## Programming Model – with 32 bit indexing enabled

|  |  |  |  |
| --- | --- | --- | --- |
| 31 0 | | | |
| X – Index Register | | | |
| Y – Index Register | | | |
| U – User stack pointer | | | |
| S – Hardware stack pointer | | | |
| PC | | | |
| A | B | E | F |

without 32 bit indexing

|  |  |  |  |
| --- | --- | --- | --- |
| 31 16 15 0 | | | |
| 0000 | | X – Index Register | |
| 0000 | | Y – Index Register | |
| 0000 | | U – User stack pointer | |
| 0000 | | S – Hardware stack pointer | |
| PC | | | |
| A | B | E | F |

A,B registers concatenate to form D register

A,B,E,F registers concatenated form the Q register

The E,F registers may be transferred or exchanged with the A, and B registers.

## Instruction Prefixes

### FAR

FAR when applied to extended addressing indicates to use a full 32 bit address rather than a 16 bit one. The FAR prefix works regardless of the index size.

When the FAR prefix is applied to indirect addressing the prefix indicates that the indirect address is 32 bit. This allows the use of a 32 bit indirect address to reach anywhere in memory.

Opcode: 0x15

### POST

The POST prefix indicates that the index register is applied after retrieving an indirect address. Normally the index register is used in the calculation of the indirect address.

Opcode: 0x1B

## Additional Instructions

JMP FAR – performs a jump using a 32 bit extended address.

Opcode: 0x8F

JSR FAR – performs a jump to subroutine using a 32 bit extended address. The full 32 bit program counter is stored on the stack.

Opcode: 0xCF

RTF – performs a far return from subroutine by loading a full 32 bit program counter from the stack.

Opcode: 0x38

LDMD – loads the mode register

bit [2] when set indicates to use 32 bit index registers

bit [1] indicates that a Firq should stack all registers

Opcode: 0x113D

Without 32 bit indexing enabled, the stacks must reside within the first 64k bank of memory.

Indirect addresses must reside within the first 64k bank of memory.

## Differences from the 6809

The program counter is a full 32 bit register. The JMP and JSR instructions modify only the low order 16 bits of the program counter. To modify the full 32 bits use the JMP FAR and JSR FAR instructions.

During interrupt processing the entire 32 bit program counter is stacked. The RTI instruction also loads the entire 32 bit program counter.

### Hardware:

This is a softcore implementation of a 6809 compatible processor. As such no attempt was made to duplicate the 6809’s bus cycle activity. Instructions may not execute in the same number of clock cycles as the 6809. Also the core uses a 32 bit WISHBONE bus to interface to the system. Occasionally the core may make use of 32 bit transfers if the address is appropriately aligned.