tWMSL | | $(n + 0.5)T - 30$ | $(n + 0.5)T + 30$ | ns |
| Data Output Setup Time (to $\overline{\text{MSTB}} \uparrow$) | tSDM | | $(n + 1)T - 50$ | | ns |
| Data Output Hold Time (from $\overline{\text{MSTB}} \uparrow$) | tHMDW | | $0.5T - 30$ | | ns |
| Address Setup Time (to $\overline{\text{IOSTB}} \downarrow$) | tDAIS | | $0.5T - 30$ | | ns |
| Data Delay Time from $\overline{\text{IOSTB}} \downarrow$ | tDISD | | | $(n + 1)T - 90$ | ns |
| $\overline{\text{IOSTB}}$ Low-Level Width | tWISL | | $(n + 1)T - 30$ | | ns |
| Address Hold Time (from $\overline{\text{IOSTB}} \uparrow$) | tHISA | | $0.5T - 30$ | | ns |
| Data Input Hold Time (from $\overline{\text{IOREQ}} \uparrow$) | tHIDR | | 0 | | ns |
| Data Output Setup Time (to $\overline{\text{IOSTB}} \uparrow$) | tSDIS | | $(n + 1)T - 50$ | | ns |
| Data Output Hold Time (from $\overline{\text{IOSTB}} \uparrow$) | tHISDW | | $0.5T - 30$ | | ns |
| $\overline{\text{DMARQ}}$ Setup Time (to $\overline{\text{MREQ}} \downarrow$) | tSDAQ | Demand release mode | | $1T$ | ns |
| $\overline{\text{DMARQ}}$ Hold Time (from $\overline{\text{DMAAK}} \downarrow$) | tHDAG | Demand release mode | 0 | | ns |
| $\overline{\text{DMAAK}}$ Output Low-Level Width | tWDMRL | Read mode | $(n + 1.5)T - 30$ | | ns |
| $\overline{\text{TC}} \downarrow$ Delay Time from $\overline{\text{DMAAK}} \downarrow$ | tDATC | | | $0.5T + 50$ | ns |
| $\overline{\text{TC}}$ Low-Level Width | tWTCL | | $2T - 30$ | | ns |
| $\overline{\text{DMAAK}}$ Output Low-Level Width | tWDMWL | Write mode | $(n + 1)T - 30$ | | ns |
| Address Setup Time (to $\overline{\text{REFRQ}} \downarrow$) | tDARF | | $0.5T - 30$ | | ns |
| $\overline{\text{REFRQ}}$ Low-Level Width | tWRFL | | $(n + 1)T - 30$ | | ns |
| Address Hold Time (from $\overline{\text{REFRQ}} \uparrow$) | tHRFA | | $0.5T - 30$ | | ns |
| $\overline{\text{RESET}}$ Low-Level Width | tWRSL1 | STOP mode release/ power-ON reset | 30 | | ms |
| | tWRSL2 | System reset | 5 | | μs |
| READY Setup Time (to $\overline{\text{MREQ}} \downarrow$, $\overline{\text{IOSTB}} \downarrow$) | tSCRY0 | $n \geq 2$ | | $T - 100$ | ns |
| | tSCRY | $n \geq 3$ | | $(n - 1)T - 100$ | ns |

Tiempos para el μP NEC V25

Obsérvese como se usa la referencia del período del reloj para indicar los tiempos máximos y mínimos.