ICOM 5217 – Microprocessor Interfacing

Control Signal Architectures, Transfer Synchronization and Multiplexed Address/Data Buses

Prof. Manuel Jimenez

Electrical & Computer Engineering

With contributions by
George Suárez-Martinez

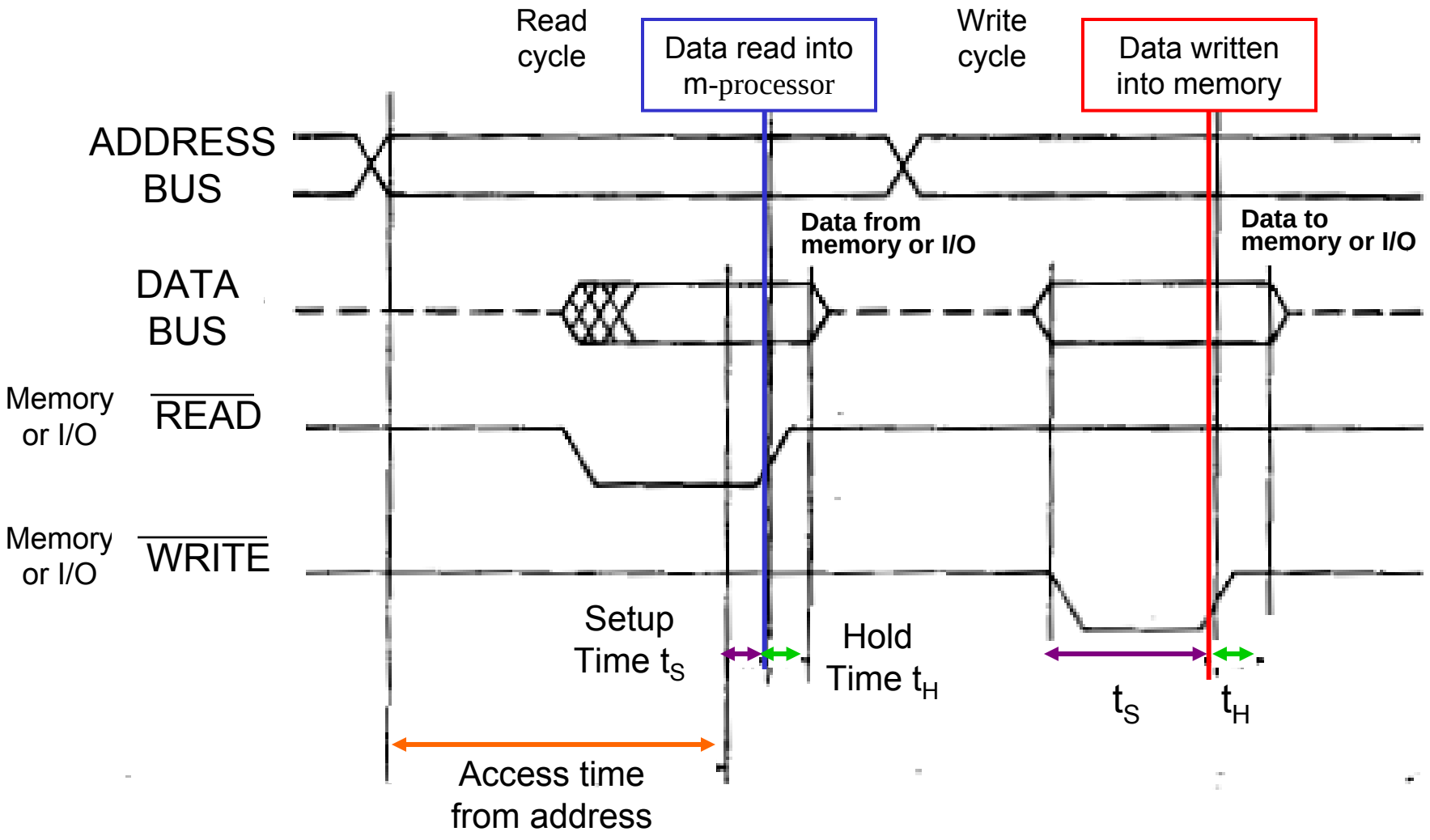
RECINTO UNIVERSITARIO DE MAYAGÜEZ
UNIVERSIDAD DE PUERTO RICO
CAAM

Topics

- Control Signal Architectures
  - Intel-style 
  - Motorola-style 
- Transfer Synchronization
  - Synchronous buses
  - Asynchronous buses
  - Semi-synchronous buses
- Multiplexed Address/Data buses

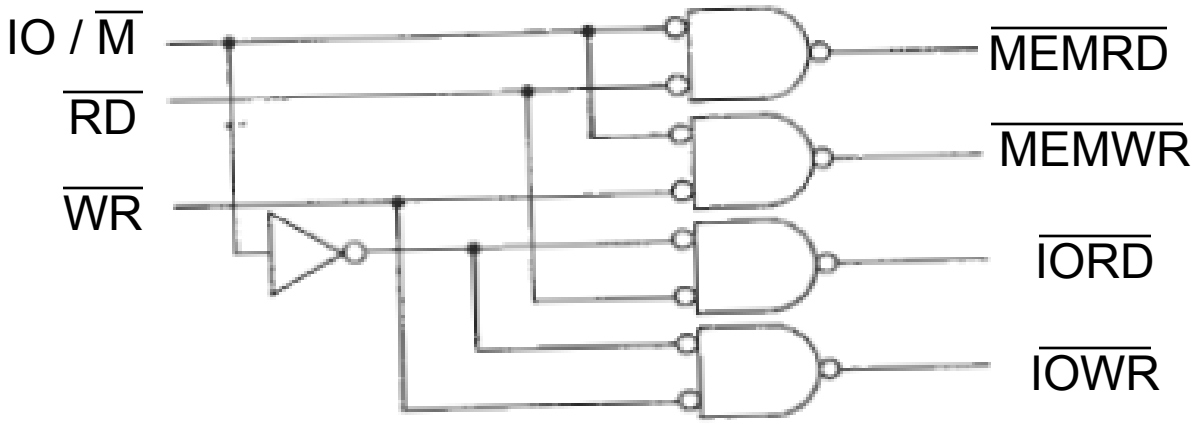
Intel-style

- Basic Intel bus timing (8080)



Intel-style

- Control signal gating

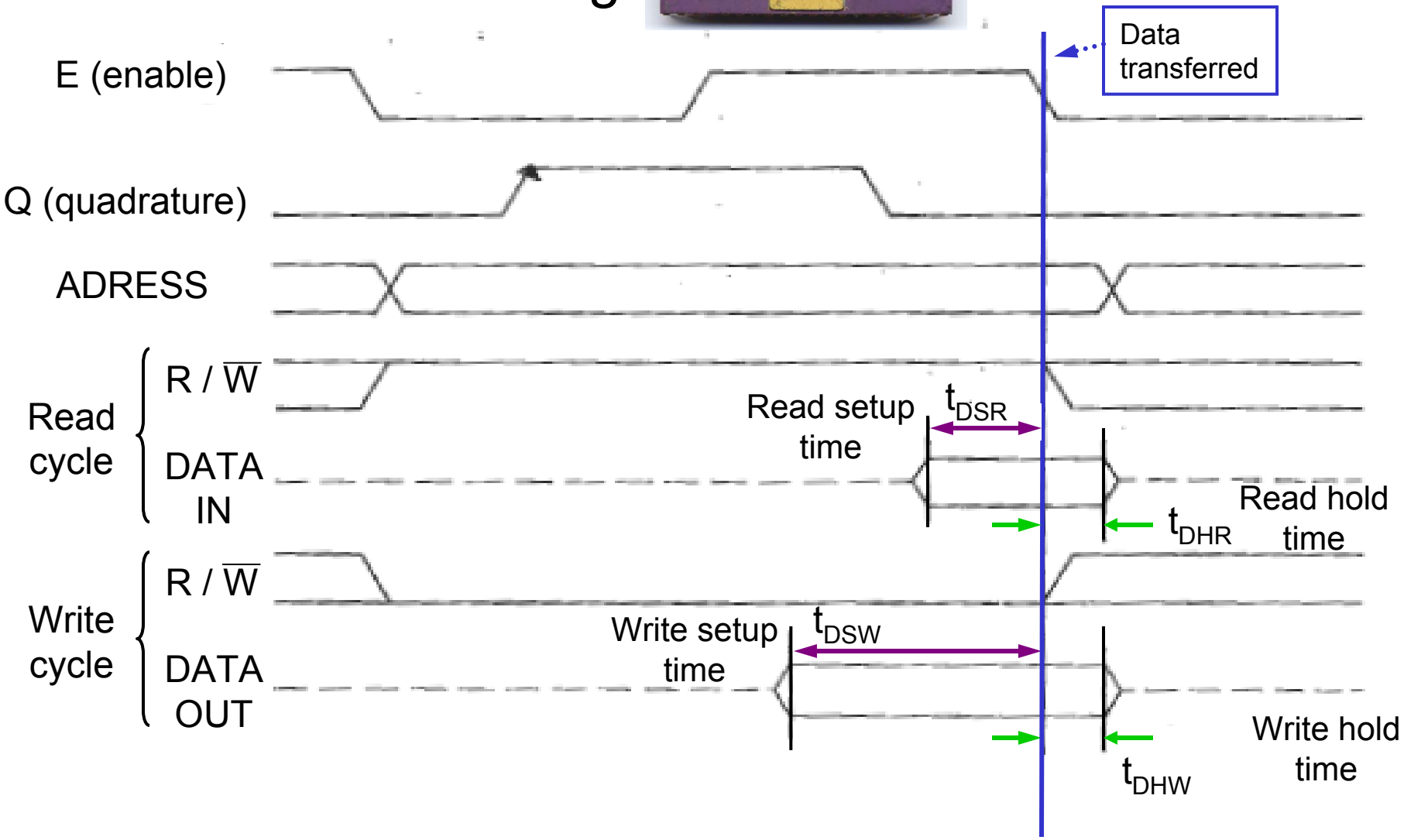


- For memory-mapped I/O signal IO/M can be ignored

- Other manufacturers such as Zilog follow a similar approach

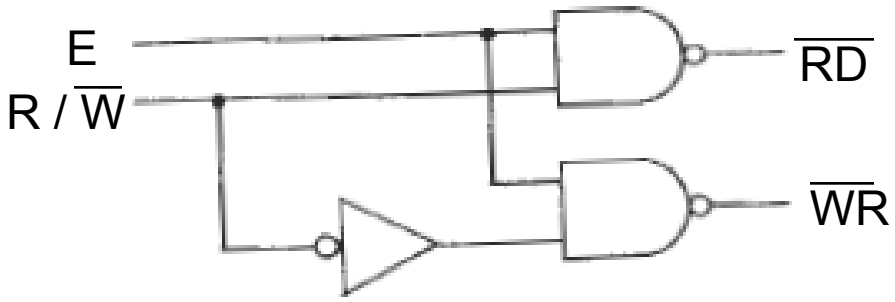
Motorola-style

- The 6809 bus timing



Motorola-style

- The 6809 control signal gating



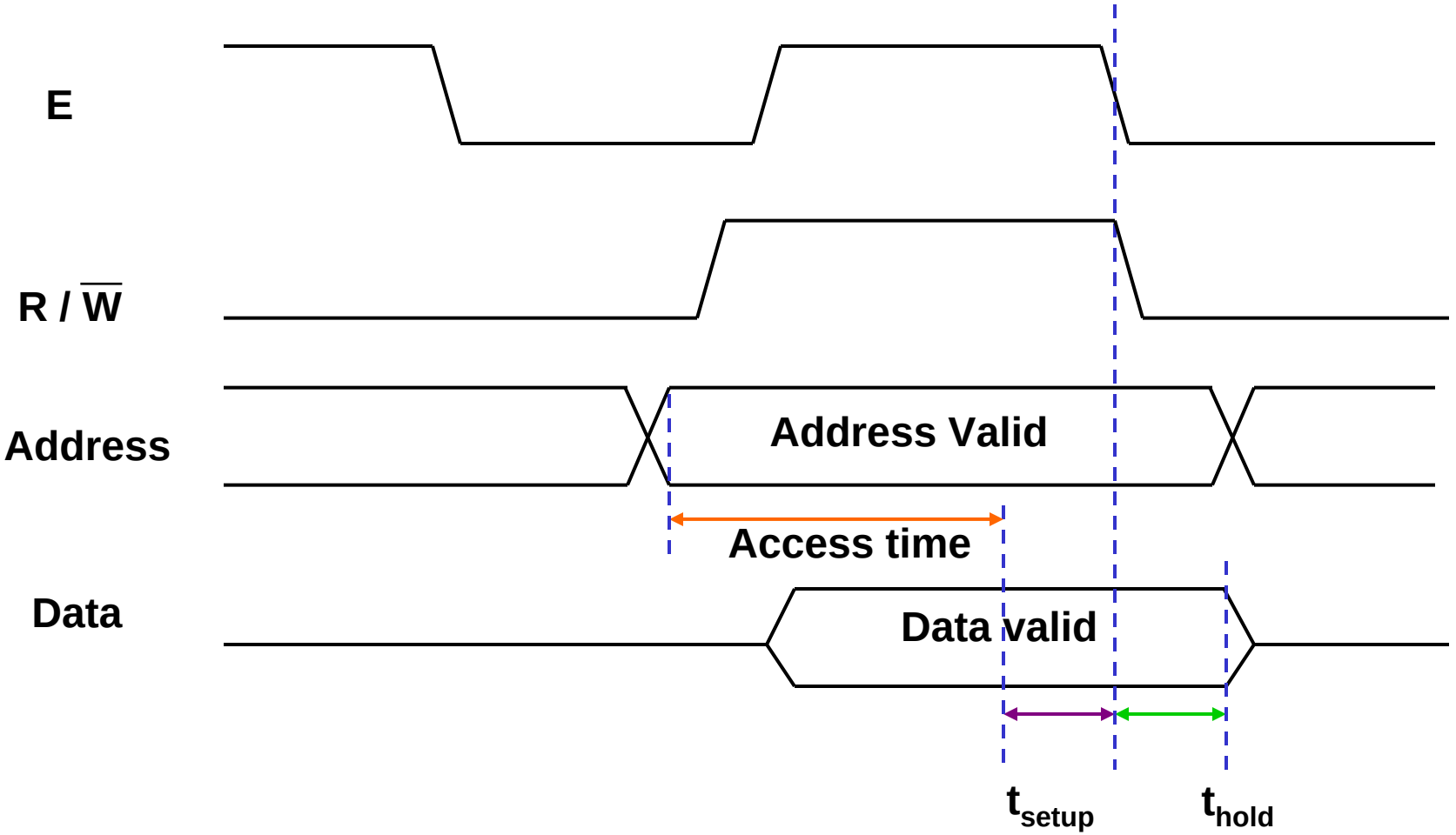
- DIFFERENCES
  - Intel-style read (R) and write (W) strobes contain both direction control and timing in the same signals
  - Motorola read/write (R/$\overline{W}$) direction control line that contains no timing information.

Synchronous buses

- **NO facilities for varying the bus timing!**
- An awkward schema might be dynamically changing the clock period
  - Make use of special clock generators to vary clock speed (“clock stretch”)
  - Change the divisor when accessing slower memory or peripheral devices
- More commonly, slowing down the system speed
  - Divide a high-speed clock to generate the µ-processor clock
  - Weakness -> bus cycles cannot be indefinitely stretched
- Clock stretching or system slowdown does not work with very slow I/O devices

Synchronous buses

- Example

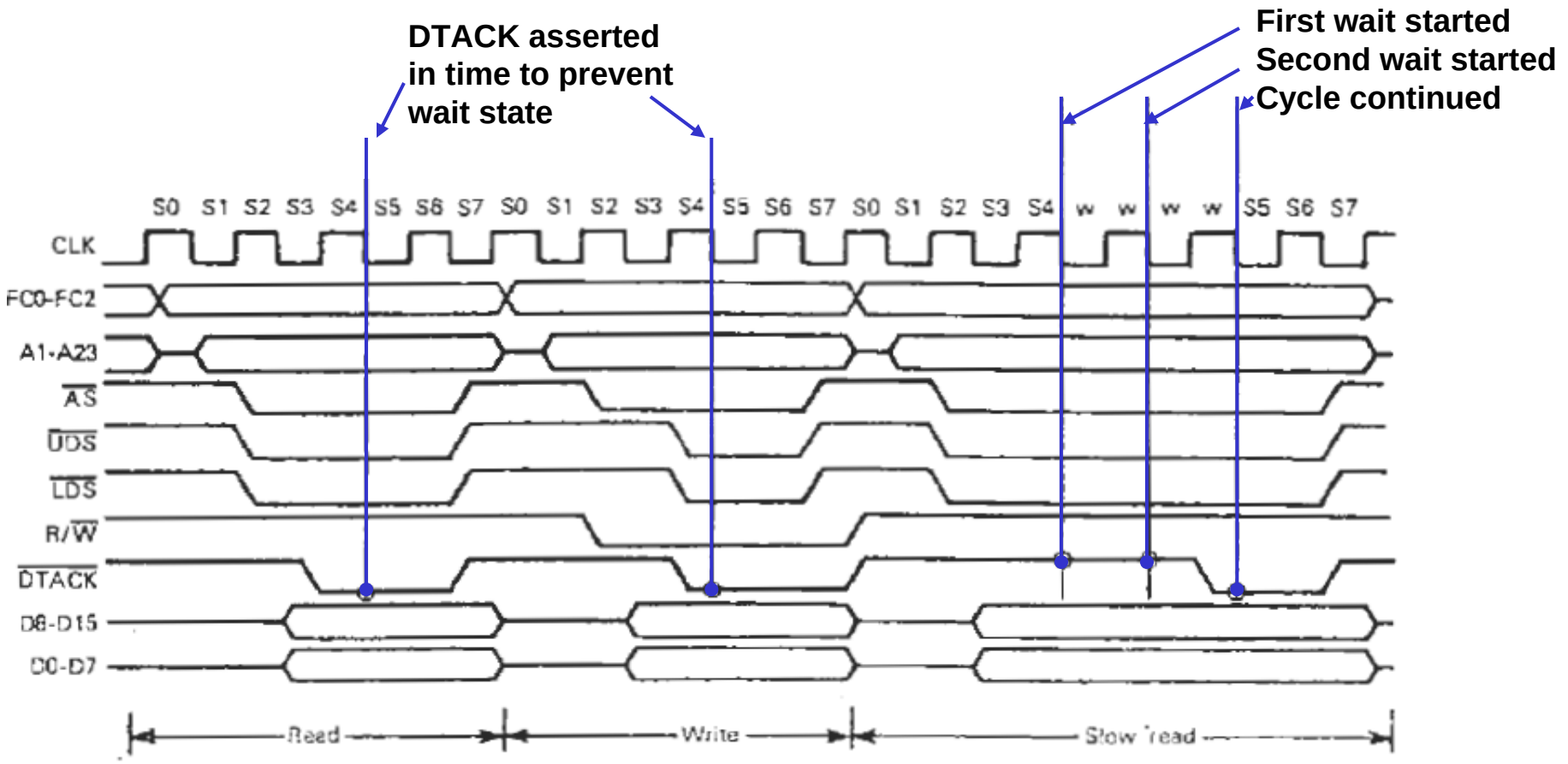


Asynchronous buses

- Require an acknowledge (ACK) signal from the addressed device on every bus cycle for operation to continue
- Useful for systems where CPU and I/O devices run at different speeds
- A bus time-out “watch-dog” is often used to detect run-away situations, causing execution of an error routine (usually non-maskable exception)

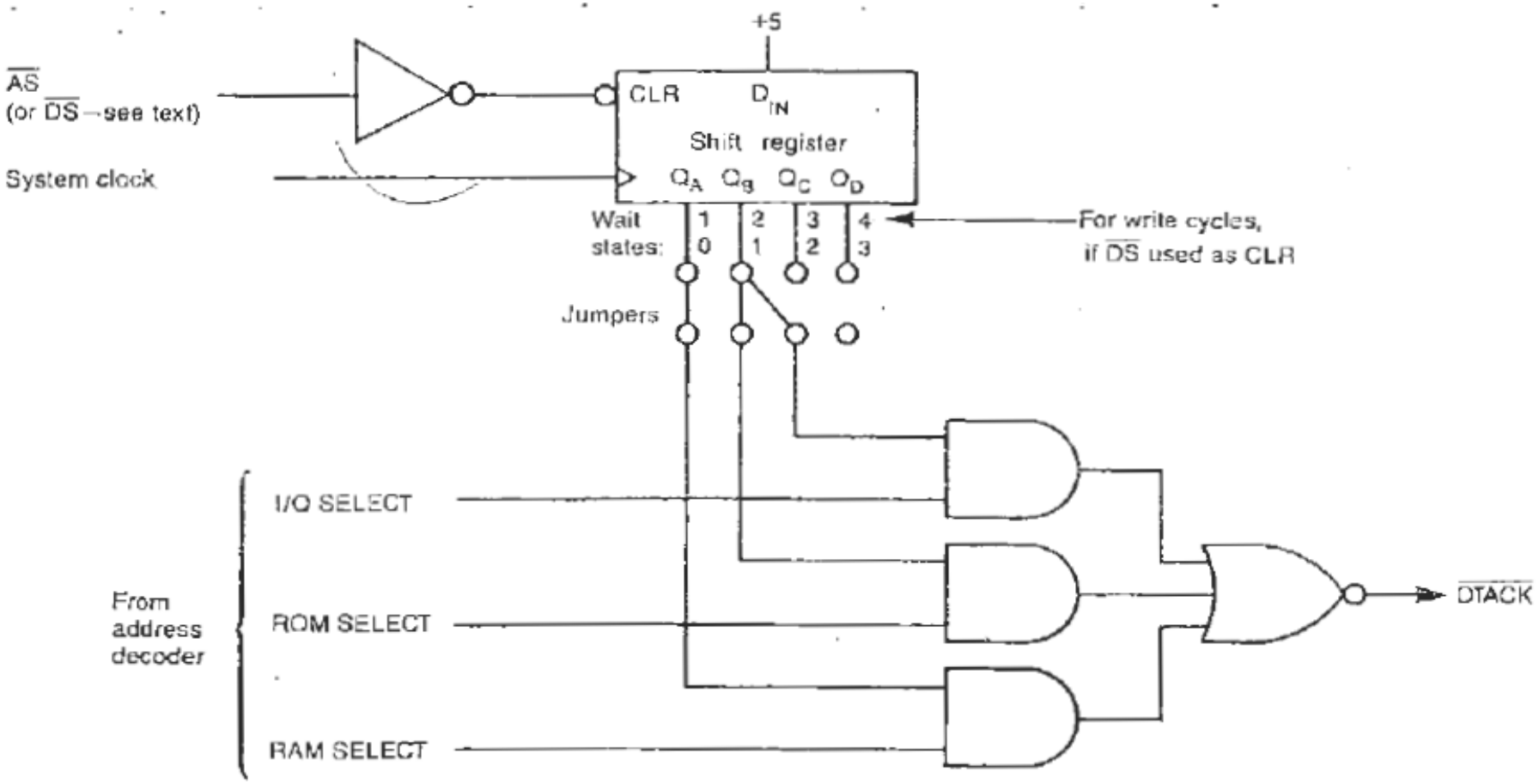
Asynchronous buses

- Example: Motorola's 68000



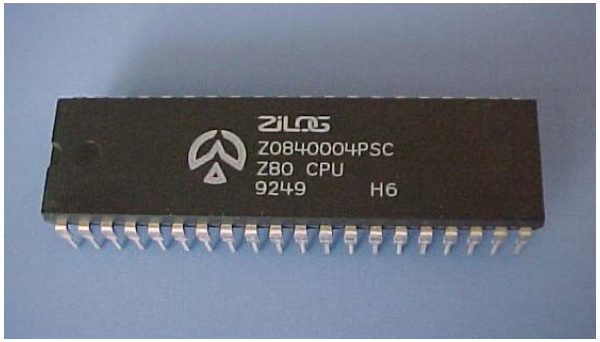
Asynchronous buses

- Basic circuit for 68000 $\overline{\text{DTACK}}$ generation



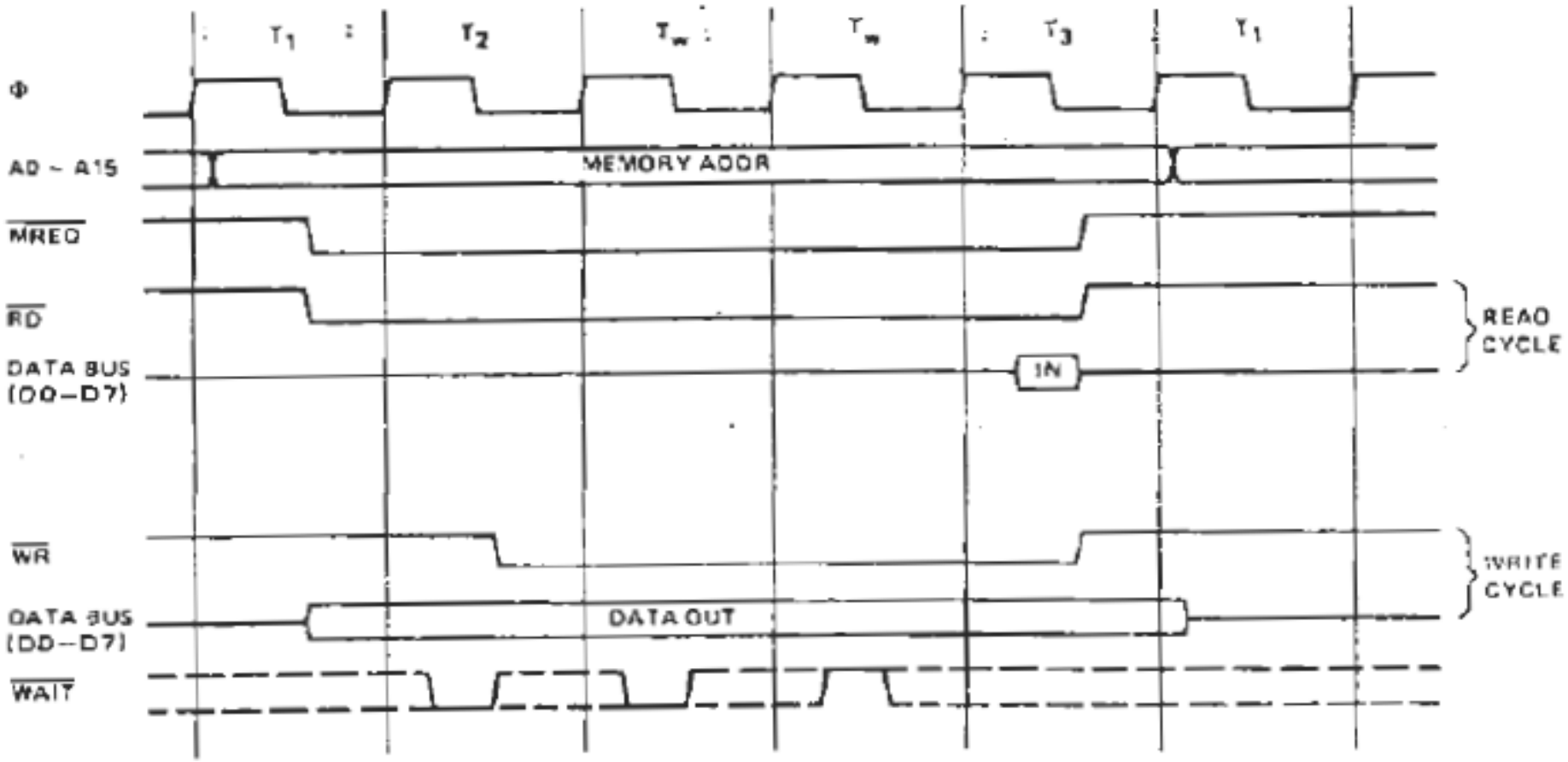
Semi-synchronous buses

- Most used approach to bus timing control!
- A “wait” input is asserted if the memory or peripheral needs to stretch the cycle
- Pros -> the bus cycle can be stretched indefinitely
- Examples: Zilog’s Z80, Intel’s 8085 and 8086 and derivatives



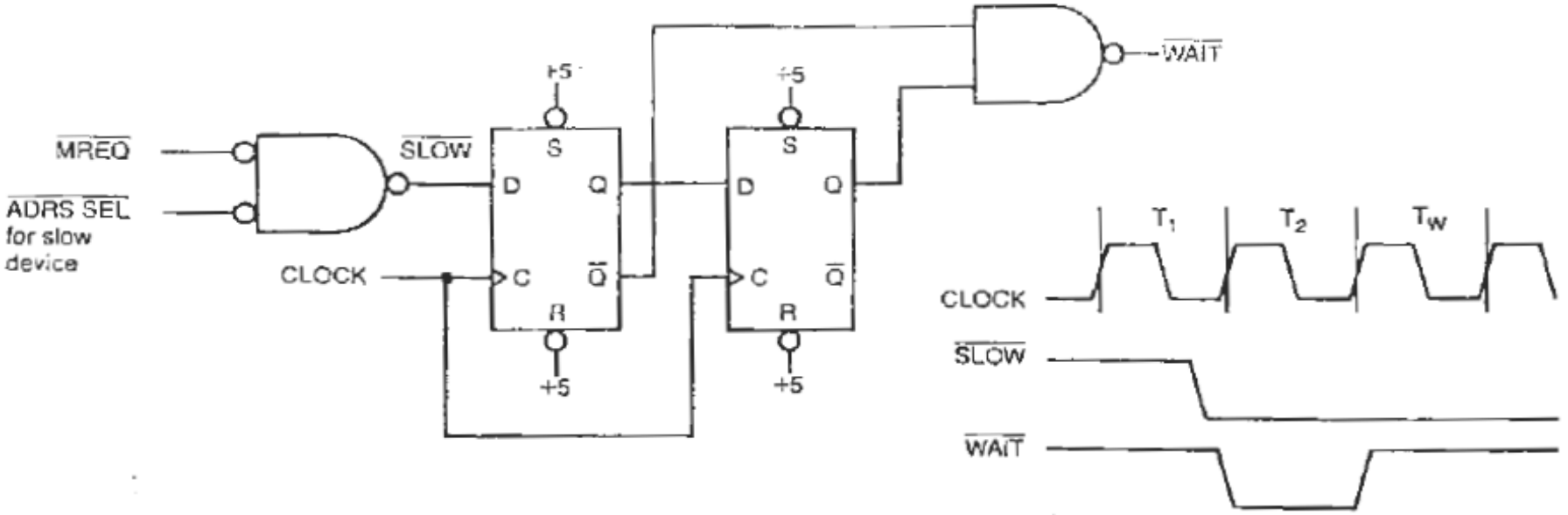
Semi-synchronous buses

- Z80 bus cycle with tow wait states



Semi-synchronous buses

- Z80 wait-state generator



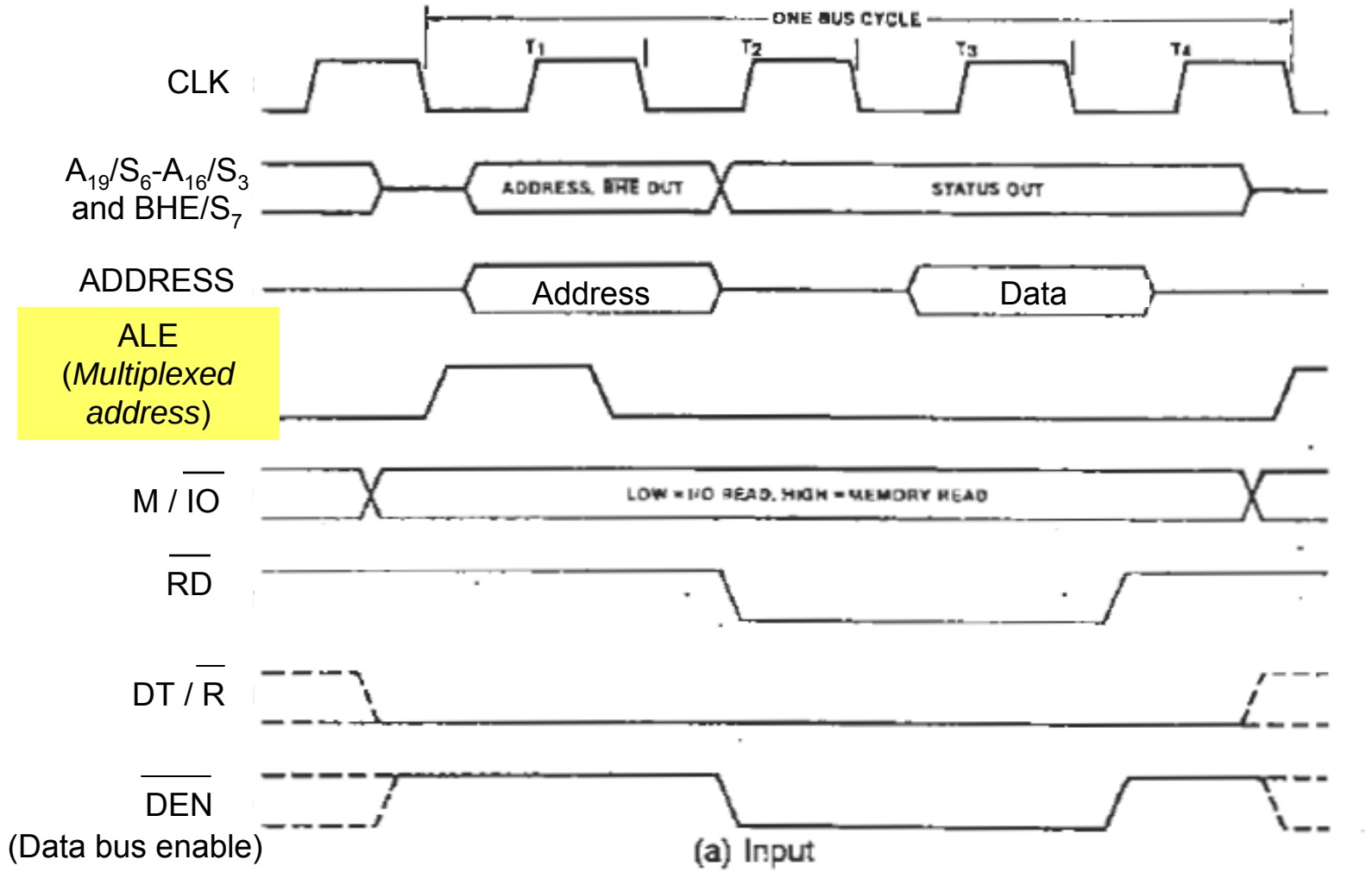
- Intel 8085 and 8086 provide a similar wait-state mechanism (signal's name is changed to READY)

Multiplexed Address/Data Buses

- Most of the pins of the microprocessor are used for the address and data buses.
- Multiplexing the address and data buses on the same pins reduces the number of pins required for the buses
- Saving pins allows the processor to provide additional interrupts and still fit in the package (e.g. 40-pin package)
- Nonmultiplexed buses are faster
- Most Intel microprocessors use multiplexed buses

Multiplexed Address/Data Buses

- Example Intel's 8086



Questions?


The newest calculator:
16 bit, with hi-tech monitor,
including mouse ...


It is not worth it - in six
months it will cost you half as much

References

- M. Slater. *MICROPROCESSOR-BASED DESING A COMPREHENSIVE GUIDE TO EFFECTIVE HARDWARE DESIGN*. Prentice Hall, 1989.
- http://www.csee.umbc.edu/~plusquel/310/slides/8086_chipset.html
- http://www.datasheetcatalog.com/motorola/51/
- http://www.intel.com/education/highered/Embedded/lectures/07_architecture.ppt
- http://courses.cs.tamu.edu/cpsc462/walker/Slides/68K_Hardware_Interfaces.ppt