Common Programming Models

Murrough Landon – 6 July 2000

Aims

- Reasonably coherent programming models for all of our modules
- Similar notation used in documentation of programming models
- Reduce complexity and variation
- Easier life for small group of software writers!

Guidelines

- All registers are readable (no write only registers). The register bits should have the same meaning on read and write accesses.
  - All status registers shall be read only
  - All control registers shall be read/write
  - Reading back a register will generally return the last value written (there may be some write only reset bits which read back zero)
- Attempts to write to read only registers or to undefined portions of registers will result in the unmodifiable fields being left unchanged.
- It is illegal to write a value to a register which the module itself is able to modify at the same time. [This may be a bit strong!].

- When the address space occupied by the module is accessed, it will always respond with a handshake to avoid a bus error.
- The power-up condition of all registers will be zero unless otherwise stated in the documentation.

Notation

- A **byte** is always an 8 bit field; a **word** is always a 16 bit field; a **longword** is always a 32 bit field.
- All offsets specified as byte offsets.
- Setting a bit-field means writing a 1 to it, clearing it means writing a 0.
- RO = read only; RW = read/write.

Module ID

- Common module ID format for all modules. The DSS defined a 32 bit format, ie two 16 bit registers:
  - High word: module type
  - Low word bits 0-7: serial number (1-255)
  - Low word bits 8-11: revision (1-15)
  - Low word bits 12-15: unused, read as zero
- Traditionally located at the module base address (convenient for scanning VME bus).