

# CMOS 8/16-BIT MICROPROCESSOR FAMILY

# **Features**

- Advanced CMOS design for low power consumption and increased noise immunity
- Emulation mode for total software compatibility with 6502 designs
- Full 16-bit ALU, Accumulator, Stack Pointer, and Index Registers
- Direct register for "zero page" addressing
- 24 addressing modes (including 13 original 6502 modes)
- Wait for Interrupt (WAI) and Stop the Clock (STP) instructions for reduced power consumption and decreased interrupt latency
- 91 instructions with 255 opcodes
- Co-Processor (COP) instruction and associated vector
- Powerful Block Move instructions

- Full 16-bit operation with 24 address lines for the 16 MBvte memory
- Program selectable 8-bit mode for 6502 coding compatibility Valid Program Address (VPA) and Valid Data Address (VDA) outputs for dual cache and DMA cycle steal implementation
- Vector Pull (VP) output indicates when interrupt vectors are being fetched. May be used for vectoring/prioritizing interrupts
- Abort interrupt and associated vector for interrupting any instruction without modifying internal registers
- Memory Lock (<u>ML</u>) for multiprocessor system implementation
- 40-pin DIP, 44-pin PLCC

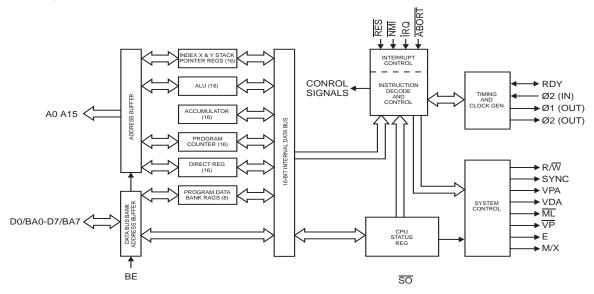
Contact factory for complete data sheet.

### **Product Description**

The G65SC816 is an advanced CMOS 16-bit microprocessor featuring total software compatibility with 8-bit NMOS and CMOS 6500 series microprocessors. The G65SC816 provides 24 address lines for 16 Mbyte addressing, while providing both 8-bit and 16-bit operation.

This microprocessor contains an Emulation (E) mode for emulating 8-bit NMOS and CMOS 6500 Series microprocessors. A software switch determines whether the processor is in the 8-bit emulation mode or in the native 16-bit mode. This allows existing 8-bit system designs to use the many powerful features of the G65SC816.

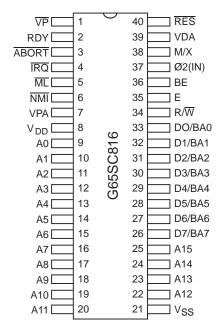
## **Block Diagram**

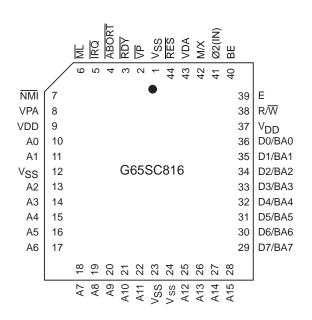


© 1987, 1996 CMD Corp. All rights reserved

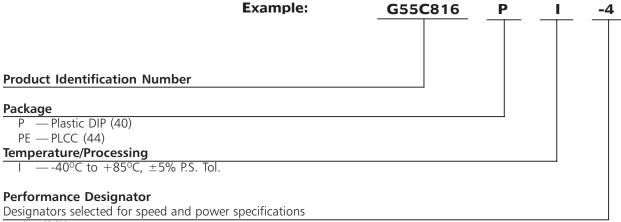


### **Pin Assignments**





# **Ordering Information**



<u>4 4MHz</u>

— 5 5MHz