ELEC 464 : Microcomputer System Design
Fall 1995

Review of the Motorola 68000 Processor Bus

This lecture covers the operation of the basic 68000 processor bus signals. The remaining CPU signals will be covered when we study exception processing and DMA. Read and write cycle timing will be covered in more detail in the next lecture.

Support Pins

Like any logic chip, the 68000 requires power supply and ground connections. Most processors use a 5 volt supply although 3 volt systems are becoming common because the lower supply voltage reduces power consumption.

Like other microprocessors, the 68000 requires a clock signal. On many processors the clock signal can be varied from DC to some maximum frequency but the 68000's clock must stay within certain bounds. The 8 MHz processor requires a clock between 4 and 8 MHz.

Almost all microprocessors include a reset input that can be used to initialize the processor to a known state. On the 68000 the RESET*[1] pin is also an open-drain output and the RESET instruction brings it low.

The HALT* pin is a bi-directional pin similar to RESET* and is used for single-stepping the processor and to indicate various error conditions.

Memory Interface

Most of the processor bus signals are devoted to the memory interface.

The 23-bit address bus (A01 to A23) outputs the address being read or written. Note that the processor always outputs an even address and uses UDS* and LDS* to select which byte(s) of that word address is/are being accessed. The AS* signal indicates that the address bus contents are valid. It is normally used to enable the outputs of address decoders.

The 16-bit bidirectional data bus (D00 to D15) inputs or outputs the data. The R/W* (read/write) output indicates whether the memory is to be read or written.

The LDS* and UDS* (lower and upper data strobes) outputs are used to strobe (clock) data into memory devices according to the type of memory access (lower byte, upper byte or word).

The DTACK* signal is driven by memory devices to tell the CPU that the data has been read or written. The CPU runs additional *wait states* until the memory circuit asserts DTACK*. The wait states give the memory enough time to complete the read or write operation.

Memory Interface Alternatives

Many processors combine the function of the data strobes (UDS* and/or LDS*) and R/W* into separate RD* and WR* strobes. Most processors have an A0 address line and so don't require separate upper and lower byte strobes. Some processors have a WAIT* input that is asserted by memory devices when they require wait states to be inserted rather than requiring DTACK*. Some processors have a separate I/O address space and use a signal to indicate whether the bus cycle is for I/O or memory.

Exercise: Draw a schematic that would convert DS* and R/W* signals into RD* and WR*.

Byte and Word Access

The 68000 can only access 16-bit words that are aligned on an even address. However, individual bytes can be read or written by accessing only the upper or lower byte. Recall that the 68000 is "big-endian" which means that the most significant byte (upper) is stored at the lowest (even) memory location.

Exercise: If the 68000 were to write a byte at address 33 (decimal) would UDS* or LDS* be asserted?

[1] Active-low signals will be denoted with an asterisk. RESET* means that the signal is true when the signal is at a low logic level.

Bus Cycles

A microprocessor operates by reading instructions from memory and executing them. The execution of an instruction may require further memory accesses. An instruction may also require additional clock cycles for internal processing (e.g. a multiply may require over 66 additional clock cycles to complete).

Each memory access involves a bus cycle during which the processor reads or writes to memory.

A memory read or write bus cycle requires at least 4 *clock cycles*. These 4 clock cycles can be considered as 8 bus states (S0 to S7). A read or write cycle may require more than 8 clock cycles if the memory requires wait states.

Exercise: At a nominal 8 MHz clock frequency how long are the following: a clock cycle? a bus state? a no-wait-state memory read cycle? a wait state? What is the shortest possible duration of an instruction?

Static RAM Operation

A review of how an SRAM is read and written should help explain the operation of the 68000 processor bus during memory read and write cycles.

A static memory chip (SRAM) memory is written by applying the desired address and data values, enabling CS* (chip select) and bringing the WE* (write enable) signal from high to low. The data is loaded into the memory on the falling *edge* of WE*.

The chip is read by applying the desired address and enabling CS* and OE* (output enable). The contents of the addressed memory location then appear on the data lines.

Write Cycle

During a write cycle the CPU outputs the desired address on address bus, sets R/W* low and puts the data on the data bus. After the address and data bus values are valid the CPU then brings DS* (UDS* and/or LDS*) low to write the data into memory. After the memory device indicates that it has completed the write operation (by bringing DTACK* low) the CPU raises DS* and the write operation is complete.

Read Cycle

During a read cycle the CPU outputs the desired address, sets R/W* high and asserts DS*. It waits until the memory asserts DTACK* to indicate that the data is valid and then it samples the data on the data bus as it brings DS* high.