Processor mode		Description
User	usr	Normal program execution mode
FIQ	fiq	Fast Interrupt for high-speed data transfer
IRQ	irq	Used for general-purpose interrupt handling
Supervisor	svc	A protected mode for the operating system
Abort	abt	Implements virtual memory and/or memory protection
Undefined	und	Supports software emulation of hardware coprocessors
System	sys	Runs privileged operating system tasks

Table 3.1: ARM processor modes

The modes other than User mode are known as *privileged modes*. They have full access to system resources and can change mode freely. Five of them are known as *exception modes*: FIQ (Fast Interrupt), IRQ (Interrupt), Supervisor, Abort, and Undefined. These are entered when specific exceptions occur. Each of them has some additional registers to avoid corrupting User mode state when the exception occurs (see 3.2 for details).

The remaining mode is System mode, it is not entered by any exception and has exactly the same registers available as User mode. However, it is a privileged mode and is therefore not subject to the User mode restrictions. It is intended for use by operating system tasks which need access to system resources, but wish to avoid using the additional registers associated with the exception modes. Avoiding such use ensures that the task state is not corrupted by the occurrence of any exception.

3.2 Registers

The ARM has a total of 37 registers. These comprise 30 general purpose registers, 6 status registers and a program counter. Figure 3.2 illustrates the registers of the ARM. Only fifteen of the general purpose registers are available at any one time depending on the processor mode.

There are a standard set of eight general purpose registers that are always available (R*0* – R*7*) no matter which mode the processor is in. These registers are truly general-purpose, with no special uses being placed on them by the processors' architecture.

A few registers (R*8* – R*12*) are common to all processor modes with the exception of the fiq mode. This means that to all intent and purpose these are general registers and have no special use. However, when the processor is in the fast interrupt mode these registers and replaced with different set of registers (R*8_fiq* - R*12_fiq*). Although the processor does not give any special purpose to these registers they can be used to hold information between fast interrupts. You can consider they to be `static` registers. The idea is that you can make a fast interrupt even faster by holding information in these registers.

The general purpose registers can be used to handle 8-bit bytes, 16-bit half-words[1], or 32-bit words. When we use a 32-bit register in a byte instruction only the least significant 8 bits are used. In a half-word instruction only the least significant 16 bits are used. Figure 3.3 demonstrates this.

The remaining registers (R*13* – R*15*) are special purpose registers and have very specific roles: R*13* is also known as the Stack Pointer, while R*14* is known as the Link Register, and R*15* is the Program Counter. The "user" (usr) and "System" (sys) modes share the same registers. The exception modes all have their own version of these registers. Making a reference to register R*14* will assume you are referring to the register for the current processor mode. If you wish to refer

[1] Although the ARM does allow for Half-Word instructions, the emulator we are using does not.