CMPE - 310

Lecture 14 – Direct Memory Access Controller (8237)

Outline

Direct Memory Access Controller :8237

- Pinout
- Operation
- Interfacing

Direct Memory Access (DMA)

An alternative to the basic and interrupt-driven I/O discussed previously.

DMA allows data to be transferred between memory and the I/O device without processor intervention.

Speed of transfer limited to speed of memory components or DMA controller (up to 32-40 Mbytes/sec).

Common DMA operations:

- DRAM refresh
- Video refresh
- Disk-memory system reads and writes.

Two signals are used to request/ack a DMA transfer:

- HOLD is an input to the micro that requests a DMA action.
- HLDA is an output from the micro granting the DMA action.

The microprocessor responds by suspending the execution of the program and by placing its address, data and control bus in high-impedance states.

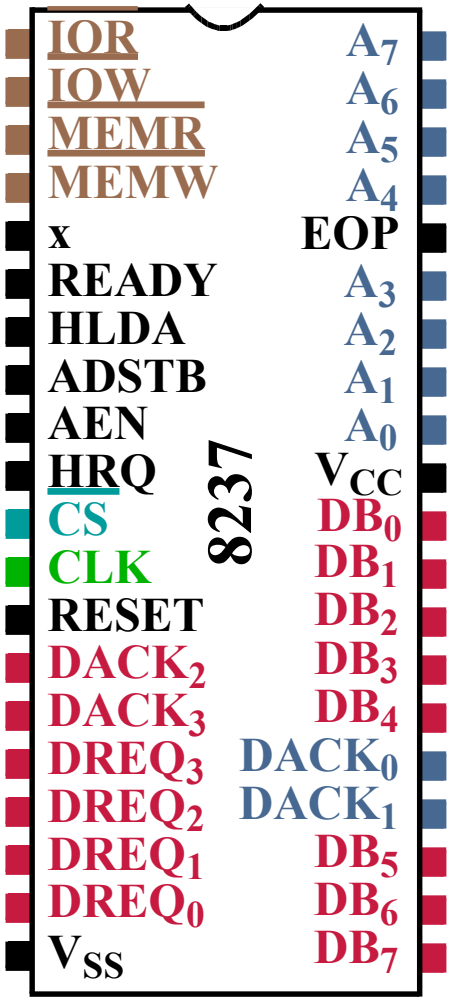
Direct Memory Access (DMA)

DMA 'reads' refer to transfers from memory to an I/O device and involves the use of $\overline{\text{MRDC}}$ and $\overline{\text{IOWC}}$.

DMA 'writes' refer to transfers from an I/O device to memory and involves the use of $\overline{\text{MWTC}}$ and $\overline{\text{IORC}}$.

The data transfer rate is determined by the speed of memory or the DMA controller (usually the latter).

The DMA controller provides memory with the address and select the appropriate I/O device (via $\overline{\text{DACK}}$).



DMA Controller: 8237

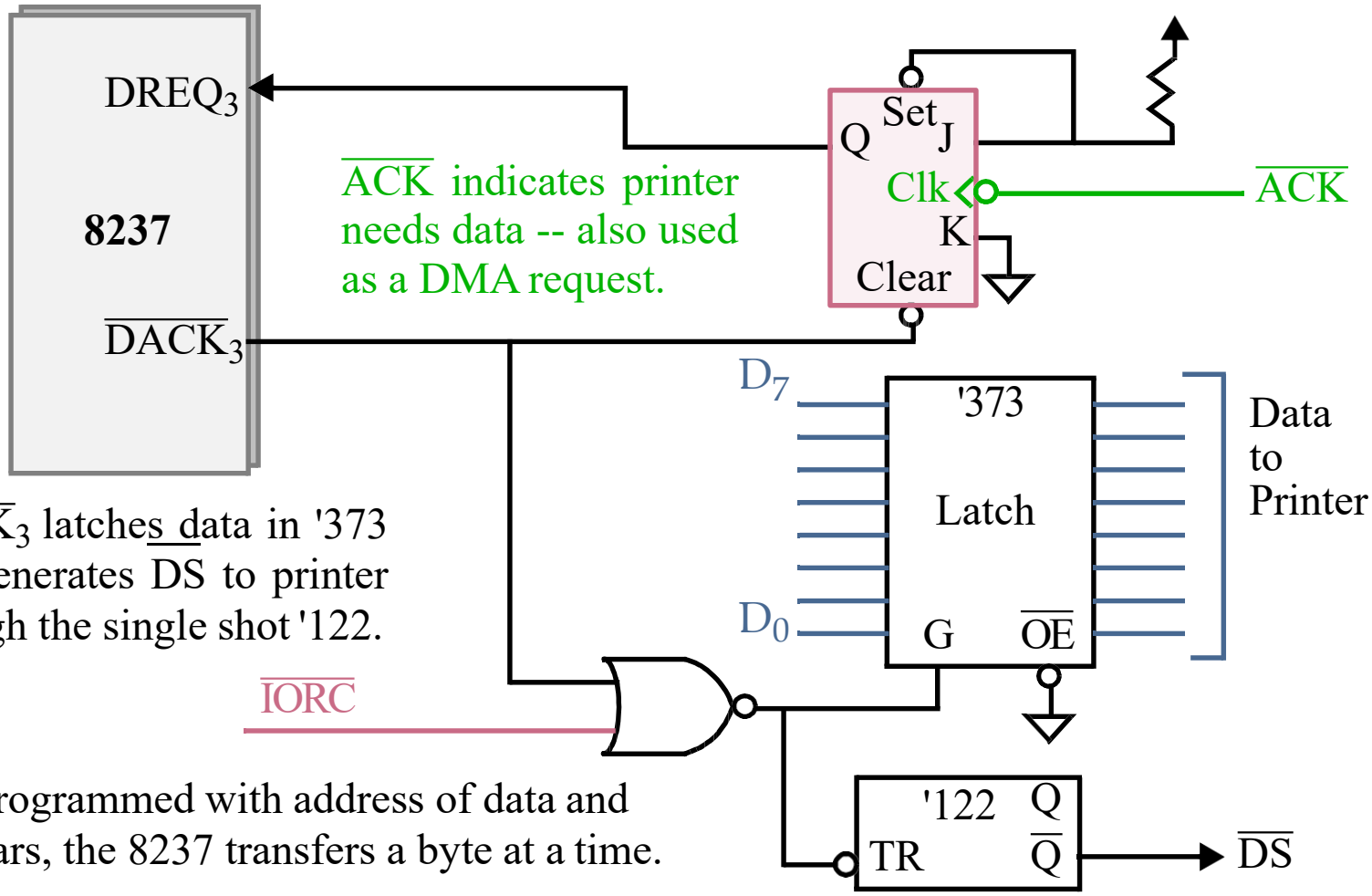
- Clk: < 5MHz.
- $\overline{CS}$: Output of a decoder.
- RESET: Clears all internal registers (command, status, request, etc.).
- READY: Allows memory and I/O to insert wait states into the 8237.
- HLDA: Input that tells 8237 that micro has released address, data and control buses.
- $DREQ_3$-$DREQ_0$: DMA request inputs used to request a DMA transfer.
- DB_7-DB_0: Used to program the 8237 and output upper 8-bits of address.
- $\overline{IOR}$, $\overline{IOW}$, $\overline{MEMR}$, $\overline{MEMW}$: Outputs used to control memory and I/O.
- $\overline{EOP}$: Bidirectional: as an input, used to terminate a DMA transfer, as an output, signals the end of the DMA transfer.
- A_3-A_0: Address pins select an internal register during programming and output part of the address for a transfer.
- A_7-A_4: Address outputs.
- HRQ: Output that connects to HOLD pin on micro to request a DMA.
- $DACK_3$ - $DACK_0$: Used to select an I/O device (ack a DMA request).
- AEN/ADSTB: Enable latch (and strobe) to transfer DB_x to upper 8 A bits.

DMA Controller: 8237

Some of the internal registers are:

- CAR_3 - CAR_0: Used to hold the 16-bit memory address used for a DMA transfer. Either incremented or decremented after a byte is transferred.
- $CWCR_3$ - $CWCR_0$: Current word count register programs a channel for the # of bytes (up to 64KB) transferred during a DMA action.
- CR: Command register programs the operation of the 8237. Bits in this register allow:
  - Memory-to-memory transfers (like MOVSB) where DMA channel 0 holds the source address and DMA channel 1 holds the dest address.
  - Memory-to-memory transfers in which DMA channel 0 holds a constant address -- used to fill a memory regions with a constant.
  - Fixed or rotating DMA channel priority, plus misc other options.
- MR: 'Mode of operation' register -- one for each channel. For example, block mode is used for memory-to-memory transfers.
- RR: Request register is used to request a DMA transfer via software -- essential for processor initiated memory-to-memory transfers.
- SR: Status register indicates when a DMA has completed.

DMA-Processed Printer Interface



DACK$_3$ latches data in '373 and generates DS to printer through the single shot '122.

Once programmed with address of data and # of chars, the 8237 transfers a byte at a time.

Note that the I/O device is NOT selected by decoding the address bus, but rather by DACK, since address bus contains a memory address.

See code in book and example of 8237 connected to an 8088.