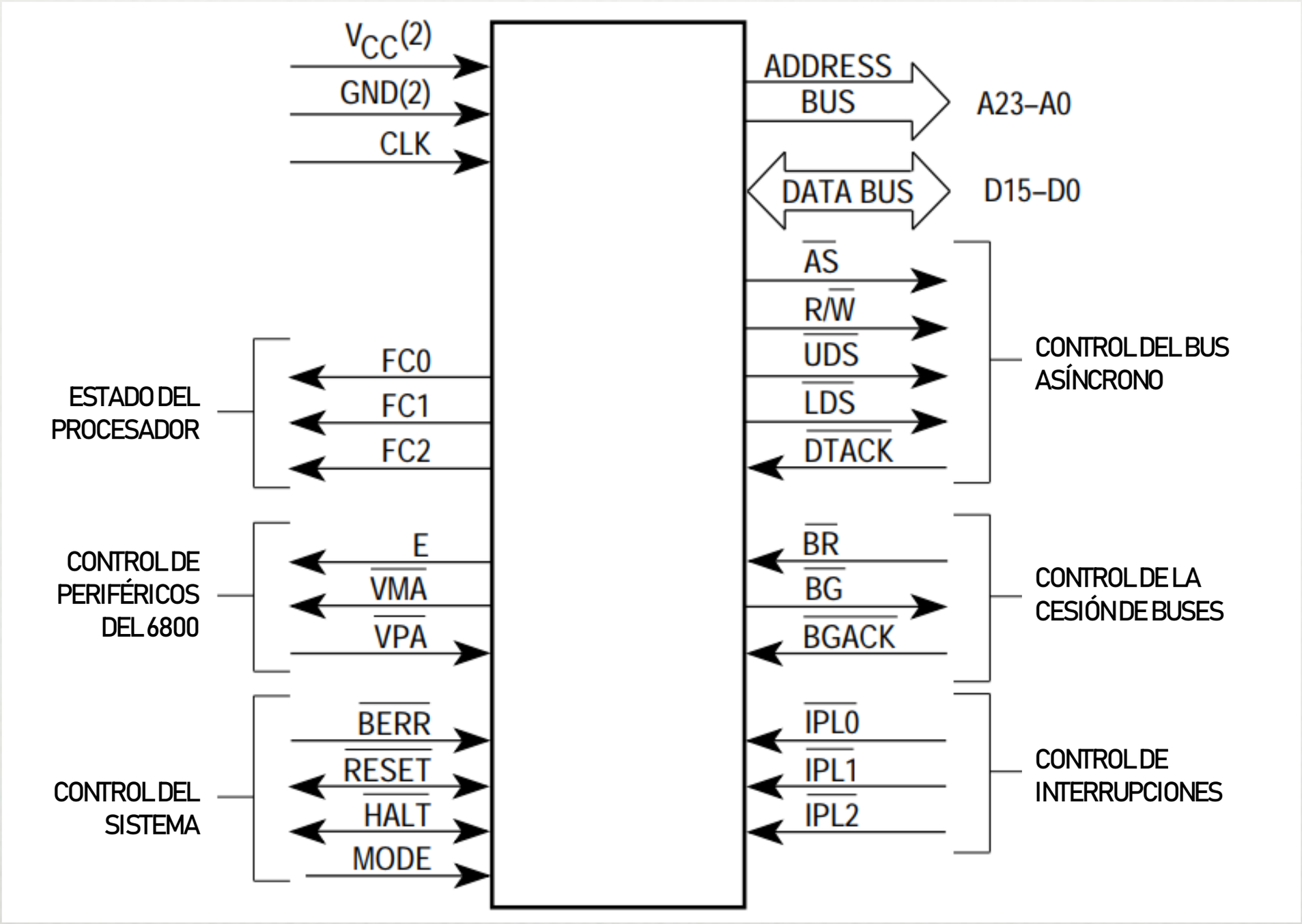
El μP 68000

Buses y señales
Ciclos de operación

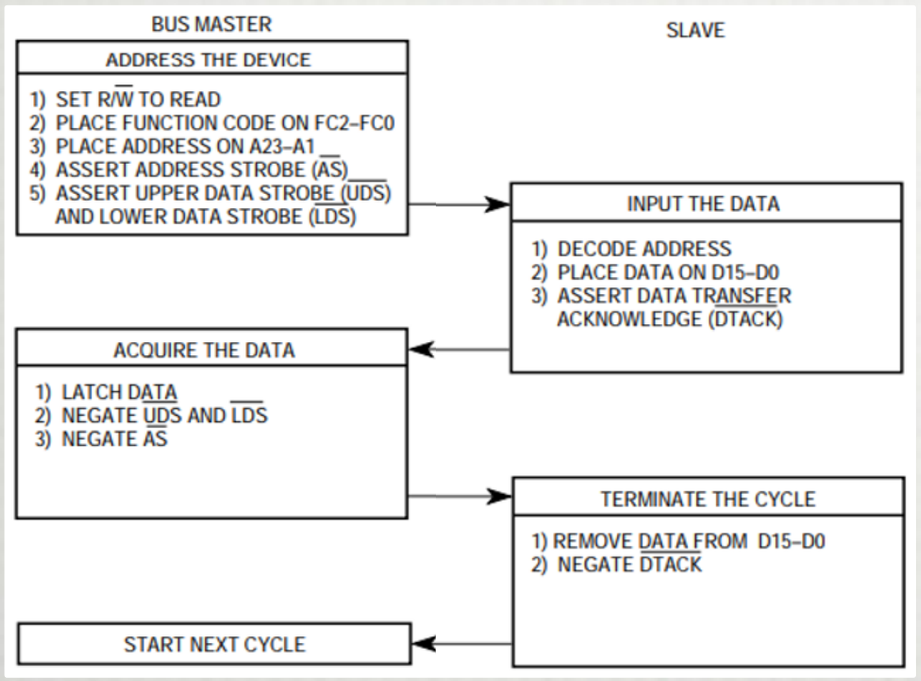
Señales del 68000

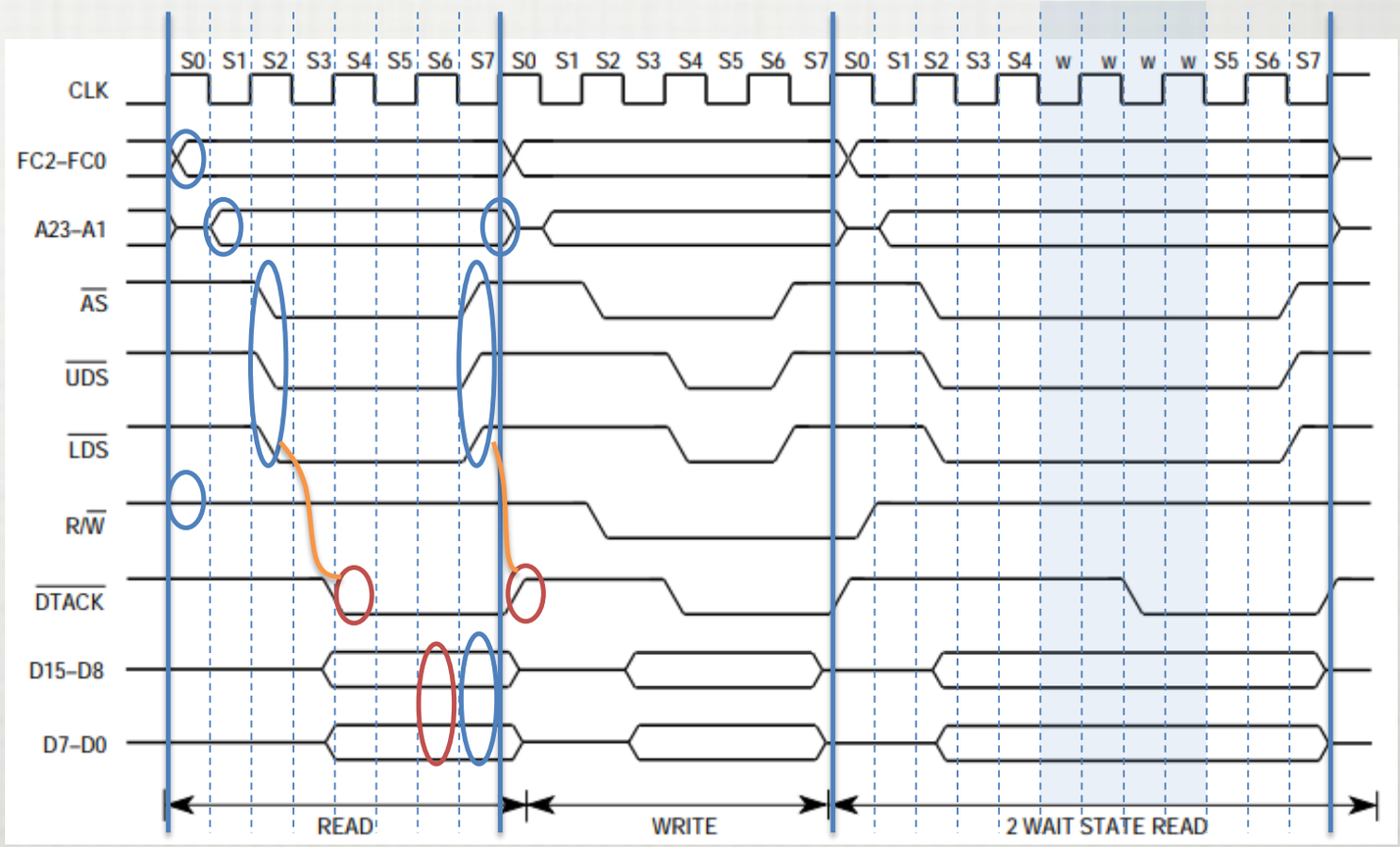


Ciclo de lectura

STATE 0 The read cycle starts in state 0 (S0). The processor places valid function codes on FC0–FC2 and drives R/W high to identify a read cycle.
STATE 1 Entering state 1 (S1), the processor drives a valid address on the address bus.
STATE 2 On the rising edge of state 2 (S2), the processor asserts AS and UDS, LDS, or DS.
STATE 3 During state 3 (S3), no bus signals are altered.
STATE 4 During state 4 (S4), the processor waits for a cycle termination signal (DTACK or BERR) or VPA, an M6800 peripheral signal. When VPA is asserted during S4, the cycle becomes a peripheral cycle. If neither termination signal is asserted before the falling edge at the end of S4, the processor inserts wait states (full clock cycles) until either DTACK or BERR is asserted.
STATE 5 During state 5 (S5), no bus signals are altered.
STATE 6 During state 6 (S6), data from the device is driven onto the data bus.
STATE 7 On the falling edge of the clock entering state 7 (S7), the processor latches data from the addressed device and negates AS, UDS, and LDS. At the rising edge of S7, the processor places the address bus in the high impedance state. The device negates DTACK or BERR at this time.

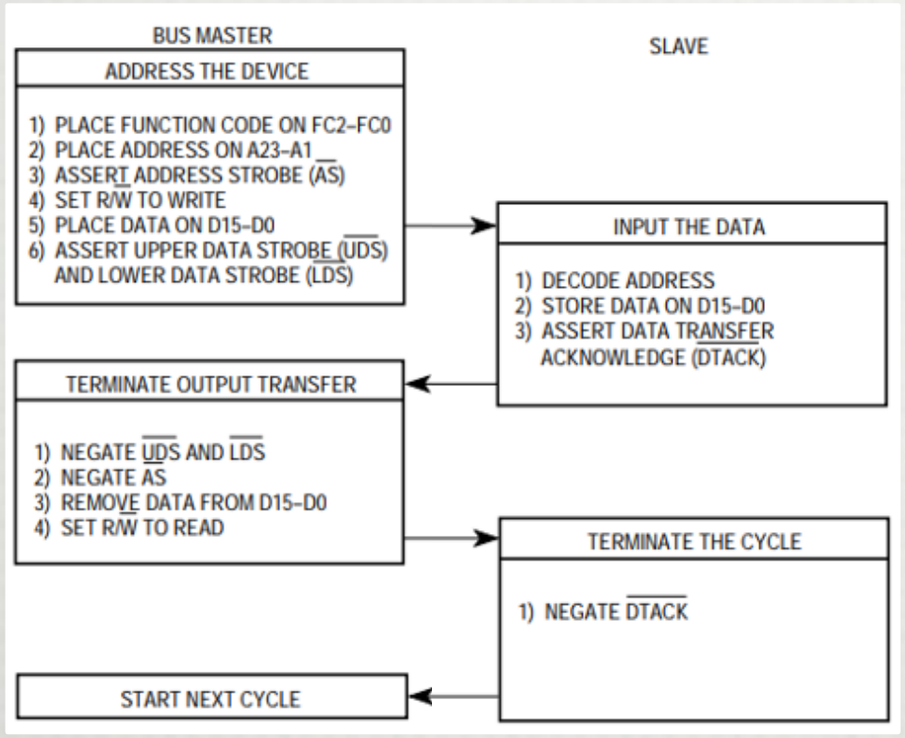
NOTE During an active bus cycle, VPA and BERR are sampled on every falling edge of the clock beginning with S4, and data is latched on the falling edge of S6 during a read cycle. The bus cycle terminates in S7, except when BERR is asserted in the absence of DTACK. In that case, the bus cycle terminates one clock cycle later in S9.

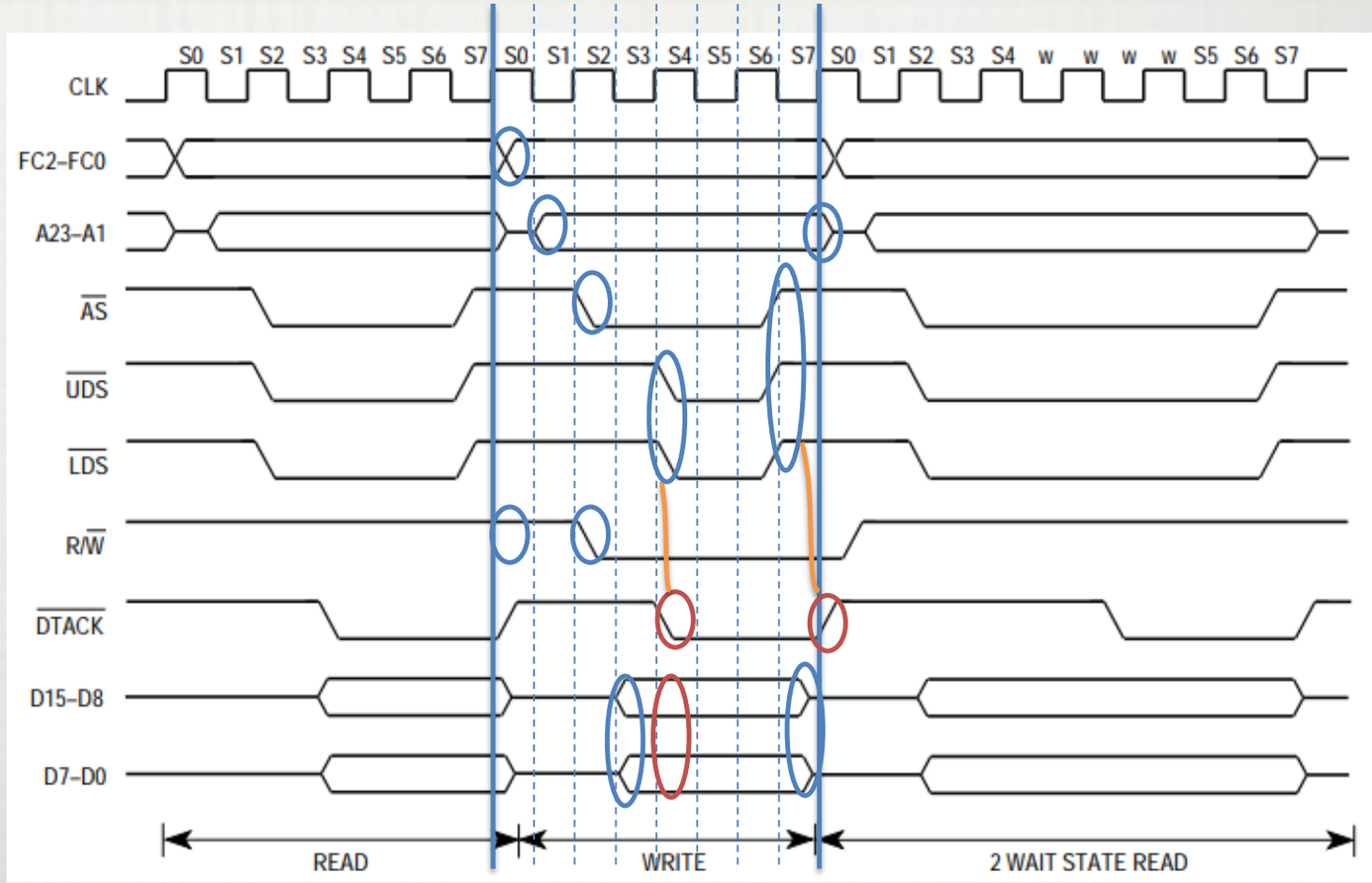




Ciclo de escritura

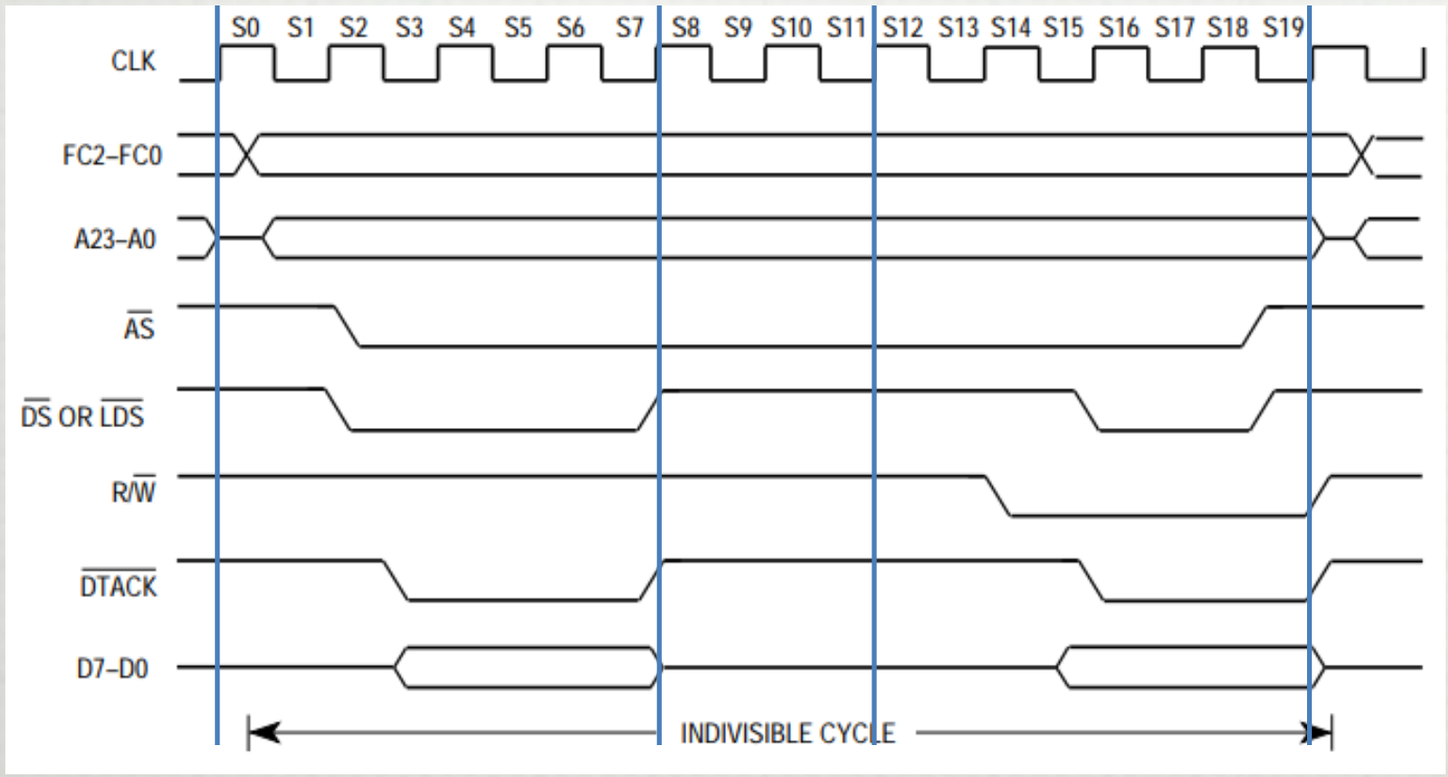
STATE 0 The write cycle starts in S0. The processor places valid function codes on FC2–FC0 and drives R/W high (if a preceding write cycle has left R/W low).
STATE 1 Entering S1, the processor drives a valid address on the address bus.
STATE 2 On the rising edge of S2, the processor asserts AS and drives R/W low.
STATE 3 During S3, the data bus is driven out of the high-impedance state as the data to be written is placed on the bus.
STATE 4 At the rising edge of S4, the processor asserts UDS, or LDS. The processor waits for a cycle termination signal (DTACK or BERR) or VPA, an M6800 peripheral signal. When VPA is asserted during S4, the cycle becomes a peripheral cycle. If neither termination signal is asserted before the falling edge at the end of S4, the processor inserts wait states (full clock cycles) until either DTACK or BERR is asserted.
STATE 5 During S5, no bus signals are altered.
STATE 6 During S6, no bus signals are altered.
STATE 7 On the falling edge of the clock entering S7, the processor negates AS, UDS, or LDS. As the clock rises at the end of S7, the processor places the address and data buses in the high-impedance state, and drives R/W high. The device negates DTACK or BERR at this time.





Ciclo de lectura-modificación-escritura

STATE 0 The read cycle starts in S0. The processor places valid function codes on FC2–FC0 and drives R/W high to identify a read cycle.
STATE 1 Entering S1, the processor drives a valid address on the address bus.
STATE 2 On the rising edge of S2, the processor asserts AS and LDS, or DS.
STATE 3 During S3, no bus signals are altered.
STATE 4 During S4the processor waits for a cycle termination signal (DTACK or BERR) or VPA, an M6800 peripheral signal. When VPA is asserted during S4, the cycle becomes a peripheral cycle. If neither termination signal is asserted before the falling edge at the end of S4, the processor inserts wait states (full clock cycles) until either DTACK or BERR is asserted.
STATE 5 During S5, no bus signals are altered.
STATE 6 During S6, data from the device are driven onto the data bus.
STATE 7 On the falling edge of the clock entering S7, the processor accepts data from the device and negates LDS, and D S. The device negates DTACK or BERR at this time.



STATES 8–11 The bus signals are unaltered during S8–S11, during which the arithmetic logic unit makes appropriate modifications to the data.

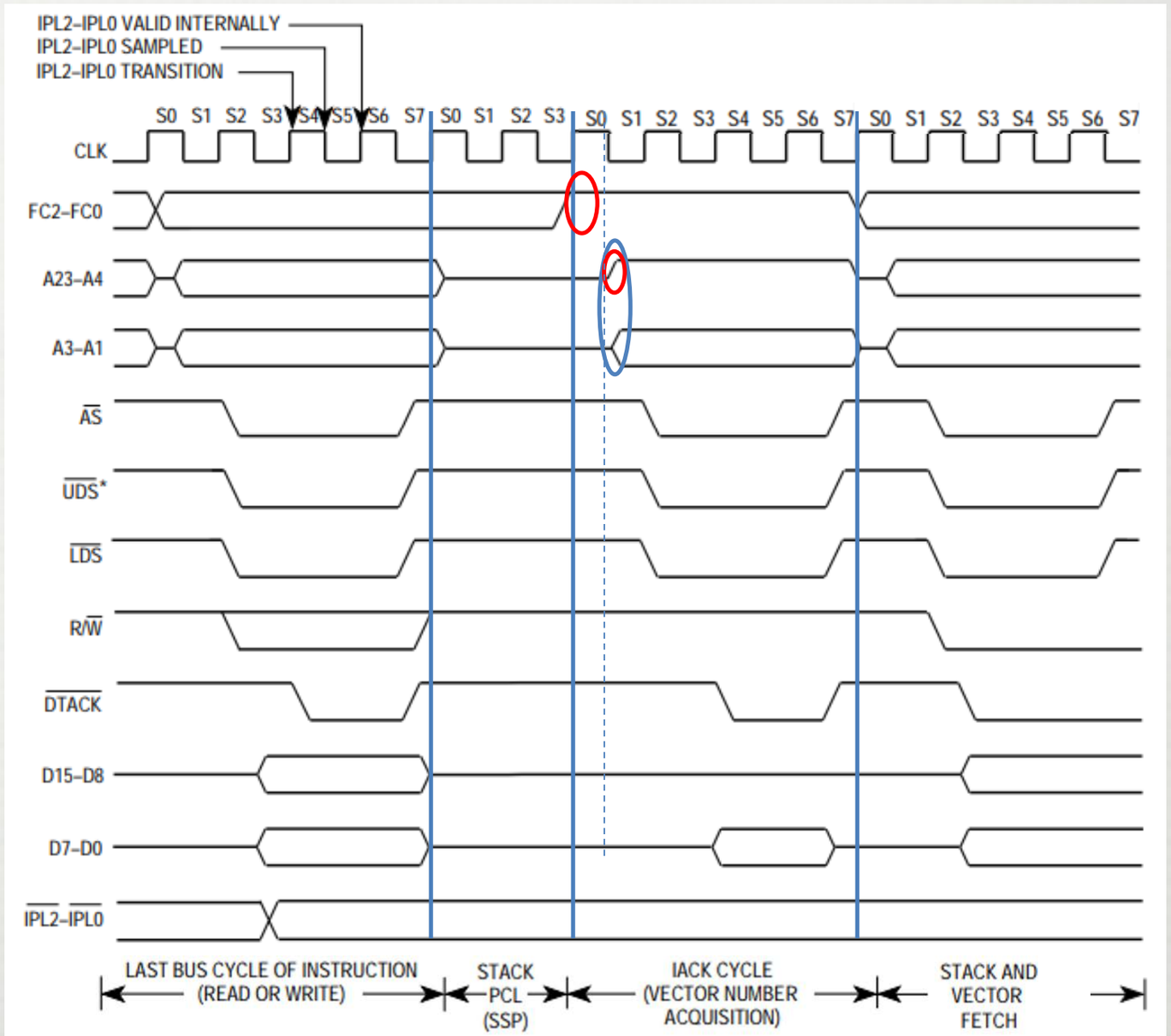
STATE 12 The write portion of the cycle starts in S12. The valid function codes on FC2–FC0, the address bus lines, AS, and R/W remain unaltered.
STATE 13 During S13, no bus signals are altered.
STATE 14 On the rising edge of S14, the processor drives R/W low.
STATE 15 During S15, the data bus is driven out of the high-impedance state as the data to be written are placed on the bus.
STATE 16 At the rising edge of S16, the processor asserts LDS, or DS. The processor waits for DTACK or BERR or VPA, an M6800 peripheral signal. When VPA is asserted during S16, the cycle becomes a peripheral cycle. If neither termination signal is asserted before the falling edge at the end of S16, the processor inserts wait states (full clock cycles) until either DTACK or BERR is asserted.
STATE 17 During S17, no bus signals are altered.
STATE 18 During S18, no bus signals are altered.
STATE 19 On the falling edge of the clock entering S19, the processor negates AS, LDS, and DS. As the clock rises at the end of S19, the processor places the address and data buses in the high-impedance state, and drives R/W high. The device negates DTACK or BERR at this time.

Ciclo de reconocimiento de interrupción

(Espacio de CPU: FC0-1-2=111)

The interrupt acknowledge cycle places the level of the interrupt being acknowledged on address bits A3–A1 and drives all other address lines high. The interrupt acknowledge cycle reads a vector number when the interrupting device places a vector number on the data bus and asserts DTACK to acknowledge the cycle.

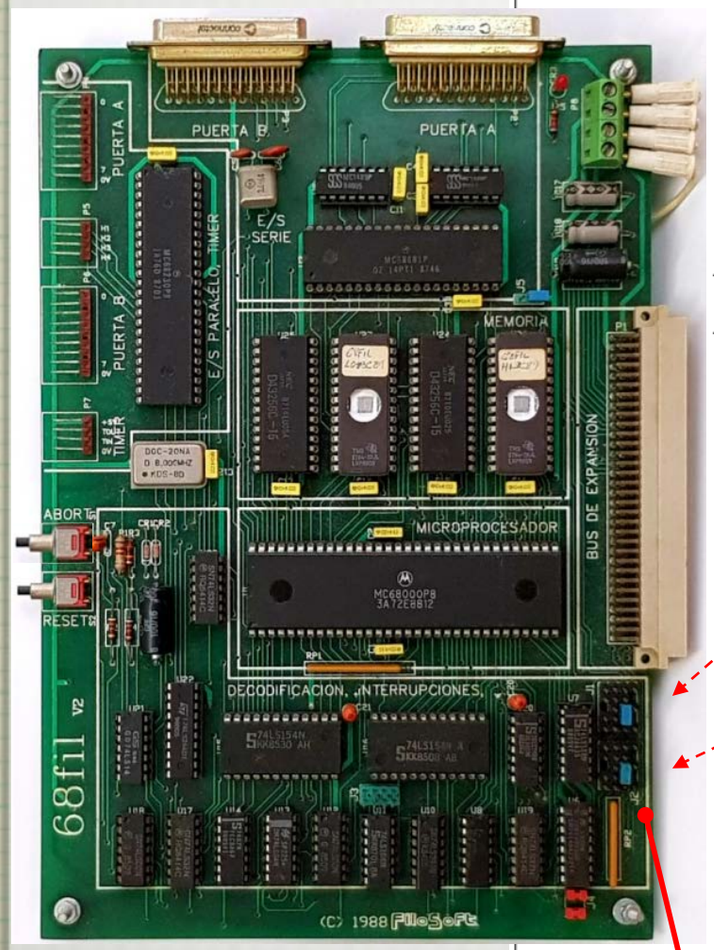
Alternately, the interrupt acknowledge cycle can be autovectored. The interrupt acknowledge cycle is the same, except the interrupting device asserts VPA instead of DTACK. For an autovectored interrupt, the vector number used is $18 plus the interrupt level. This is generated internally by the microprocessor when VPA is asserted on an interrupt acknowledge cycle. DTACK and VPA should never be simultaneously asserted.

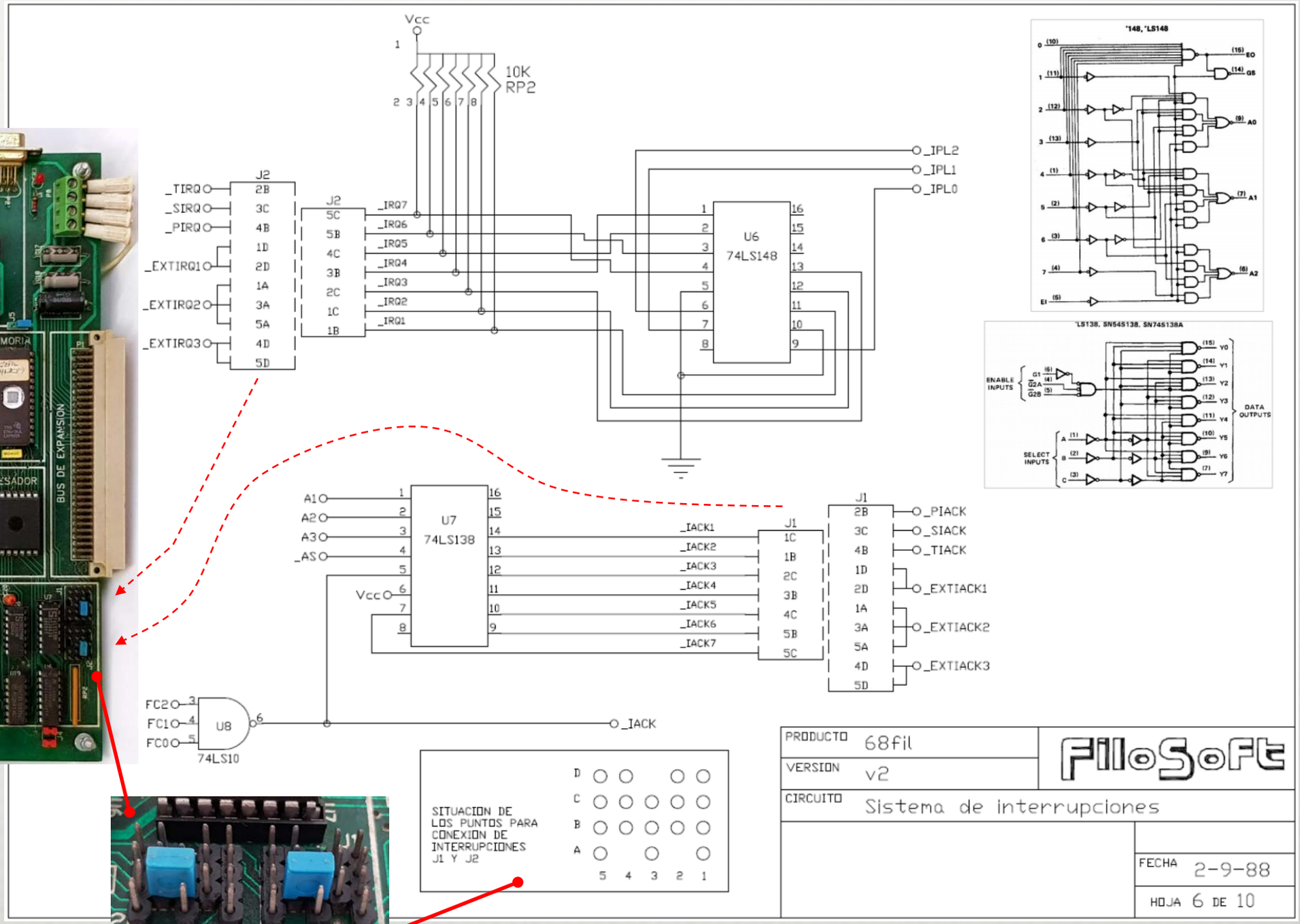


*Although a vector number is one byte, both data strobes are asserted due to the microcode used for exception processing. The processor does not recognize anything on data lines D8 through D15 at this time.

Ciclo de reconocimiento de interrupción

(un circuito de interrupción e IACK)





Reconocimiento de interrupciones: un circuito que las genera

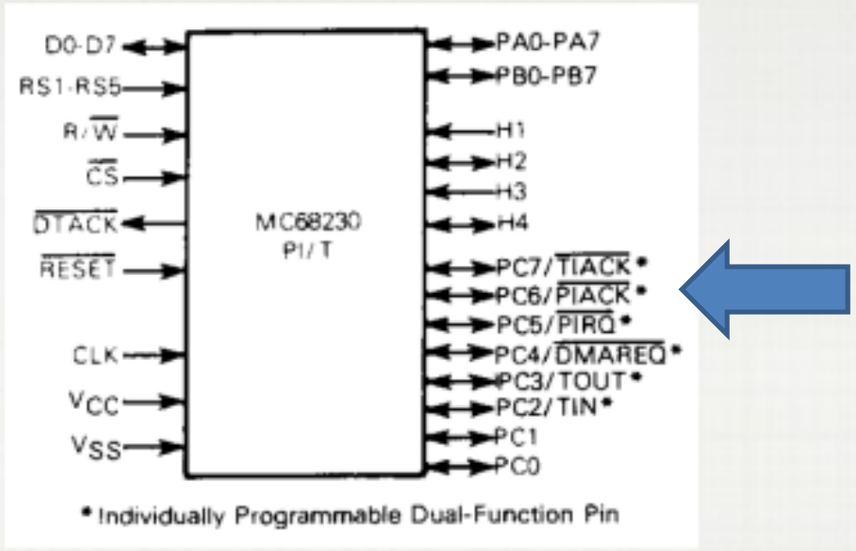


Table 1-3. Register Model (Sheet 1 of 2)

Register Select Bits 5	4	3	2	1	7	6	5	4	3	2	1	0	Register Value After RESET (Hex Value)	
0	0	0	0	0	Port Mode Control		H34 Enable	H12 Enable	H4 Sense	H3 Sense	H2 Sense	H1 Sense	0 0	Port General Contol Register
0	0	0	0	1	*	SVCRQ Select		IPF Select		Port Interrupt Priority Control			0 0	Port Service Request Register
0	0	0	1	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0 0	Port A Data Direction Register
0	0	0	1	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0 0	Port B Data Direction Register
0	0	1	0	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0 0	Port C Data Direction Register
0	0	1	0	1	Interrupt Vector Number						*	*	0 F	Port Interrupt Vector Register
0	0	1	1	0	Port A Submode		H2 Control			H2 Int Enable	H1 SVCRQ Enable	H1 Stat Ctrl	0 0	Port A Control Register
0	0	1	1	1	Port B Submode		H4 Control			H4 Int Enable	H3 SVCRQ Enable	H3 Stat Ctrl	0 0	Port B Control Register
0	1	0	0	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	**	Port A Data Register
0	1	0	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	**	Port B Data Register
0	1	0	1	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	***	Port A Alternate Register
0	1	0	1	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	***	Port B Alternate Register
0	1	1	0	0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	****	Port C Data Register
0	1	1	0	1	H4 Level	H3 Level	H2 Level	H1 Level	H4S	H3S	H2S	H1S	****	Port Status Register
0	1	1	1	0	*	*	*	*	*	*	*	*	0 0	(Null)
0	1	1	1	1	*	*	*	*	*	*	*	*	0 0	(Null)

* Unused, read as zero
** Value before RESET
*** Current value on pins
**** Undetermined value

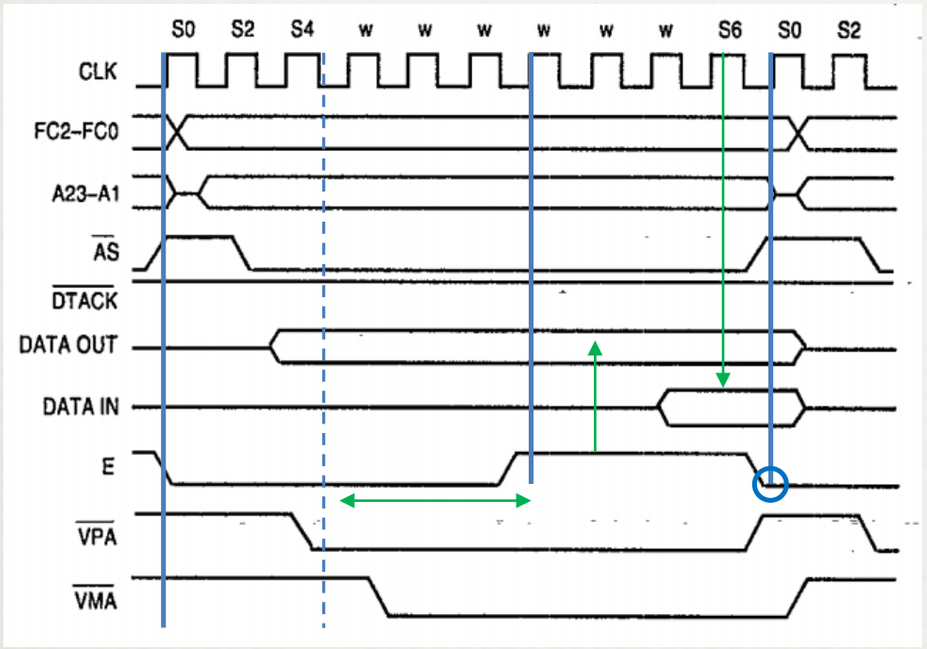
Table 1-3. Register Model (Sheet 2 of 2)

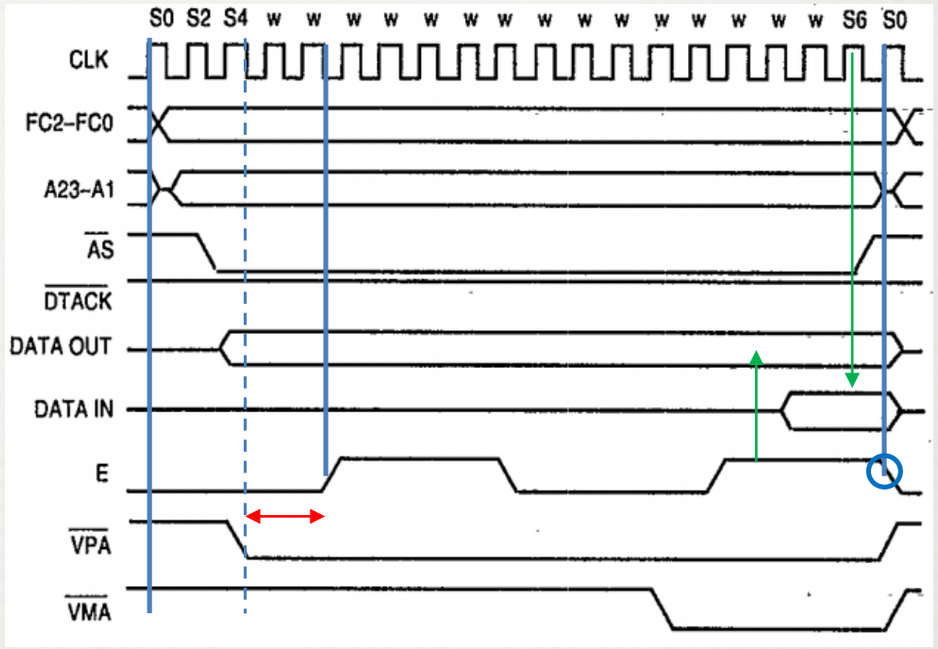
Register Select Bits 5	4	3	2	1	7	6	5	4	3	2	1	0	Register Value After RESET (Hex Value)	
1	0	0	0	0	TOUT/TIACK Control			Z D Ctrl	*	Clock Control		Timer Enable	0 0	Timer Control Register
1	0	0	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0 F	Timer Interrupt Vector Register
1	0	0	1	0	*	*	*	*	*	*	*	*	0 0	(Null)
1	0	0	1	1	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	**	Counter Preload Register (High)
1	0	1	0	0	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	**	Counter Preload Register (Mid)
1	0	1	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	**	Counter Preload Register (Low)
1	0	1	1	0	*	*	*	*	*	*	*	*	0 0	(Null)
1	0	1	1	1	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	**	Count Register (High)
1	1	0	0	0	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	**	Count Register (Mid)
1	1	0	0	1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	**	Count Register (Low)
1	1	0	1	0	*	*	*	*	*	*	*	ZDS	0 0	Timer Status Register
1	1	0	1	1	*	*	*	*	*	*	*	*	0 0	(Null)
1	1	1	0	0	*	*	*	*	*	*	*	*	0 0	(Null)
1	1	1	0	1	*	*	*	*	*	*	*	*	0 0	(Null)
1	1	1	1	0	*	*	*	*	*	*	*	*	0 0	(Null)
1	1	1	1	1	*	*	*	*	*	*	*	*	0 0	(Null)

* Unused, read as zero
** Value before RESET

Ciclos síncronos

Los 4 estados iniciales son los habituales en la iniciación de un ciclo de lectura/escritura.
El procesador queda esperando a DTACK/VPA(Valid Peripheral Address)/ BERR.
Cuando aparece VPA “entiende” que se trata de una transferencia síncrona con la señal E (generada por él mismo manteniéndola baja durante 6 ciclos de CLK y alta durante 4 ciclos)
Espera (si es necesario) a que E esté baja, y activa (baja) VMA (Valid Memory Address), que participa en el CS (Chip Select) del periférico.
El periférico realiza la acción (lectura/escritura) durante el estado alto de E. Si es una lectura, el procesador la realiza en el estado S6.





Hay que tener en cuenta que los instantes en que terminan los ciclos asíncronos y E no se mantienen sincronizados (en la fig. 1 lo están y en la 2 no).

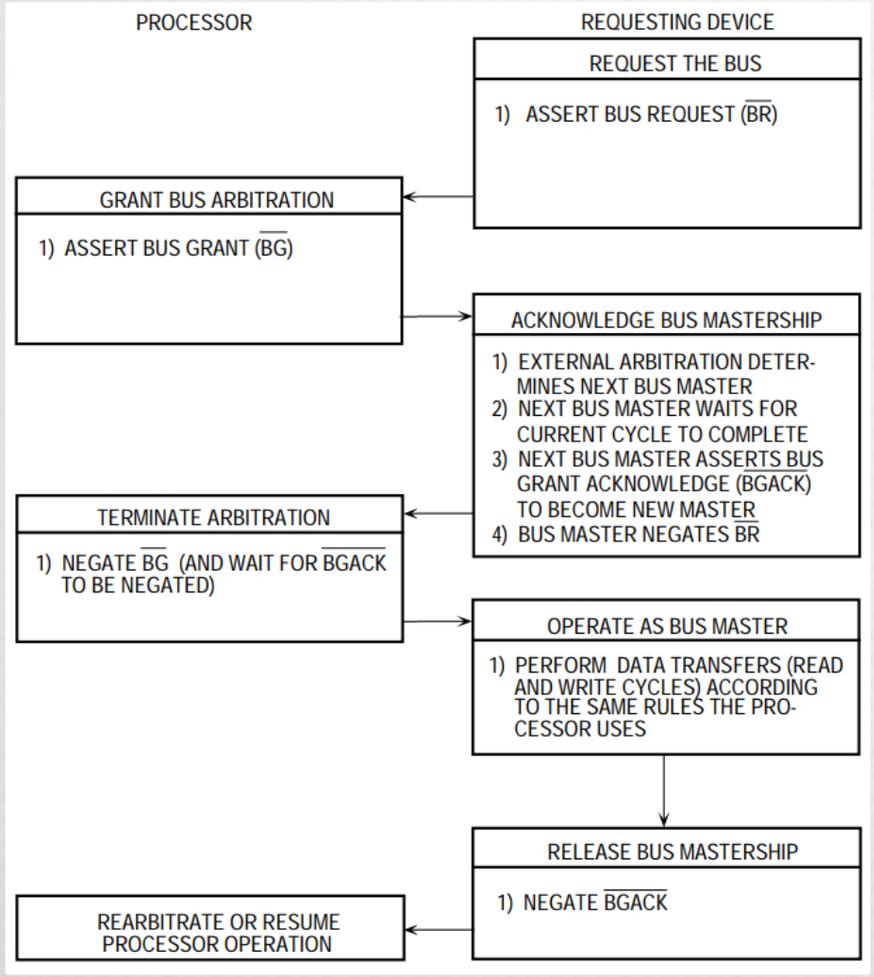
El “mejor caso”, que se da cuando el ciclo anterior termina justo en el flanco de bajada de E (al procesador le da tiempo a ver que E está baja tras ver la VPA). Se introducen los mínimos estados de espera para sincronizar con E.

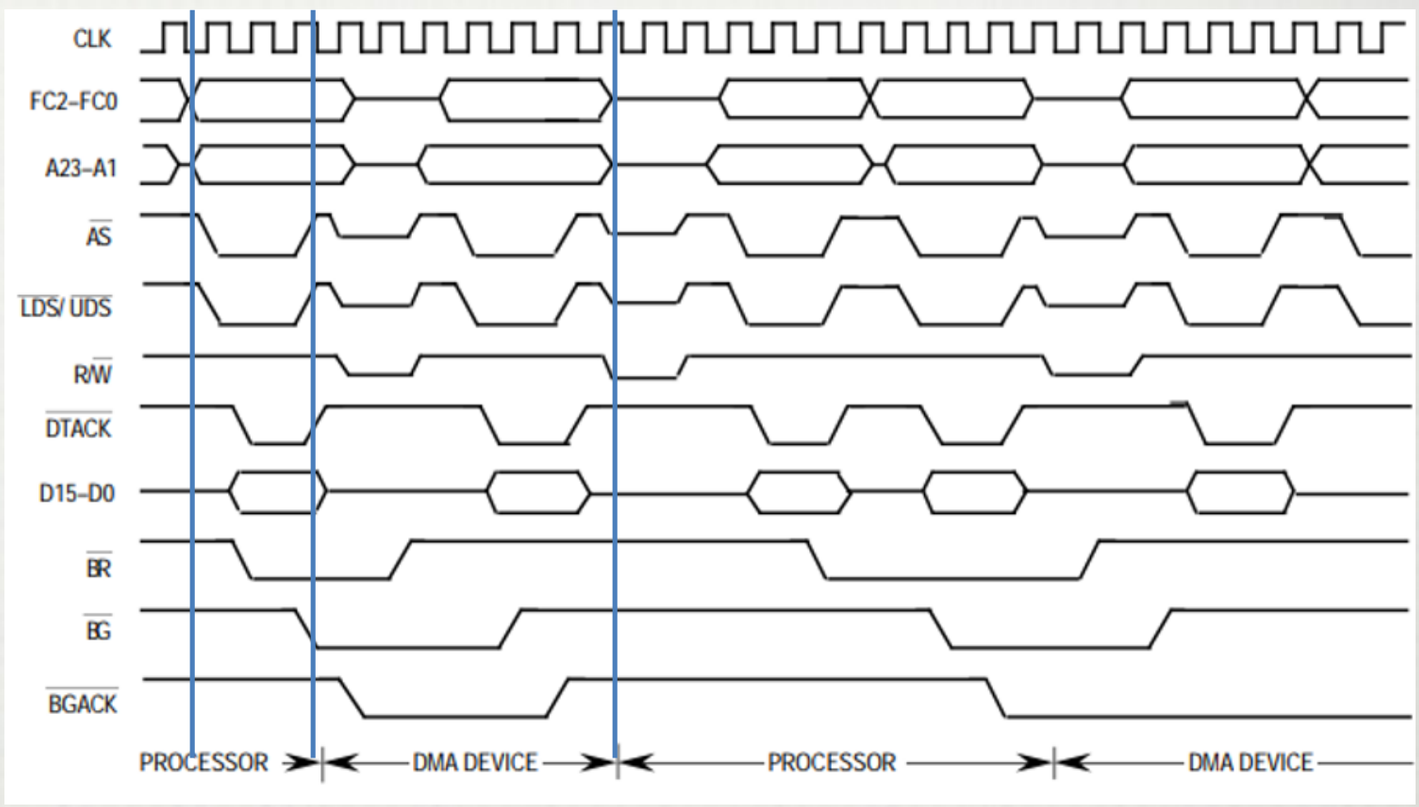
El peor caso se da cuando VPA aparece cuando faltan menos de dos ciclos para que E suba (el procesador no tiene tiempo de comprobar el estado e E como bajo).

Cesión de buses

Bus arbitration is a technique used by bus master devices to request, to be granted, and to acknowledge bus mastership. Bus arbitration consists of the following:

1. Asserting a bus mastership request
2. Receiving a grant indicating that the bus is available at the end of the current cycle
3. Acknowledging that mastership has been assumed

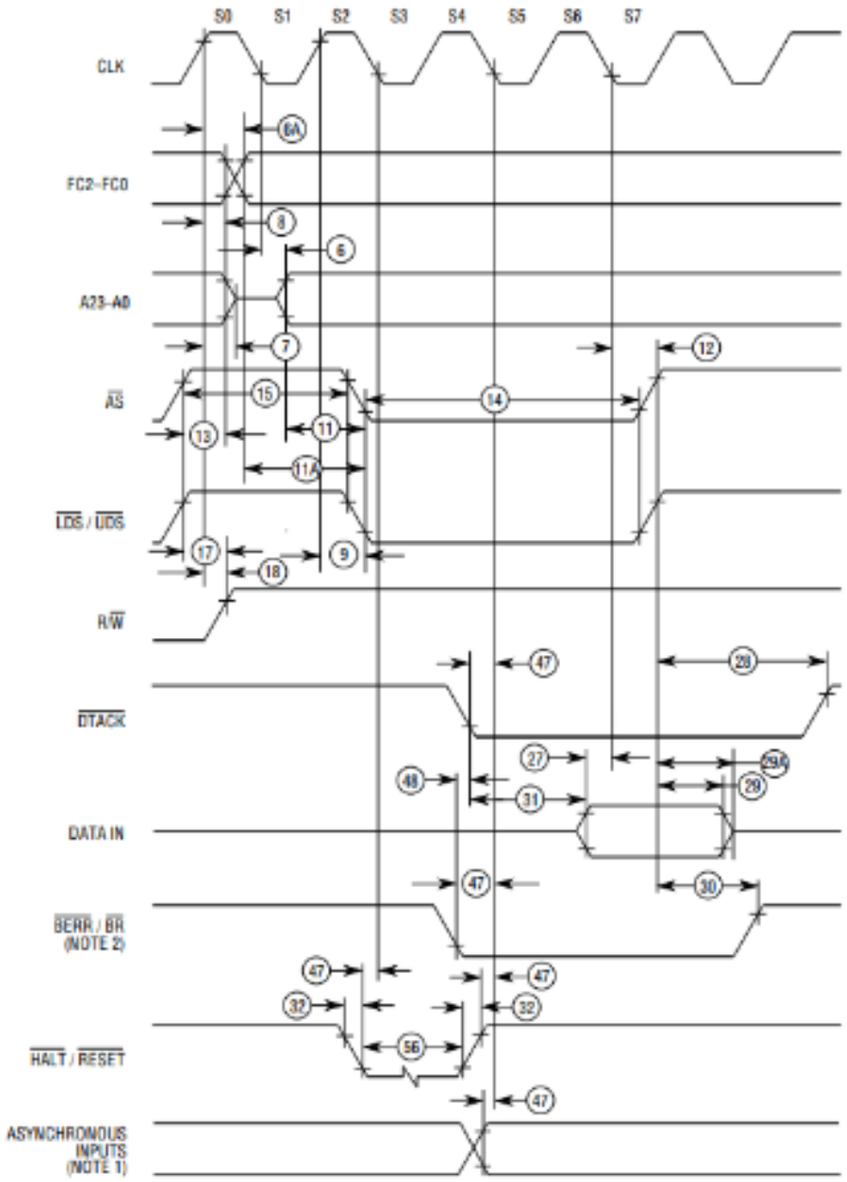




**Este gráfico es bastante impreciso*

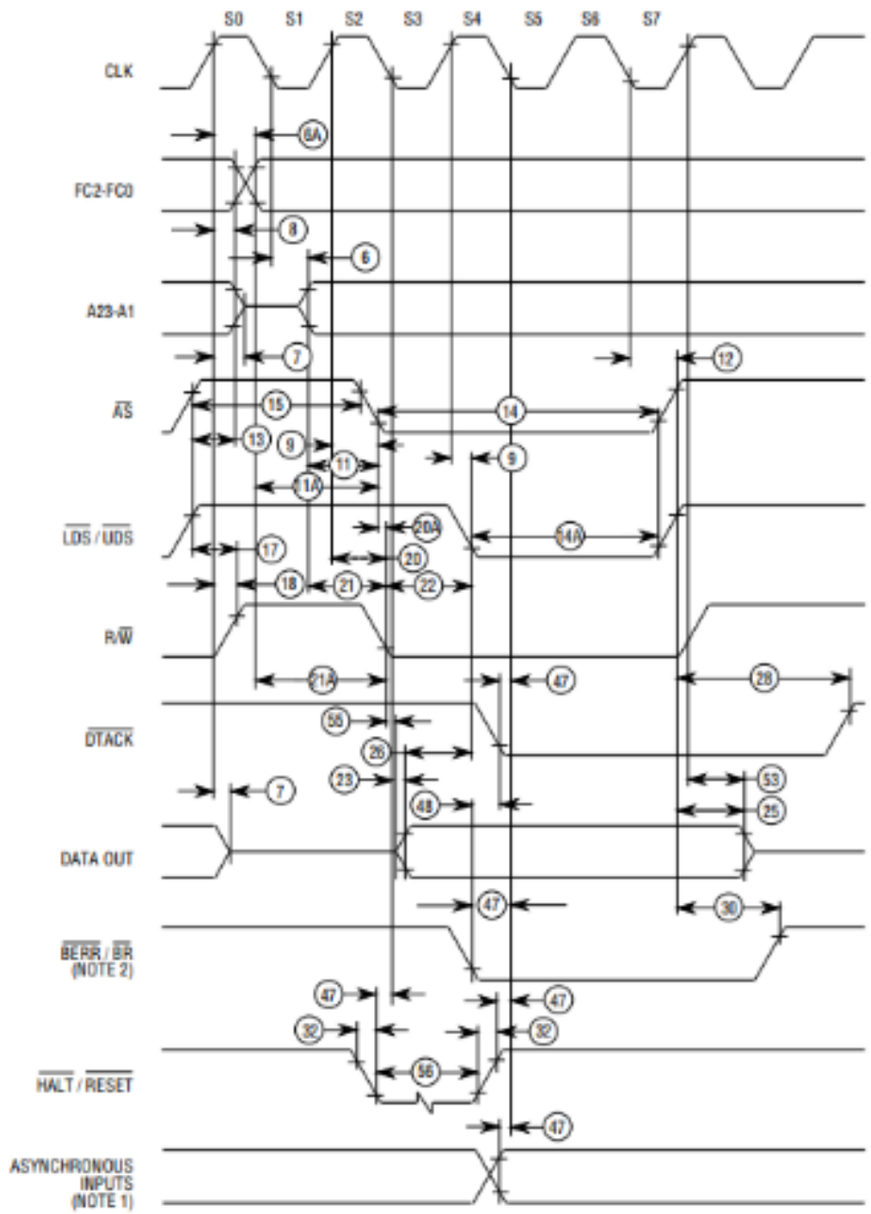
También existe un protocolo de cesión a 2 hilos (BR-BG)

Ciclos: todos los intervalos tienen sus "timings".



NOTES:
1. Setup time for the asynchronous inputs IPL2–IPL0 and AVEC (#47) guarantees their recognition at the next falling edge of the clock.
2. BR need fall at this time only to insure being recognized at the end of the bus cycle.
3. Timing measurements are referenced to and from a low voltage of 0.8 V and a high voltage of 2.0 V, unless otherwise noted. The voltage swing through this range should start outside and pass through the range such that the rise or fall is linear between 0.8 V and 2.0 V.

Figure 10-4. Read Cycle Timing Diagram
(Applies To All Processors Except The MC68EC000)



NOTES:
1. Timing measurements are referenced to and from a low voltage of 0.8 V and a high voltage of 2.0 V, unless otherwise noted. The voltage swing through this range should start outside and pass through the range such that the rise or fall is linear between 0.8 V and 2.0 V.
2. Because of loading variations, R/W may be valid after AS even though both are initiated by the rising edge of S2 (specification #20A).

Figure 10-5. Write Cycle Timing Diagram
(Applies To All Processors Except The MC68EC000)