SCM68000
revision 1

Applications Information
Low Power on the SCM68000 (EC000 Core)

The SCM68000 (EC000 core) has been redesigned to provide fully static and low power operation. This document describes the recommended method for placing the SCM68000 into a low-power mode to reduce the power consumption to its quiescent value[1] while maintaining the internal state of the processor. The low-power mode described below will be routinely tested as part of the SCM68000 test vectors provided by Motorola.

NOTE

The terms *assertion* and *negation* are used in this document to avoid confusion when describing a mixture of "active-low" and "active-high" signals. The term *assert* or *assertion* is used to indicate that a signal is active or true, independently of whether that level is represented by a high or low voltage. The term *negate* or *negation* is used to indicate that a signal is inactive or false. The names of all "active-low" signals end with the letter *B*.

To successfully enter the low-power mode, the SCM68000 must be in the supervisor mode. A recommended method for entering the low-power mode is by using the TRAP instruction which causes the processor to begin exception processing, thus entering the supervisor mode. The following steps during the trap routine should be accomplished by external circuitry:

1.) Externally detect a write to the low-power address. This address should be chosen by the user and can be any address in the 4 Gbyte addressing range of the SCM68000. A write to the low-power address can be detected by polling A31–A0, RWB, and FC2–FC0. When the low-power address is detected, RWB is a logic low, and the function codes have a five (101) on their output, then the processor is writing to the low-power address in supervisor mode and user-designed circuitry should assert the ADDRESS_MATCH signal shown in Figure 1 and Figure 2.

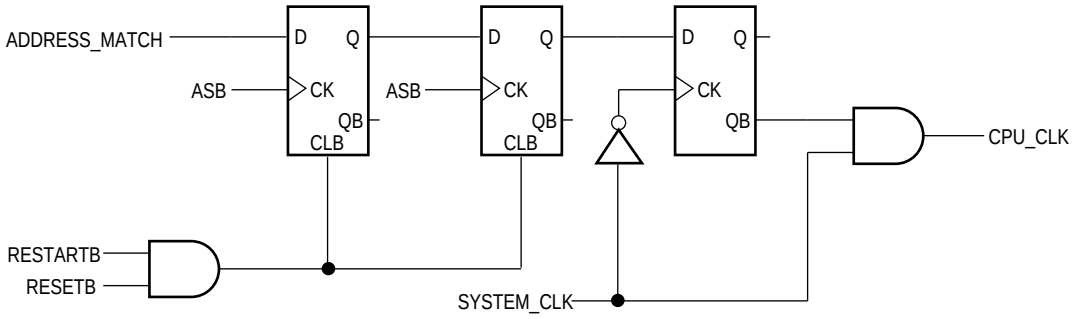

Figure 1. Low-Power Circuitry for 16-bit data bus

[1] The preliminary specification for the SCM68000's current drain while in the low-power mode is Idd < 10µA.

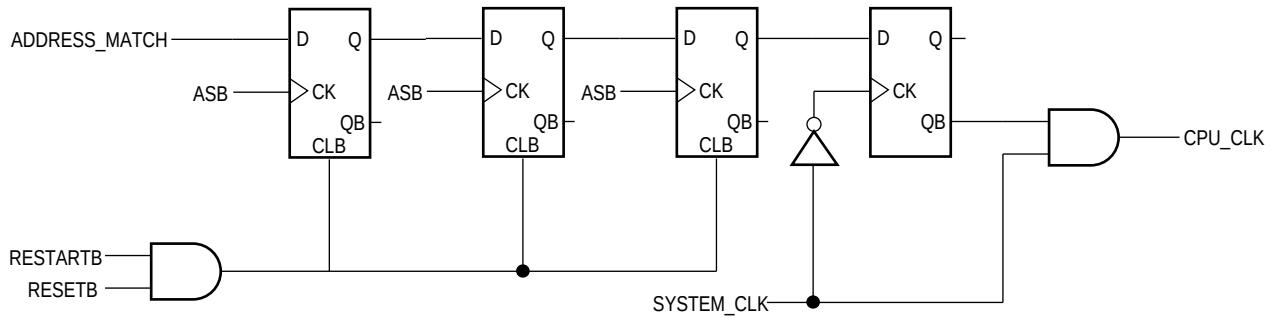

Figure 2. Low-Power Circuitry for 8-bit data bus

2.) Execute the STOP instruction. The external circuitry shown in Figure 1 and Figure 2 will count the number of bus cycles starting with the write to the low-power address and will stop the processor's clock on the first falling edge of the system clock after the bus cycle that reads the immediate data of the STOP instruction. Figure 2 has one more flip-flop than Figure 1 because the SCM68000 in 8-bit mode requires two bus cycles to fetch the immediate data of the STOP instruction. After the processor's clock is disabled it is often desirable to disable the clock to other sections of the user's circuit. This can be done, but care must be taken to ensure that runt clocks and spurious glitches are not presented to the SCM68000. A timing diagram is shown in Figure 3.

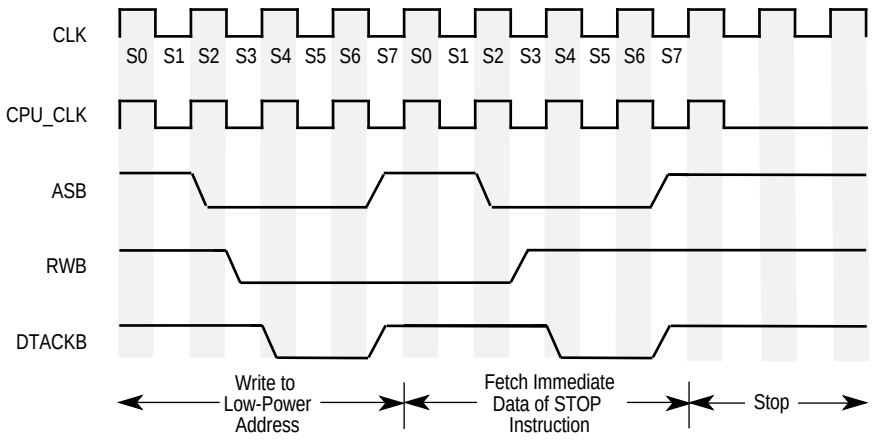

Figure 3. Clock Stop Timing for 16-bit Data Bus

NOTE

While the SCM68000 is in the low-power mode, all inputs must be driven to VDD or VSS, or have a pull-up or pull-down resistor.

3.) This step is optional depending on whether the user's applications require the three-stateable signals of the SCM68000 to be put into a high-impedance state. To put the SCM68000 into a three-state condition, the proper method for arbitrating the bus as described in **3.2 Bus Arbitration** in the *EC000 Core Processor (SCM68000) User's Manual* should be completed during the fetch of the status register data for the STOP instruction. A timing diagram with the bus arbitration sequence is shown in Figure 4.

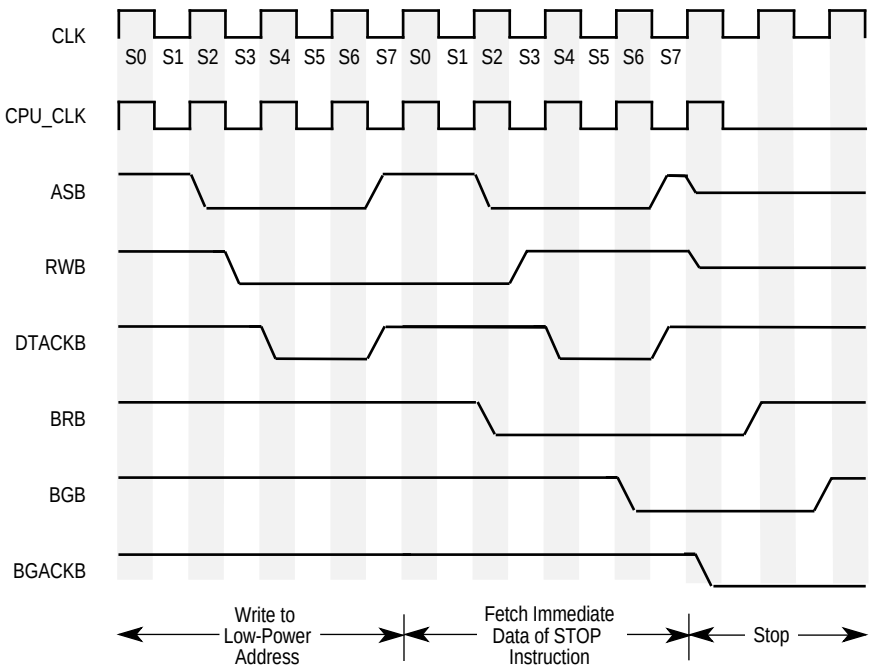

Figure 4. Clock Stop Timing with Bus Arbitration for 16-bit Data Bus

After the previous steps are completed, the SCM68000 will remain in the low-power mode until the appropriate interrupt is recognized. External logic will also have to poll IPLB2–IPLB0 to detect the proper interrupt. When the correct interrupt level is received, the following steps will bring the processor out of the low-power mode:

1.) Restart the system clock if it was stopped.

2.) Wait for the system clock to become stable.

3.) Assert the RESTARTB signal. This will cause the processor's clock to start on the next falling edge of the system clock. Figure 5 shows the timing for bringing the processor out of the low power mode. Both the RESTARTB and RESETB signals are subject to the asynchronous setup time as specified in **Section 7 Electrical Characteristics** in the *EC000 Core Processor (SCM68000) User's Manual*.

WARNING

The system clock must be stable before the RESTARTB signal is asserted to prevent glitches in the clock. An unstable clock may cause unpredictable results in the SCM68000.

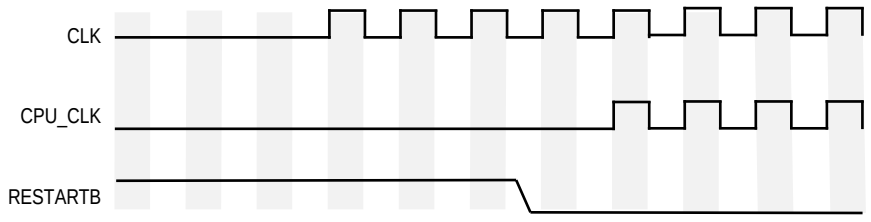

Figure 5. Clock Start Timing

4.) If the SCM68000 was put into a three-state condition the BGACKB signal (used for 3-wire bus arbitration) or the BRB signal (used for 2-wire bus arbitration) must be negated before the processor can begin executing instructions.

An example trap routine follows:

```
TRAP_x   MOVE.B #0,$low_power_address   /* Write that causes ADDRESS_MATCH to assert */
         STOP #$2000                    /* STOP instruction with desired interrupt mask */
         RTE                            /* Return from the exception */
```

The first instruction (MOVE.B #0,$low_power_address) writes a byte to the low-power address which will cause the external circuitry to begin the sequence that will stop the processor's clock. The second instruction (STOP #$2000) is the STOP instruction that loads the SR with the immediate data. This allows the user to set the interrupt that will cause the processor to come out of the low-power mode. The final instruction (RTE) instructs the processor to return from the exception and resume normal processing.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912, Arizona 85036.
EUROPE: Motorola Ltd.; European Literature Centre; 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.
JAPAN: Nippon Motorola Ltd.; 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141 Japan.
ASIA-PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Center, No. 2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.