# PART 1 The 68000 MPU

# 1 THE 68000 PROCESSOR

At the heart of the Sinclair QL there is a member of the Motorola 68000 family of processors; the Motorola 68008. From a software point of view the 68008 is a full 68000 implementation. Its major difference is that the device package is smaller, and only caters for an 8-bit data bus. An effect of this is that the actual throughput of the processor is reduced, due to overheads in memory addressing. This particular detail should not deter the QL assembly language programmer, who still has at his disposal one of the most powerful state-of-the-art 16/32-bit processors currently available. Also, the 68008 only has 20 of its maximum 32 address lines brought out to its package pins. This means that the addressing range is limited to 1 Megabyte (if you can call 1 Megabyte a 'limitation'!). Before going on to see how this processor may be used within the QL, let us look first at the general features of the 68000.

## 1.1 Operating modes

Two distinct operating modes are available with the 68000 processor. The two modes are called 'user' mode, and 'supervisor' mode. A flag in the status register will determine which state the processor is in at any one time. Certain instructions (e.g., STOP) cannot be executed while the 68000 is in user mode, and a privilege violation exception process will be initiated by the processor if such an execution is attempted.

When the processor is in user mode, the user stack pointer (USP) will be used by stack related operations. Conversely, the supervisor stack pointer (SSP) will be used when the processor is in supervisor mode.

## 1.2 68000 registers

The 68000 has eighteen 32-bit registers and one 16-bit status register (see Fig.1.1). The 32-bit register set is divided up into eight data registers, seven address registers, two stack pointers, and a program counter.

## DATA REGISTERS

The eight data registers are labelled D0 to D7. Data operations using these registers may be bit, BCD (nibble), byte, word (16-bit), or long-word (32-bit) orientated. Within instructions that permit a data register to be one or more of its operands, any data register may be used. In effect this means that any data register may be used as an accumulator, index register, general purpose register, or loop counter. This is an extremely flexible approach to processor register allocation, and is one of the reasons why the 68000 is so easy to program efficiently.

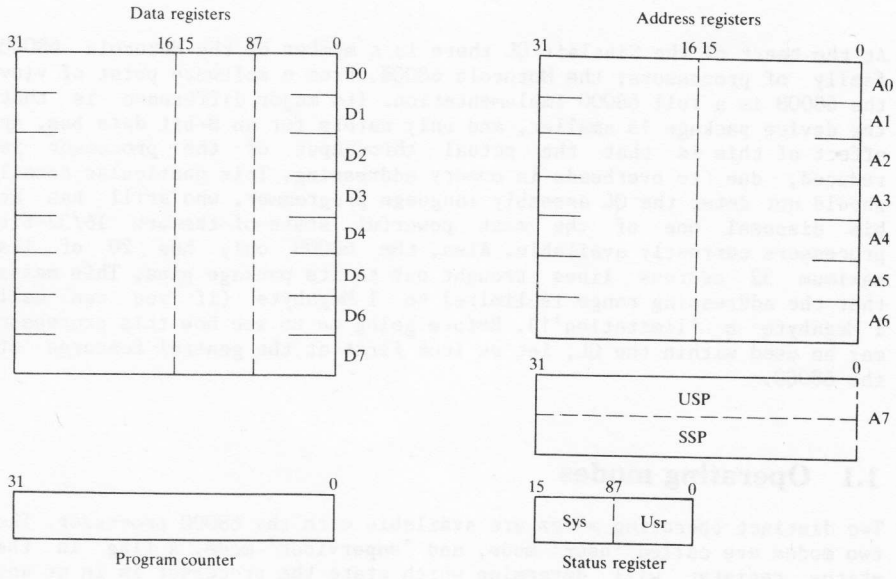


**Figure 1.1 68000 internal registers**

## ADDRESS REGISTERS AND STACK POINTERS

The seven address registers are labelled A0 to A6. The two stack pointers are also treated as address registers and are both labelled A7. Alternative mnemonics for the two stack pointers are USP (user stack pointer) and SSP (supervisor stack pointer). A flag in the status register will determine which state the processor is in (i.e., user or supervisor) and the respective stack pointer will be used accordingly. Operations using the address registers are limited to the types word (16-bit) and long-word (32-bit). In other words, address registers cannot be the source or destination for bit, byte, or logical operations. If an address register is the destination operand, the operation will always be long-word, and the source will be sign extended, if necessary, before use. The address registers are normally used for manipulating and holding addresses rather than data. They may be used also as index registers.

Note that, because the stack pointers are in fact the address register

A7, any legal addressing mode for instructions using address registers will also be legal for the stack pointers. This means that stack pointer register addressing modes for the 68000 are much more versatile than for many other processors.

## STATUS REGISTER

The 68000 status register is a 16-bit register split into two distinct bytes. The two bytes correspond to the system status byte (bits 8 to 15) and the user status byte (bits 0 to 7). The system status byte can only be modified when the processor is in supervisor mode. Two mnemonics are allocated to the status register. First, there is CCR, and this refers to the low order user byte of condition codes. Any instruction using this mnemonic will refer to eight bits of data only. Second, there is SR, and this refers to the whole status register. Any instruction using this mnemonic, in order to modify the contents of the status register, will only be executed if the 68000 is in supervisor mode.

Figure 1.2 shows the allocation of flags within the status word. The Carry (C), Zero (Z), Negative (N), and Overflow (V) bits are standard condition flags. There is also an Extend (X) bit flag which is always set to the same state as the Carry flag, if it is affected by any particular instruction. The Extend flag is used for multi-precision arithmetic operations. Chapter 2 describes the relevance of these flags for instructions.

The three least significant bits of the system status byte are used as the interrupt disable mask (IDM) for the 68000. Seven prioritized levels of interrupt are catered for, and each priority interrupt causes execution to be routed through an interrupt vector. The mask in the status register specifies the range of interrupts which are to be ignored. If, for example, the mask is set to binary 011 (3), interrupts 1 to 3 will be ignored by the processor.

Note that the 68008 only has three levels of interrupt (i.e., 2, 5, and 7). On the QL, a level 5 interrupt is transitory and will always generate a level 7 interrupt (non-maskable) within 20 ms.

The Supervisor flag (S) determines whether or not the 68000 is running in supervisor mode. If the bit is set (i.e., 1), the processor will be in supervisor mode. Last, but far from least, is the Trace (T) flag. This flag enables the processor to be run in single-step mode, and permits system debuggers to obtain control over instruction execution.

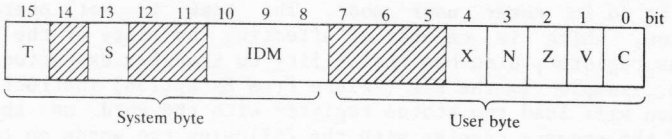


**Figure 1.2 68000 status register**

## 1.3 Use of memory

Up to 1 Megabyte of memory may be directly accessed by the 68008. A 20-bit address bus is required to address this amount of memory. Addresses can, therefore, be represented by five digit hexadecimal numbers in the range $00000 to $FFFFF. To access a byte of data in memory, any one of the possible addresses may be used. Accessing a word (i.e., 16 bits) or a long-word (i.e., 32 bits) of memory is a little more restrictive. Words or long-words can only be addressed at even addresses; that is, $00000, $00002, and so on up to $FFFFE.

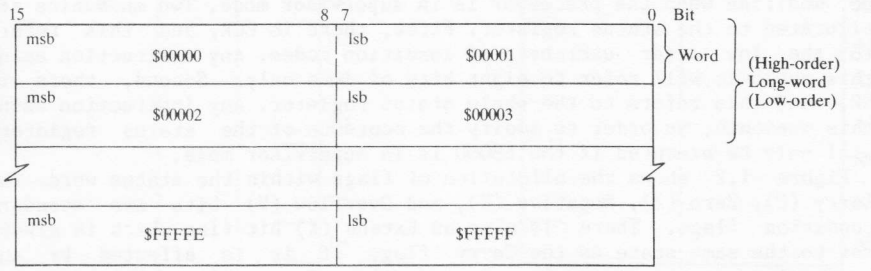


**Figure 1.3 Memory usage**

In the case of word addressing the most significant byte of the word will be found in the even address, and the least significant byte of the word will be found in the following odd address (see Fig.1.3). Long-word addressing is similar to word addressing in that it involves the equivalent of two word accesses. The high-order word of the long-word will be found first, followed by the low-order word of the long-word.

## 1.4 Moving between supervisor and user modes

To know that the 68000 has two modes of operation is not particularly useful unless you know how to swap between them. When the 68000 is reset (e.g., at power-on) the bottom eight bytes of memory are loaded into the supervisor stack pointer and the program counter, and instruction execution commences in supervisor mode.

Because we start off in supervisor mode the first thing we need to know how to do is enter user mode. The task is not onerous! Any instruction, which is capable of affecting the state of the S flag in the status register, also has the ability to transfer execution to user mode. An example is the RTE (ReTurn from Exception) instruction. This instruction will load the status register with the word on the stack, and load the program counter with the following two words on the stack. If the word loaded into the status register reset the S flag, user mode will be entered. If the S flag remains set, supervisor mode will continue.

Once you are in user mode, the method of getting back into supervisor mode is to cause some form of exception processing. Exception processing will occur under a number of conditions:

1. Addressing violation. A word or long-word was addressed on an odd byte boundary.
2. Privileged instruction violation. A privileged instruction was executed.
3. Illegal or unimplemented opcode. The instruction executed was not a legal instruction.
4. TRAP instruction execution. All TRAP instructions are treated as internal exceptions.
5. TRAPV, CHK, DIVS, DIVU exception error condition has occurred (e.g., divide by zero).
6. Trace active. If the T flag in the status register is set, exception processing will be performed after each user instruction is executed.
7. External interrupt request. One of seven (n.b., three on the QL) prioritized interrupts has been received.
8. Reset. The 68000 processor has physically been reset.
9. Bus error. An error has occurred on the physical address/data bus of the 68000 processor.

The last two of these exception processes (reset and bus error) are clearly not of much use to the applications programmer! The most common way of entering supervisor mode from a program is through the use of a TRAP instruction.