## SGS-THOMSON MICROELECTRONICS

# EF6805U3

# 8-BIT MICROCOMPUTER UNIT

## HARDWARE FEATURES

- 32 TTL/CMOS COMPATIBLE I/O LINES
- 24 BIDIRECTIONAL (8 lines are LED compatible)
- 8 INPUT-ONLY
- 3776 BYTES OF USER ROM
- 112 BYTES OF RAM
- SELF-CHECK MODE
- ZERO-CROSSING DETECT/INTERRUPT
- INTERNAL 8-BIT TIMER WITH 7-BIT SOFTWARE PROGRAMMABLE PRESCALER AND CLOCK SOURCE
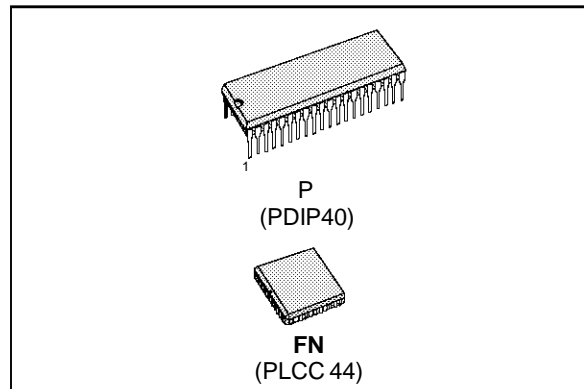- 5V SINGLE SUPPLY

## SOFTWARE FEATURES

- 10 POWERFUL ADDRESSING MODES
- BYTE EFFICIENT INSTRUCTION SET WITH TRUE BIT MANIPULATION, BIT TEST, AND BRANCH INSTRUCTIONS
- SINGLE INSTRUCTION MEMORY EXAMINE/CHANGE
- POWERFUL INDEXED ADDRESSING FOR TABLES
- FULL SET OF CONDITIONAL BRANCHES
- MEMORY USABLE AS REGISTER/FLAGS
- COMPLETE DEVELOPMENT SYSTEM SUPPORT ON INICE

## USER SELECTABLE OPTIONS

- 8 BIDIRECTIONAL I/O LINES WITH TTL OR TTL/CMOS INTERFACE OPTION
- 8 BIDIRECTIONAL I/O LINES WITH TTL OR OPEN-DRAIN INTERFACE OPTION
- CRYSTAL OR LOW-COST RESISTOR OSCILLATOR OPTION
- LOW VOLTAGE INHIBIT OPTION
- VECTORED INTERRUPTS : TIMER, SOFTWARE, AND EXTERNAL
- USER CALLABLE SELF-CHECK SUBROUTINES

## DESCRIPTION

The EF6805U3 Microcomputer Unit (MCU) is a member of the 6805 Family of low-cost single-chip Microcomputers. The 8-bit microcomputer contains a CPU, on-chip CLOCK, ROM, RAM, I/O, and TIMER. It is designed for the user who needs an economical microcomputer with the proven capabilities of the 6800-based instruction set. A comparison of the key features of several members of the 6805 Family of Microcomputers is shown at the end of this data sheet. The following are some of the hardware and software highlights of the EF6805U3 MCU.
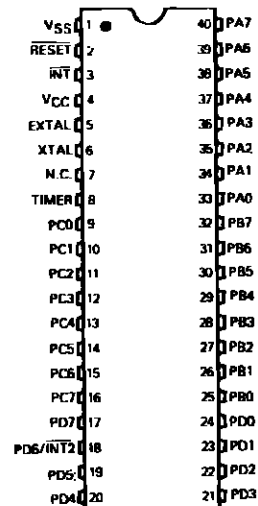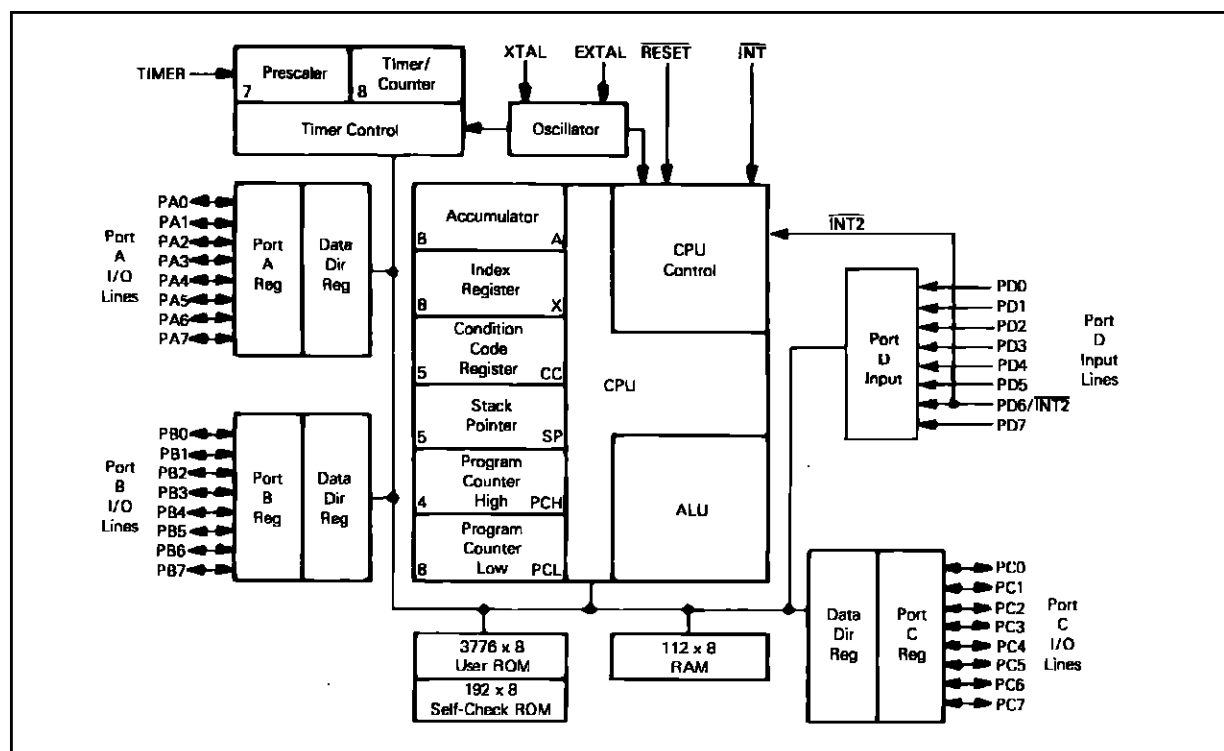


**P**
(PDIP40)

**FN**
(PLCC 44)



### PIN CONNECTIONS

| | | |
|---|---|---|
| V$_{SS}$ | 1 | 40 PA7 |
| RESET | 2 | 39 PA6 |
| INT | 3 | 38 PA5 |
| V$_{CC}$ | 4 | 37 PA4 |
| EXTAL | 5 | 36 PA3 |
| XTAL | 6 | 35 PA2 |
| N.C. | 7 | 34 PA1 |
| TIMER | 8 | 33 PA0 |
| PC0 | 9 | 32 PB7 |
| PC1 | 10 | 31 PB6 |
| PC2 | 11 | 30 PB5 |
| PC3 | 12 | 29 PB4 |
| PC4 | 13 | 28 PB3 |
| PC5 | 14 | 27 PB2 |
| PC6 | 15 | 26 PB1 |
| PC7 | 16 | 25 PB0 |
| PD7 | 17 | 24 PD0 |
| PD6/INT2 | 18 | 23 PD1 |
| PD5 | 19 | 22 PD2 |
| PD4 | 20 | 21 PD3 |

**Figure 1 :** EF6805U3 HMOS Microcomputer Block Diagram.



## ABSOLUTE MAXIMUM RATINGS

| Symbol | Parameter | Value | Unit |
|--------|-----------|-------|------|
| $V_{CC}$ | Supply Voltage | – 0.3 to + 7.0 | V |
| $V_{in}$ | Input Voltage (except TIMER in self-check mode and open-drain inputs) | – 0.3 to + 7.0 | V |
| $V_{in}$ | Input Voltage (open-drain pins, TIMER pin in self-check mode) | – 0.3 to + 15.0 | V |
| $T_A$ | Operating Temperature Range<br>($T_L$ to $T_H$)  V Suffix  T Suffix | 0 to + 70<br>– 40 to + 85<br>– 40 to + 105 | °C |
| $T_{stg}$ | Storage Temperature Range | – 55 to + 150 | °C |
| $T_j$ | Junction Temperature<br>Plastic Package<br>PLCC | <br>150<br>150 | °C |

This device contains circuitry to protect the inputs against damage due to high static voltages or electrical fields, however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that $V_{in}$ and $V_{out}$ be constrained to the range $V_{SS}$  ($V_{in}$ or $V_{out}$)  $V_{CC}$. Reliability of operation is enhanced if unused inputs except EXTAL are tied to an appropriate logic voltage level (e.g., either $V_{SS}$ or $V_{CC}$).

## THERMAL DATA

| | | | | |
|---|---|---|---|---|
| $\theta_{JA}$ | Thermal Resistance | Plastic<br>PLCC | 50<br>80 | °C/W |

**SGS-THOMSON**
**MICROELECTRONICS**

## POWER CONSIDERATIONS

The average chip-junction temperature, $T_J$, in C can be obtained from :

$T_J = T_A + (P_D.J_A)$ (1)

Where :

$T_A$ = Ambient Temperature, C

$J_A$ = Package Thermal Resistance, Junction-to-Ambient, C/W

$P_D = P_{INT} + P_{PORT}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts - Chip Internal Power

$P_{PORT}$ = Port Power Dissipation, Watts - User Determined

For most applications $P_{PORT}$ $P_{INT}$ and can be neglected. $P_{PORT}$ may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between $P_D$ and $T_J$ (if $P_{PORT}$ is neglected) is :

$P_D = K + (T_J + 273C)$ (2)

Solving equations 1 and 2 for K gives :

$K = P_D.(T_A + 273C) + J_A.P_D^2$ (3)

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring $P_D$ (at equilibrium) for a known $T_A$. Using this value of K the values of $P_D$ and $T_J$ can be obtained by solving equations (1) and (2) iteratively for any value of $T_A$.

**ELECTRICAL CHARACTERISTICS** ($V_{CC}$ = + 5.25Vdc ± 0.5Vdc, $V_{SS}$ = 0Vdc, $T_A = T_L$ to $T_H$ unless otherwise noted)

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $V_{IH}$ | Input High Voltage<br>RESET ($4.75 \leq V_{CC} \leq 5.75$)<br>($V_{CC} < 4.75$)<br>INT ($4.75 \leq V_{CC} \leq 5.75$)<br>($V_{CC} < 4.75$)<br>All Other (except timer) | 4.0<br>$V_{CC} - 0.5$<br>4.0<br>$V_{CC} - 0.5$<br>2.0 | •<br>• | $V_{CC}$<br>$V_{CC}$<br>$V_{CC}$<br>$V_{CC}$<br>$V_{CC}$ | V |
| $V_{IH}$ | Input High Voltage Timer<br>Timer Mode<br>Self-check Mode | 2.0<br>9.0 | 10.0 | $V_{CC} + 1.0$<br>15.0 | V |
| $V_{IL}$ | Input Low Voltage<br>RESET<br>INT<br>All Other | $V_{SS}$<br>$V_{SS}$<br>$V_{SS}$ | • | 0.8<br>1.5<br>0.8 | V |
| $V_{IRES +}$<br>$V_{IRES -}$ | RESET Hystereris Voltages (see figures 10, 11 and 12)<br>"Out of Reset"<br>"Into Reset" | 2.1<br>0.8 | | 4.0<br>2.0 | V |
| $V_{INT}$ | INT Zero Crossing Input Voltage, Through a Capacitor | 2 | | 4 | $V_{ac\ p-p}$ |
| $P_D$ | Power Dissipation - (no port loading, $V_{CC}$ = 5.75V)<br>$T_A = 0°C$<br>$T_A = - 40°C$ | | 520<br>580 | 740<br>800 | mW |
| $C_{in}$ | Input Capacitance<br>EXTAL<br>All Other | | 25<br>10 | | pF |
| $V_{LVR}$ | Low Voltage Recover | | | 4.75 | V |
| $VL_{VI}$ | Low Voltage Inhibit | 2.75 | 3.75 | 4.70 | V |
| $I_{in}$ | Input Current<br>TIMER ($V_{in}$ = 0.4V)<br>INT ($V_{in}$ = 2.4V to $V_{CC}$)<br>EXTAL ($V_{in}$ = 2.4V to $V_{CC}$ - crystal option)<br>($V_{in}$ = 0.4V - crystal option)<br>RESET ($V_{in}$ = 0.8V) - External Capacitor Charging Current | - 40.0 | 20 | 20<br>50<br>10<br>- 1600<br>- 40 | µA |

* Due to internal biasing this input (when unused) floats to approximately 2.2V.

### SWITCHING CHARACTERISTICS

($V_{CC}$ = + 5.25Vdc ± 0.5Vdc, $V_{SS}$ = 0Vdc, $T_A$ = $T_L$ to $T_H$ unless otherwise noted)

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $f_{osc}$ | Oscillator Frequency | 0.4 | | 4.2 | MHz |
| $t_{cyc}$ | Cycle Time (4/$f_{osc}$) | 0.95 | | 10 | µs |
| $t_{WL}$, $t_{WH}$ | $\overline{INT}$, $\overline{INT2}$, and TIMER Pulse Width (see interrupt section) | $t_{cyc}$ + 250 | | | ns |
| $t_{RWL}$ | $\overline{RESET}$ Pulse Width | $t_{cyc}$ + 250 | | | ns |
| $f_{INT}$ | $\overline{INT}$ Zero-crossing Detection Input Frequency | 0.03 | | 1 | kHz |
| | External Clock Input Duty Cycle (EXTAL) | 40 | 50 | 60 | % |
| | Crystal Oscillator Start-up Time* | | | 100 | ms |

### PORT ELECTRICAL CHARACTERISTICS

($V_{CC}$ = + 5.25Vdc ± 0.5Vdc, $V_{SS}$ = 0Vdc, $T_A$ = $T_L$ to $T_H$ unless otherwise noted)

#### PORT A WITH CMOS DRIVE ENABLED

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $V_{OL}$ | Output Low Voltage ($I_{Load}$ = 1.6mA) | | | 0.4 | V |
| $V_{OH}$ | Output High Voltage<br>$I_{Load}$ = − 100µA<br>$I_{Load}$ = − 10µA | <br>2.4<br>$V_{CC}$ − 1.0 | | | V |
| $V_{IH}$ | Input High Voltage ($I_{Load}$ = − 300µA max.) | 2.0 | | $V_{CC}$ | V |
| $V_{IL}$ | Input Low Voltage ($I_{Load}$ = − 500µA max.) | $V_{SS}$ | | 0.8 | V |
| $I_{IH}$ | High Z State Input Current ($V_{in}$ = 2.0V to $V_{CC}$) | | | − 300 | µA |
| $I_{IL}$ | High Z State Input Current ($V_{in}$ = 0.4V) | | | − 500 | µA |

#### PORT B

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $V_{OL}$ | Output Low Voltage<br>$I_{Load}$ = 3.2mA<br>$I_{Load}$ = 10mA (sink) | | | <br>0.4<br>1.0 | V |
| $V_{OH}$ | Output High Voltage $I_{Load}$ = − 200µA | 2.4 | | | V |
| $I_{OH}$ | Darlington Current Drive (source) $V_O$ = 1.5V | − 1.0 | | − 10 | mA |
| $V_{IH}$ | Input High Voltage | 2.0 | | $V_{CC}$ | V |
| $V_{IL}$ | Input Low Voltage | $V_{SS}$ | | 0.8 | V |
| $I_{TSI}$ | High Z State Input Current | | < 2 | 10 | µA |

#### PORT C AND PORT A WITH CMOS DRIVE DISABLED

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $V_{OL}$ | Output Low Voltage $I_{Load}$ = 1.6mA | | | 0.4 | V |
| $V_{OH}$ | Output High Voltage $I_{Load}$ = − 100µA | 2.4 | | | V |
| $V_{IH}$ | Input High Voltage | 2.0 | | $V_{CC}$ | V |
| $V_{IL}$ | Input Low Voltage | $V_{SS}$ | | 0.8 | V |
| $I_{TSI}$ | High Z State Input Current | | < 2 | 10 | µs |

**SGS-THOMSON**
**MICROELECTRONICS**

**Figure 2 :** TTL Equivalent Test Load (port B).



**Figure 3 :** CMOS Equivalent Test Load (port A).



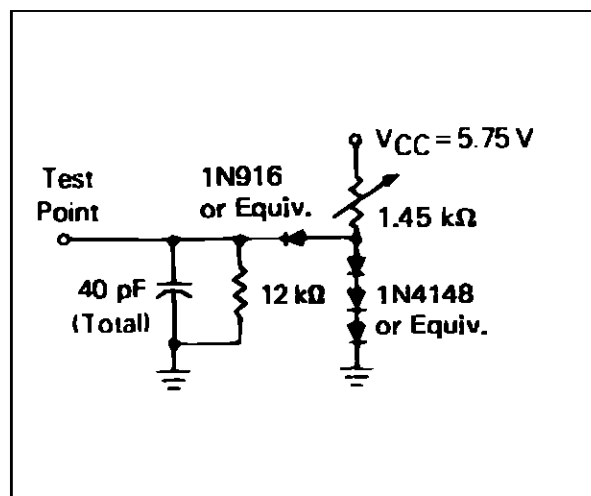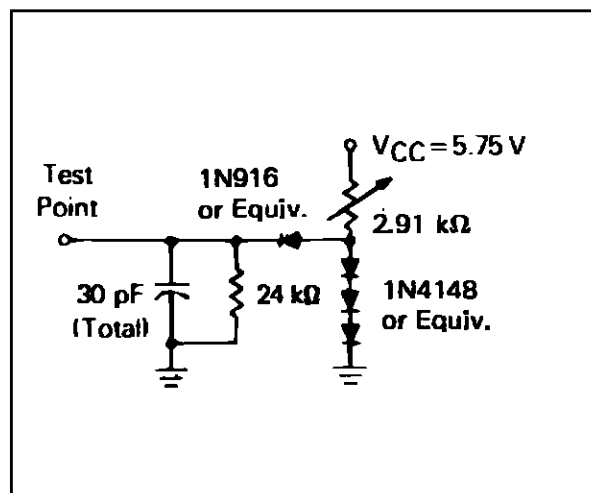**Figure 4 :** TTL Equivalent Test Load (port A and C).
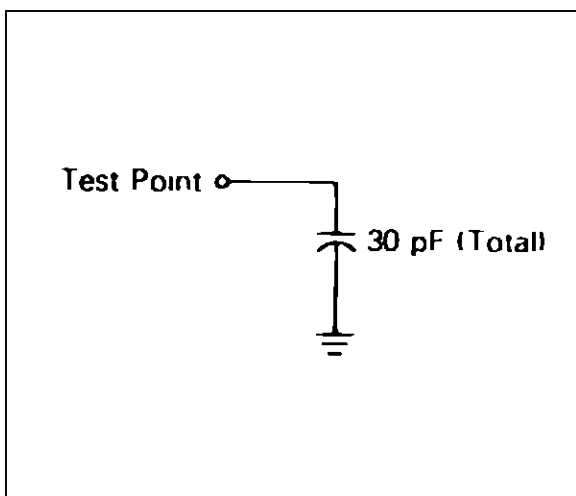


**Figure 5 :** Open-drain Equivalent Test Load (port C).



## SIGNAL DESCRIPTION

The input and output signals for the MCU, shown in figure 1, are described in the following paragraphs.

$V_{CC}$ AND $V_{SS}$ - Power is supplied to the MCU using these two pins. $V_{CC}$ is power and $V_{SS}$ is the ground connection.

INT - This pin provides the capability for asynchronously applying an external interrupt to the MCU. Refer to Interrupts Section for additional information.

XTAL AND EXTAL - These pins provide control input for theon-chip clock oscillator circuit. A crystal, a resistor, or an external signal, depending on user selectable manufacturing mask option, can be connected to these pins to provide a system clock with various degrees of stability/cost tradeoffs. Lead length and stray capacitance on these two pins should be minimized. Refer to Internal Clock Generator Options Section for recommendations about these inputs.

**NOTE : Pin 7 in DIL package/pin 8 in PLCC package is connected to internal protection.**

TIMER - The pin allows an external input to be used to control the internal timer circuitry and also to initiate the self test program. Refer to Timer Section for additional information about the timer circuitry.

RESET - This pin allows resetting of the MCU at times other than the automatic resetting capability already in the MCU. The MCU can be reset by pulling RESET low. Refer to Resets Section for additional information.

**SGS-THOMSON**
**MICROELECTRONICS**

INPUT/OUTPUT LINES (PA0-PA7, PB0-PB7, PC0-PC7, PD0-PD7) - These 32 liens are arranged into four 8-bit ports (A, B, C, and D). Ports A, B, and C are programmable as either inputs or outputs under software control of the data direction registers (DDRs). Port D is for digital input only and bit 6 may be used for a second interrupt INT2. Refer to Input/Output Section and Interrupts Section for additional information.

MEMORY - The MCU is capable of addressing 4096 bytes of memory and I/O registers with its program counter. The EF6805U3 MCU has implemented 4090 of these bytes. This consists of : 3776 user ROM bytes, 192 self-check ROM bytes, 112 user RAM bytes, 7 port I/O bytes, 2 timer registers, and a miscellaneous register ; see figure 6 for the Address map. The user ROM has been split into two areas. The main user ROM area is from $080 to $F37. The last 8 user ROM locations at the bottom of memory are for the interrupt vectors.

The MCU reserves the first-16 memory locations for I/O features, of which 10 have been implemented. These locations are used for the ports, the port DDRs, the timer and the INT2 miscellaneous register, and the 112 RAM bytes, 31 bytes are shared with the stack area. The stack must be used with care when data shares the stack area.

The shared stack area is used during the processing of an interrupt or subroutine calls to save the contents of the CPU state. The register contents are pushed onto the stack in the order shown in figure 7. Since the stack pointer decrements during pushes, the low order byte (PCL) of the program counter is stacked first, then the high order four bits (PCH) are stacked. This ensures that the program counter is loaded correctly during pulls from the stack since the stack pointer increments when it pulls data from the stack. A subroutine call results in only the program counter (PCL, PCH) contents being pushed onto the stack ; the remaining CPU registers are not pushed.

**Figure 6 :** EF6805U3 MCU Address Map.
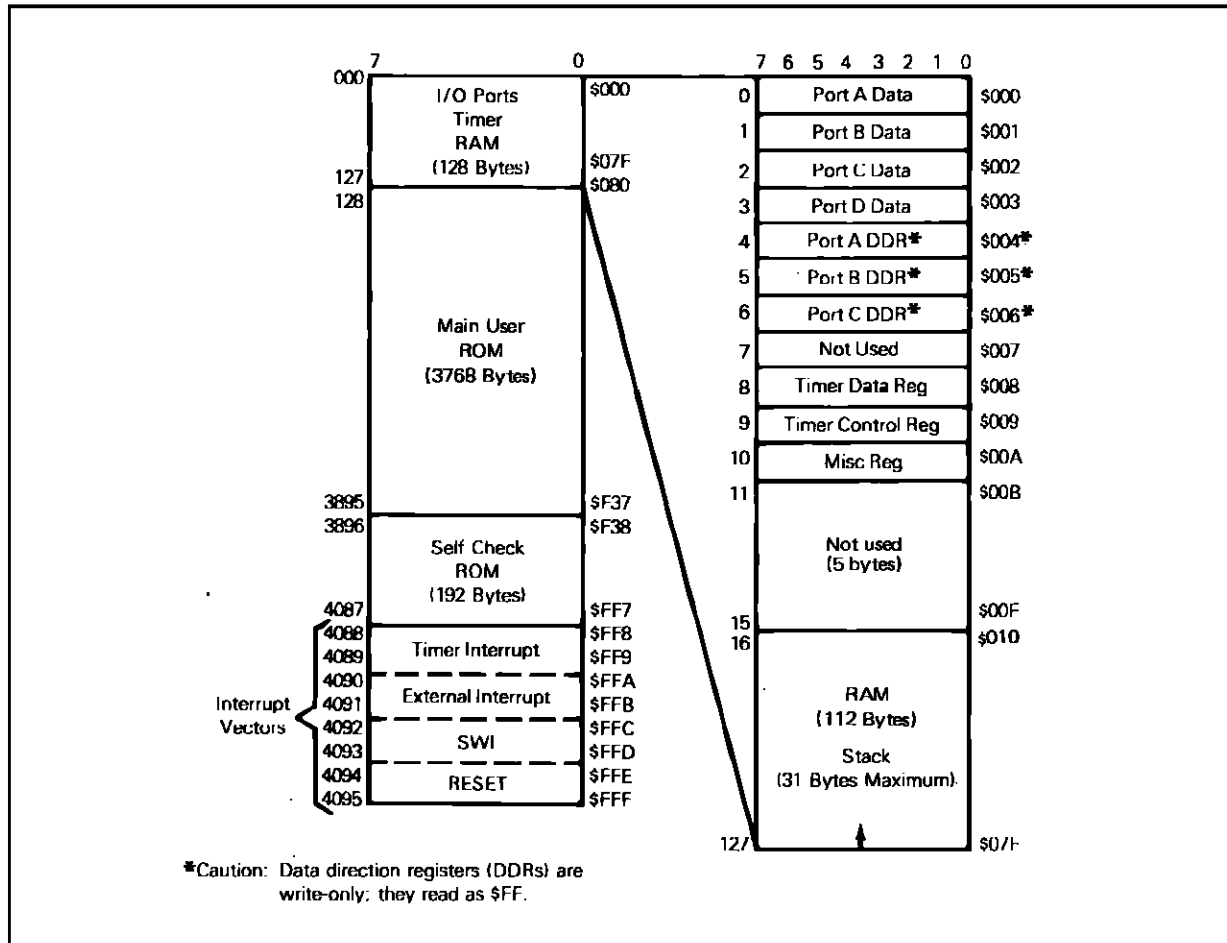


*Caution : Data direction registers (DDRs) are write only, they read as $FF.

**Figure 6 :** Interrupt Stacking Order.
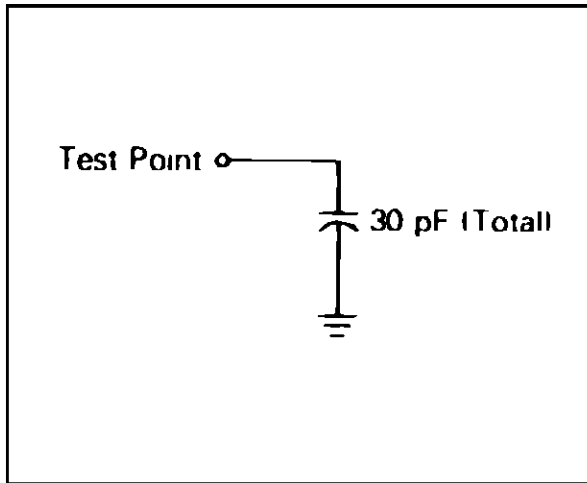


## CENTRAL PROCESSING UNIT

The CPU of the EF6805 Family is implemented independently from the I/O or memory configuration.

Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal address, data, and control buses.
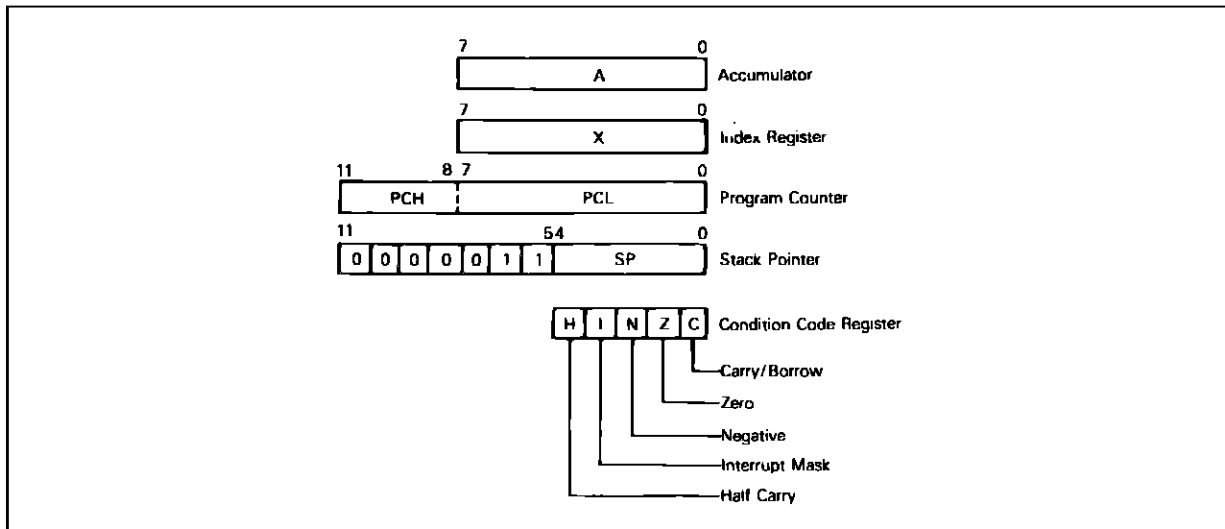
## REGISTERS

The 6805 Family CPU has five registers available to the programmer. They are shown in figure 8 and are explained in the following paragraphs.

ACCUMULATOR (A) - The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manupulations.

INDEX REGISTER (X) - The index register is an 8-bit register used for the indexed addressing mode. It contains an 6-bit value that may be added to an instruction value to create an effective address. The index register can also be used for data manipulations using the read-modify-write instructions. The Index Register may also be used as a temporary storage area.

**Figure 8 :** Programming Model.



PROGRAM COUNTER (PC) - The Program Counter is a 12 bit register that contains th address of the next instruction to be executed.

STACK POINTER (SP) - The stack pointer is a 12-bit register that contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $07F. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is then pulled from the stack. The seven most significant bits of the stack pointer are permanently set to 0000011. Subroutines and

interrupts may be nested down to location $061 (31 bytes maximum) which allows the programmer to use up to 15 levels of subroutine calls (less if interrupts are allowed).

CONDITION CODE REGISTER (CC) - The condition code register is a 5-bit register in which foour bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each bit is explained in the following paragraphs.

Half Carry (H) - Set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

Interrupt (I) - When this bit is set, the timer an external interrupts (INT and INT2) are masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and is processed as soon as the interrupt bit is cleared.

Negative (N) - When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logical "1").

Zero (Z) - When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

Carry/Borrow (C) - When set, this bit indicates that a carry or borrow ou of the Arithmetic Logic Unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions plus shifts and rotates.

## TIMER

The timer circuitry for the EF6805U3 is shown in figure 10. The timer contains a single 8-bit software programmable counter with a 7-bit software selectable prescaler. The counter may be preset under program control and decrements toward zero. When the counter decrements to zero, the timer interrupt request bit, i.e., bit 7 of the timer control register (TCR), is set. Then if the timer interrupt is not masked, i.e.,bit 6 of the TCR and the I bit in the condition code register are both cleared, the processor receives an interrupt. After completion of the current instruction, the processor proceeds to store the appropriate registers on the stack, and then fetches the timer interrupt vector from locations $FF8 and $FF9 in order to begin servicing the interrupt.

The counter continues to count after it reaches zero, allowing the software to determine the number of internal or external input clocks since the timer interrupt request bit was set. The counter may be read at any time by the processor without disturbing the count. The contents of the counter become stable prior to the read portion of a cycle and do not change during the read. The timer interrupt request bit remains set until cleared by the software. If a write occurs before the timer interrup is sericed, the interrupt is lost. TCR7 may also be used as a scanned status bit in a non-interrupt mode of operation (TCR6 = 1).

The prescaler is a 7-bit divider which is used to extend the maximum length of the timer. Bit 0, bit 1, and bit 2 of the TCR are programmed to choose the appropriate prescaler outputu which is used as the counter input. The processor cannot writ eijto or

read from the prescaler ; however, its contents are cleared to all zeros by the write operation into TCR when bit 3 of the written data equals one, which allows for truncation-free counting.

The timer input can be configured for three different operating modes, plus a disable mode, depending on the value written to the TCR4 and TCR5 control bits. For further information see figure 9.
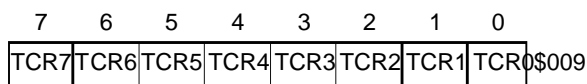
Timer Input Mode 1 - If TCR5 adn TCR4 are both programmed to a zero, the inpt to the timer is from an internal clock and the external TIMER input is disabled. The internal clock mode canbe used for periodic interrupt generation, as well as a referene in frequency and event measurement. The internal clock is the instruction cycle clock.

Timer Input Mode 2 - With TCR5 = 0 and TCR4 = 1, the internal clock and the TIMER input pin are ANDed to form the timer input signal. This mode can be used to measure external pulse widths. The external timer input pulse simply turns on the internal clock for the duration of the pulse widths.

Timer Input Mode 3 - If TCR5 = 1 and TCR4 = 0, then all inputs to the timer are disabled.

Timer Input Mode 4 - If TCR5 = 1 and TCR4 = 1, the internal clock input to the timer is disabled and the TIMER input pin becomes the input to the timer. The external TIMER pin can, in this mode, be used to count external events as well as external frequencies for generating periodic interrupts.

TCR7 - Timer Interrupt Request Bit :

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|
| TCR7 | TCR6 | TCR5 | TCR4 | TCR3 | TCR2 | TCR1 | TCR0 | $009 |

\* Write only (read as zero).

    1 - Set when TDR goes to zero, or under program control
    0 - Cleared on external Reset, Power-On-Reset, or under Program Control.

TCR6 - Timer Interrupt Mask Bit :
    1 - Timer Interrupt masked (disabled) Set on external Reset, Power-On-Reset, or under Program Control
    0 - Cleared under Program Control.

TCR5 - External or Internal Clock Source Bit :
    1 - External Clock Source. Set on external Reset, Power-On-Reset, or under Program Control
    0 - Cleared under Program Control.

TCR4 - External Enable Bit :
    1 - Enable external TIMER pin. Set on external Reset, Poxer-On-Reset, or under Program

Control.

0 - Cleared under Program Control.

TCR3 - Timer prescaler reset bit : A read of TCR3

| TCR5 | TCR4 | Result |
|---|---|---|
| 0 | 0 | Internal Clock to Timer |
| 0 | 1 | AND of Internal Clock and TIMER Pin to Timer |
| 1 | 0 | Input to timer disabled. |
| 1 | 1 | TIMER Pin to Timer |

always indicates a zero.

1 - Set on external Reset, Power-On-Reset or under Program Control.

0 - Cleared under Program Control

**Figure 10 :** Timer Block Diagram.

TCR2 , TCR1, and TCR0 - Prescaler address bits :

1 - All set on external Reset, Power-On-Reset or under Program Control.

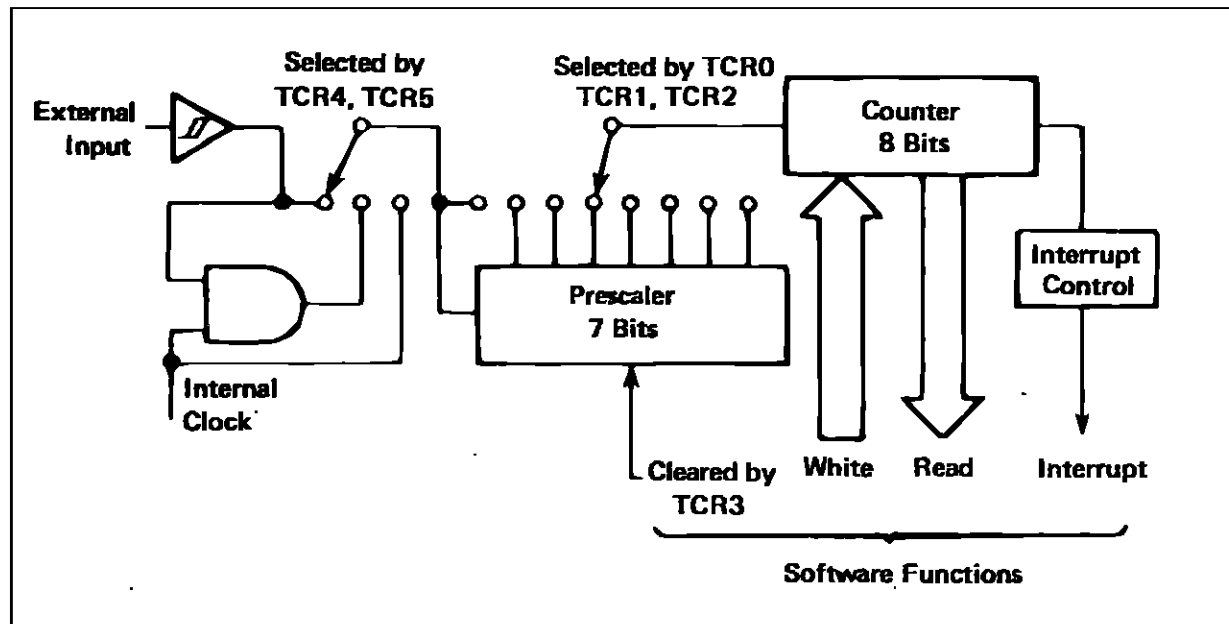0 - Cleared under Program Control.

**Figure 9 :** Timer Control Register (TCR).

| TCR2 | TCR1 | TCR0 | Result | TCR2 | TCR1 | TCR0 | Result |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | + 1 | 1 | 0 | 0 | + 16 |
| 0 | 0 | 1 | + 2 | 1 | 0 | 1 | + 32 |
| 0 | 1 | 0 | + 4 | 1 | 1 | 0 | + 64 |
| 0 | 1 | 1 | + 8 | 1 | 1 | 1 | + 128 |



Notes :   1.  Prescaler and 8-bit counter are clocked on the failing edge of the internal clock (AS) or external input.
2.  Counter is written to during dat strobe (DS) and counts down continuously.

SELF-CHECK - The self-check capability of the EF6805U3 MCU provides an internal check to determine if the part is functional. Connect the MCU as shown in figure 11 and monitor the output of Port C bit 3 for an oscillation of approximately 7Hz. A 10-volt level (through a 10k resistor) on the timer input, pin 8 and pressing then releasing the RESET button, energizes the ROM-based self-check feature. The self-check program exercices the RAM, ROM, TIMER, interrupts, and I/O ports.

Several of the self-check subroutines can be called by a user program with a JSR or BSR instruction. They are the RAM, ROM. The timer routine may also be called if the timer input is the internal 2 clock.

To call those subroutines in customer application, please contact your local SGS-THOMSON Micro-electronics sales office in order to obtain the complete description of the self-check program and the entrance/exit conditions.

RAM SELF-CHECK SUBROUTINE - The RAM self-check is called at location $F84 and returns with the Z bit clear if any error is detected ; otherwise the Z bit is set. The RAM test causes each byte to count from 0 up to 0 again with a check after each count.

The RAM test must be called with the stack pointer at $07F and A = 0. When run, the test checks every RAM cell except for $07F and $07E which are assumed to contain the return address.

**SGS-THOMSON**
**MICROELECTRONICS**

The A and X registers and all RAM locations except $07F and $07E are modified.

ROM CHECKSUM SUBROUTINE - The ROM self-check is called at location $F95. The A register should be cleared before calling the routine. If any error is detected, it returns with the Z bit cleared ;

otherwise Z = 1, X = 0 on return, and A is zero if the test passes. RAM location $040 to $043 is overwritten. The checksum is the complement of the execution OR of the contents of the user ROM.

* This connection depends on clock oscillator user selectable mask option. Use jumper if the RC mask option is selected.

**Figure 11 :** Self-check Connections.
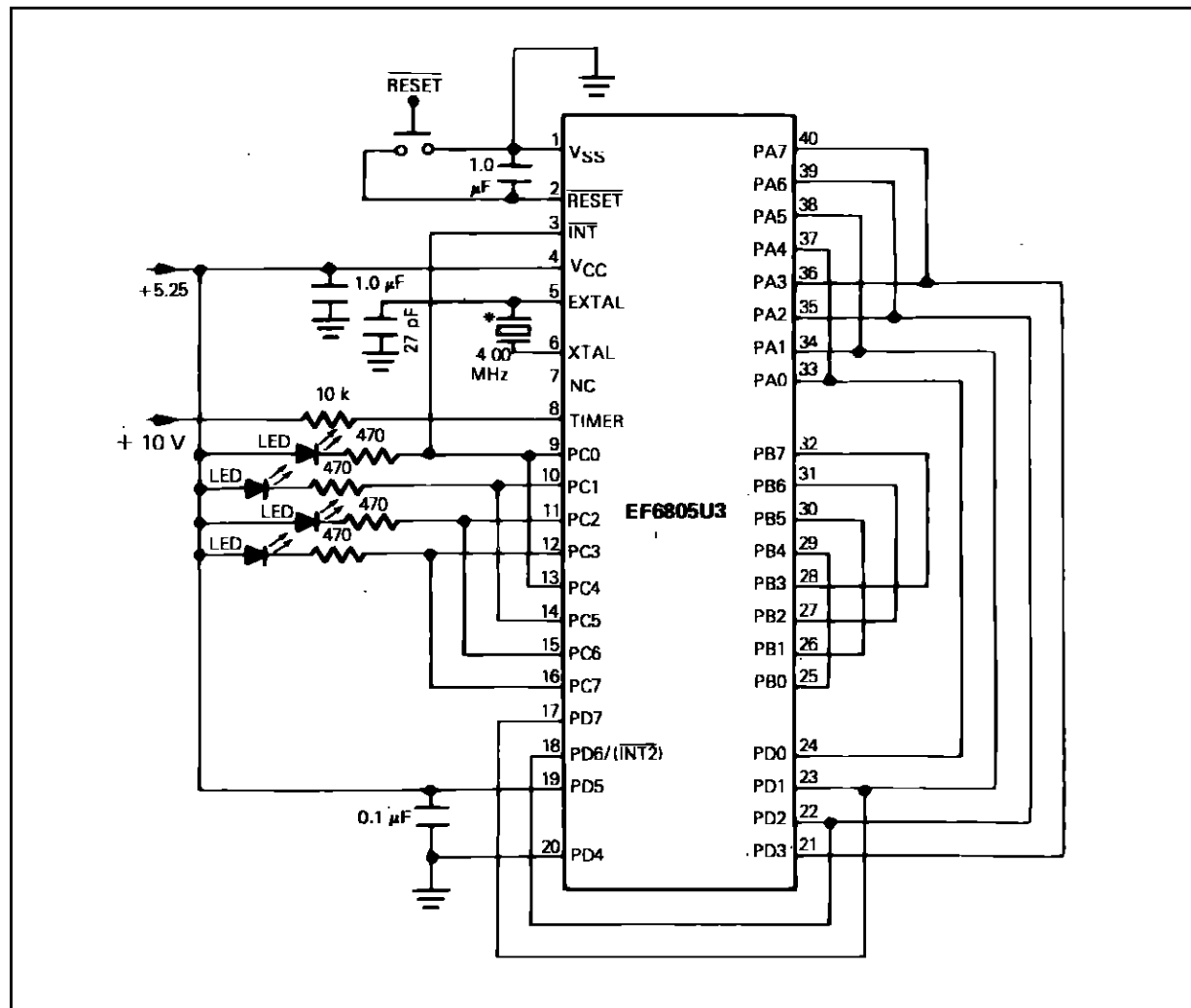


## LED MEANINGS

| PC0 | PC1 | PC2 | PC3 | Remarks (1 : LED ON ; 0 : LED OFF) |
|-----|-----|-----|-----|-----------------------------------|
| 1 | 0 | 1 | 0 | Bad I/O |
| 0 | 0 | 1 | 0 | Bad Timer |
| 1 | 1 | 0 | 0 | Bad RAM |
| 0 | 1 | 0 | 0 | Bad ROM |
| 0 | 0 | 0 | 0 | Bad Interrupts or Request Flag |
| All Flashing | | | | Good Device |

**SGS-THOMSON**
**MICROELECTRONICS**

TIMER SELF-CHECK SUBROUTINE - The timer self-check is called at location $F6D and returns with the Z bit cleared if any error was found ; otherwise Z = 1.

In order to work correctly as a user subroutine, the internal 2 clock must be the clocking source and interrupts must be disabled. Also, on exit, the clock is running and the interrupt mask is not set so the caller must protect from interrupts if necessary.

The A and X register contents are lost. This routine sets the prescaler for divide-by-128 and the timer data register is cleared. The X register is configured to count down the same as the timer data register. The two registers are then compared every 128 cycles until they both count down to zero. Any mismatch during the count down is considered as an error. The A and X registers are cleared on exit from the routine.

## RESET

The MCU can be reset three ways : by initial power-up, by the external reset input (RESET) and by an optional internal low-voltage detect circuit. The RESET input consists mainly of a Schmitt trigger which senses the RESET line logic level. A typical reset Schmitt trigger hysteresis curve is shown in figure 12. The Schmitt trigger provides an internal reset voltage if it senses a logical zero on the RESET pin. Power-On Reset (POR) - An internal reset is generated upon powerup that allows the internal clock generator to stabilize. A delay of $t_{RHL}$ milliseconds is required before allowing the RESET input to go high. Refer to the power and reset timing diagram of figure 13. Connecting a capacitor to the RESET input (as illustrated in figure 14) typically provides sufficient delay. During powerup, the Schmitt trigger switches on (removes reset) when RESET rises to $V_{IRES+}$.

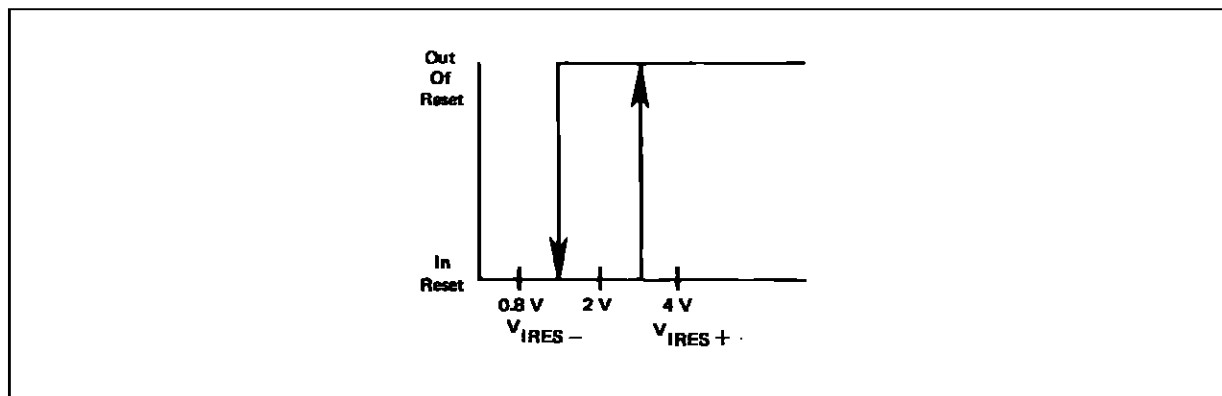**Figure 12 :** Typical Reset Schmitt Trigger Hysteresis.
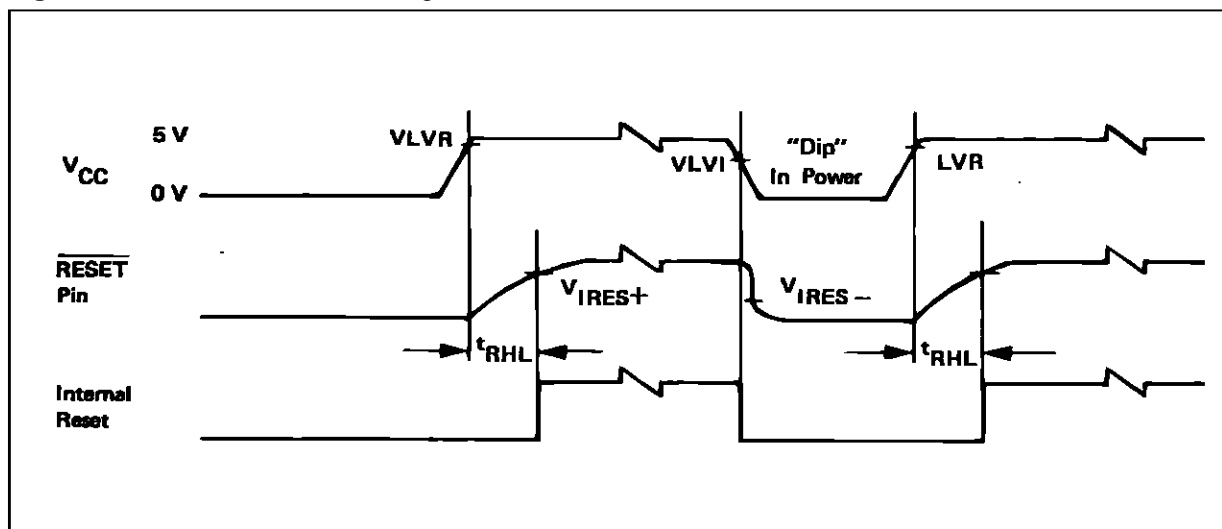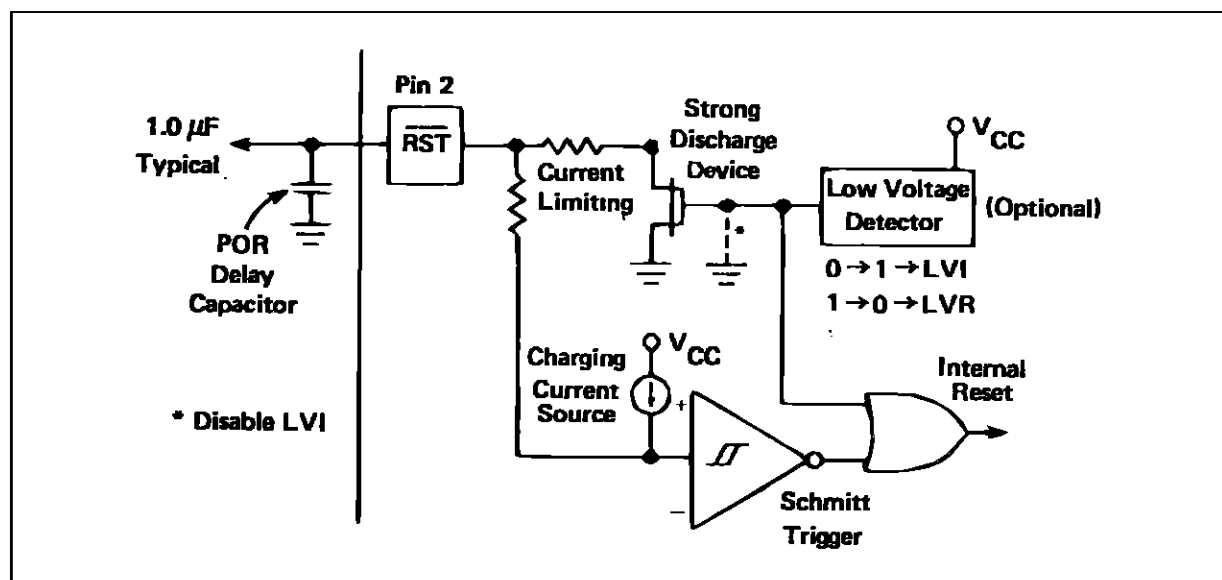


**Figure 13 :** Power and Reset Timing.

**Figure 14 :** RESET Configuration.

Pin 2 — RST
1.0 µF Typical — POR Delay Capacitor — * Disable LVI — Current Limiting — Strong Discharge Device — Charging Current Source — Schmitt Trigger — Low Voltage Detector (Optional) — V_CC — $0 \rightarrow 1 \rightarrow$ LVI — $1 \rightarrow 0 \rightarrow$ LVR — Internal Reset

External Reset Input - The MCU will be reset if a logical zero is applied to the RESET input for a period longer than one machine cycle ($t_{cyc}$). Under this type of reset, the Schmitt trigger switches off at $V_{IRES-}$ to provide an internal reset voltage.

Low-Voltage Inhibit (LVI) - The optional low-voltage detection circuit causes a reset of the MCU if the power supply voltage falls below a certain level ($V_{LVI}$). The only requirement is that $V_{CC}$ remains at or below the $V_{LVI}$ threshold for one $t_{cyc}$ minimum. In typical applications, the $V_{CC}$ bus filter capacitor will eliminate negative-going voltage glitches of less than one $t_{cyc}$. The output from the low-voltage detector is connected directly to the internal reset circuitry. It also forces the RESET pin low via a strong discharge device through a resistor. The internal reset will be removed once the power supply voltage rises above a recovery level ($V_{LVR}$), at which time a normal power-on-reset occurs.

## INTERNAL CLOCK GENERATOR OPTIONS

The internal clock generator circuit is designed to require a minimum of external components. A crystal, a resistor, a jumper wire, or an external signal may be used to generate a system clock with various stability/cost tradeoffs. The oscillator frequency is internally divided by four to produce the internal system clocks. A manufacturing mask option is used to select crystal or resistor operation.

The different connection methods are shown in figure 15. Crystal specifications and suggested PC board layouts are given in figure 16. A resistor selection graph is given in figure 17.

The crystal oscillator start-up time is a function of many variables : crystal parameters (especially $R_S$), oscillator load capacitances, IC parameters, ambient temperature, and supply voltage. To ensure rapid oscillator start up, neither the crystal characteristics nor the load capacitances should exceed recommendations.

When utilizing the on-board oscillator, the MCU should remain in a reset condition (reset pin voltage below $V_{IRES+}$) until the oscillator has stabilized at its operating frequency. Several factors are involved in calculating the external reset capacitor required to satisfy this condition ; the oscillator start-up voltage, the oscillator stabilization time, the minimum $V_{IRES+}$, and the reset charging current specification.

Once $V_{CC}$ minimum is reached, the external RESET capacitor will begin to charge at a rate dependent on the capacitor value. The charging current is supplied from $V_{CC}$ through a large resistor, so it appears almost like a constant current source until the reset voltage rises above $V_{IRES+}$. Therefore, the RESET pin will charge at approximately :

$$(V_{IRES+}).C_{ext} = I_{RES}.t_{RHL}$$

Assuming the external capacitor is initially discharged.

**Figure 15 :** Clock Generator Options.
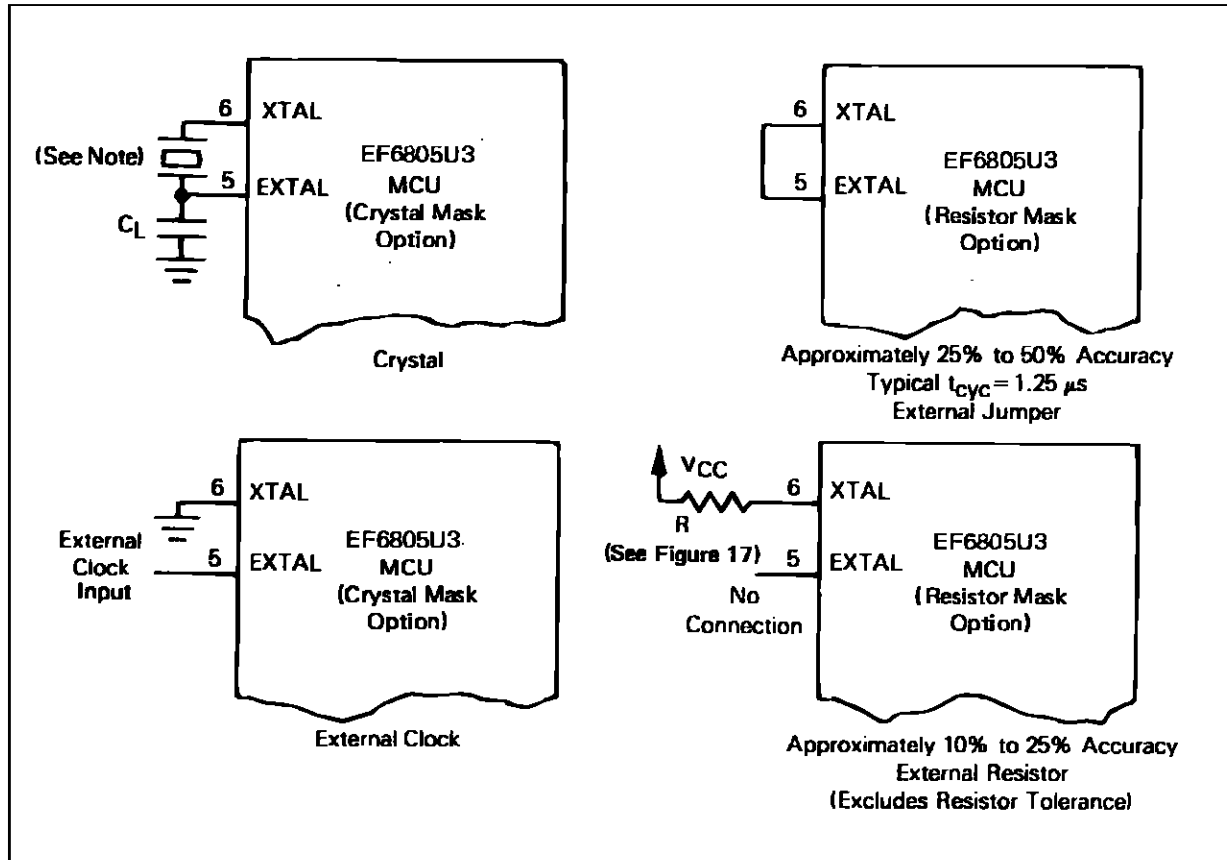


Note : The recommended CL value with a 4.0 MHz crystal is 27pF, maximum, including system distributed capacitance. There is an internal capacitance of approximately 25pF on the XTAL pin. For crystal frequencies other than 4MHz, the total capacitance on each pin should be scaled as the inverse of the frquency ratio. For example, with a 2MHz crystal, use approximately 50pF on EXTAL and approximately 25pF on XTAL. The exact value depends on the Motional-Arm parameters of the crystal used.

**Figure 16 :** Crystal Motional ARM parameters and Suggested PC Board layout.

SGS-THOMSON
MICROELECTRONICS
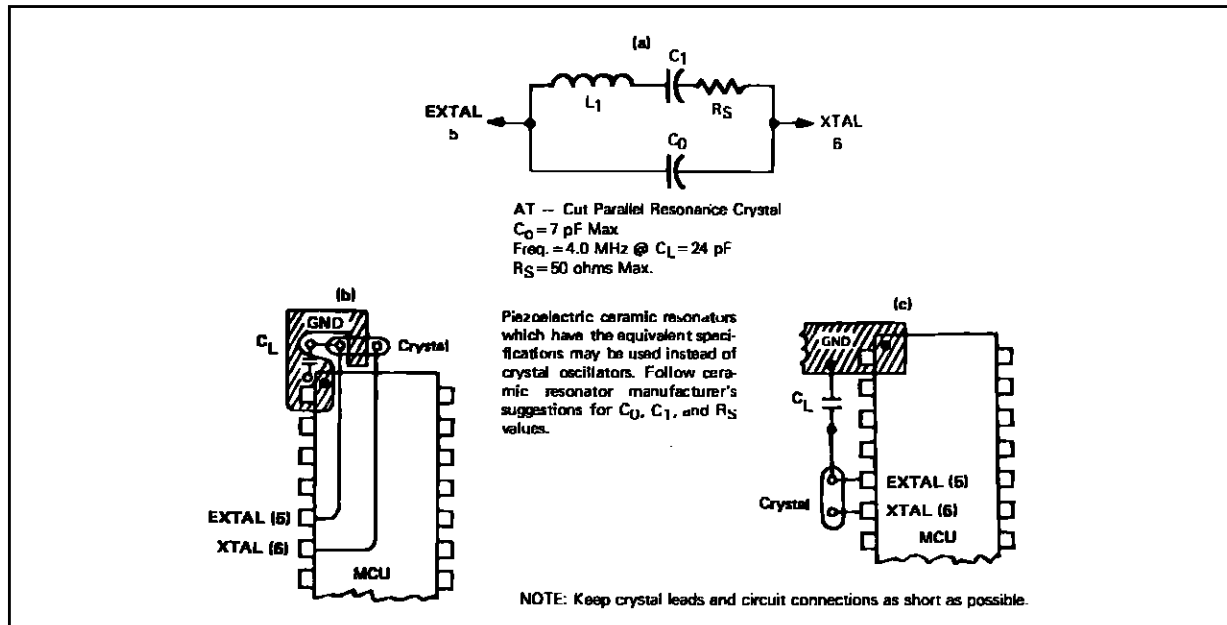
**Figure 17 :** Typical Frequency Selection for resistor (oscillator option).
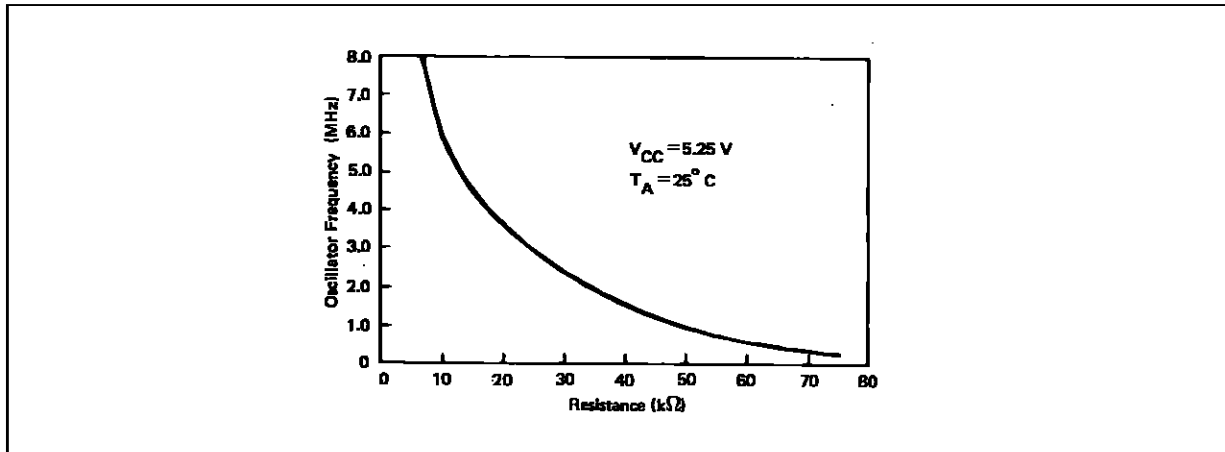


## INTERRUPTS

The microcomputers can be interrupted four different ways : through the external interrupt (INT) input pin, the internal timer interrupt request, the external port D bit 6 (INT2) input pin, or the software interrupt instruction (SWI). When any interrupt occurs : the current instruction (including SWI) is completed, processing is suspended, the present CPU state is pushed onto the stack, the interrupt bit (I) in the condition code register is set, the address of the interrupt routine is obtained from the appropriate interrupt vector address, and the interrupt routine is executed. Stacking the CPU register, setting the I bit, and vector fetching require a total of 11 $t_{cyc}$ periods for completion. A flowchart of the interrupt sequence is shown in figure 18. The interrupt service routine must end with a return from interrupt (RTI)

instruction which allows the MCU to resume processing of the program prior to the interrupt (by unstacking the previous CPU state). Unlike RESET, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing ; otherwise the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction.

**SGS-THOMSON**
**MICROELECTRONICS**

**Figure 18 :** RESET and interrupt Processing Flowchard.
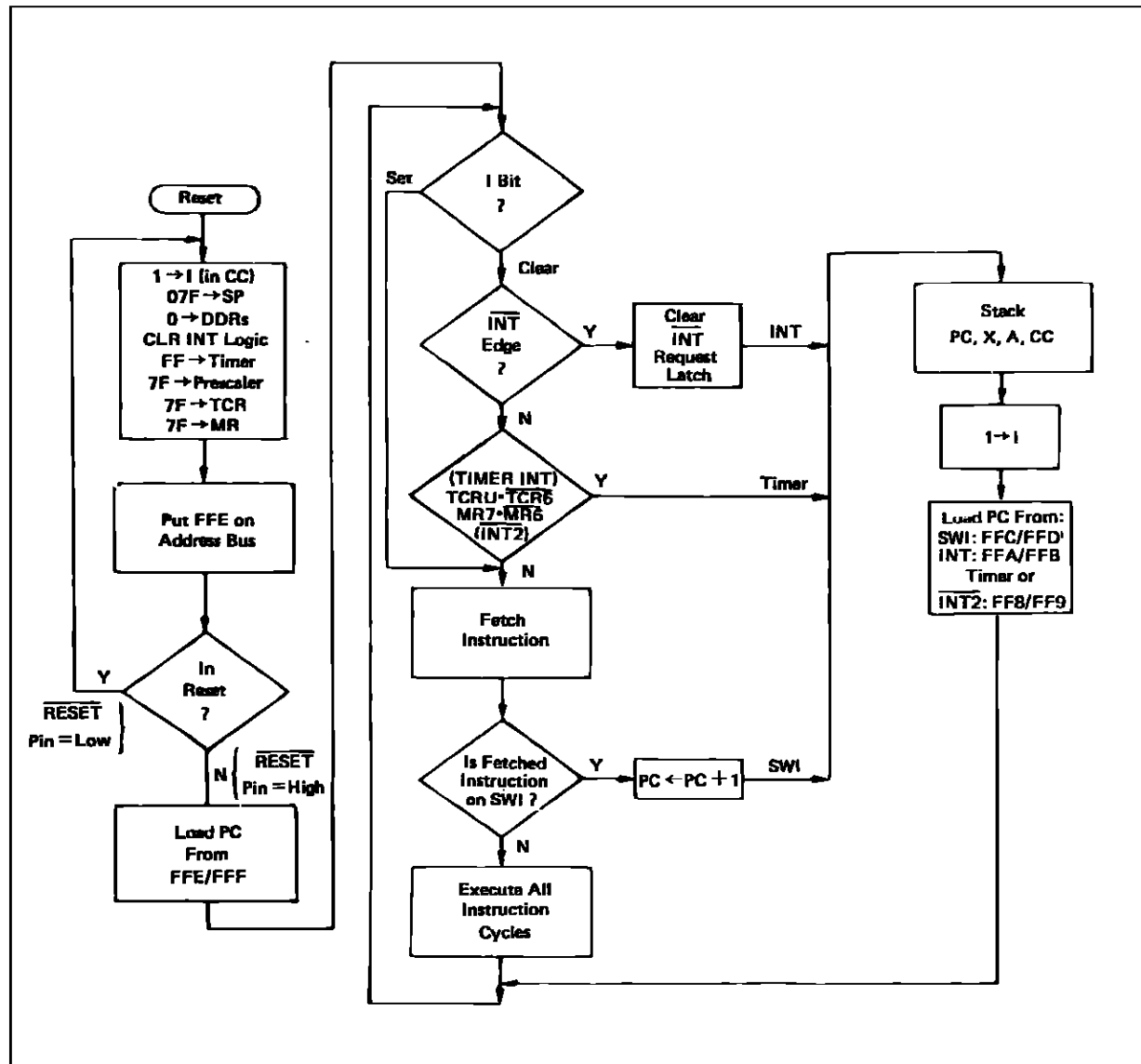


**NOTE**

The timer and INT2 interrupts share the same vector address. The interrupt routine must determine the source by examining the interrupt request bits (TCR b7 and MR b7). Both TCR b7 and MR b7 can only be written to zero by software.

The external interrupt, INT and INT2, are synchronized and then latched on the falling edge of the input signal. The INT2 interrupt has an interrupt request bit (bit 7) and a mask bit (bit 6) located in the miscellaneous register (MR). The INT2 interrupt is inhibited when the mask bit is set. The INT2 is always read as a digital input on port D. The INT2 and timer interrup requests bits, if set, cause the MCU to process an interrupt when the condition code I bit is clear.

A sinuoidal input signal ($f_{INT}$ maximum) can be used to generate an external interrupt for use as a zero-crossing detector. This allows applications such as servicing time-of-day routines and engaging/disengaging ac power control devices. Off-chip full wave rectification provides an interrupt at every zero crossing of the ac signal and thereby provides a 2f clock. See figure 19.

**NOTE**

The INT (pin 3) is internally biased at approximately 2.2V due to the internal zero-crossing detection.

A software interrupt (SWI) is an executable instruction which is executed regardless of the state of the I bit in the condition code register. SWIs are usually used as break-points for debugging or a system calls.

**Figure 19 :** Typical Interrupt Circuits.
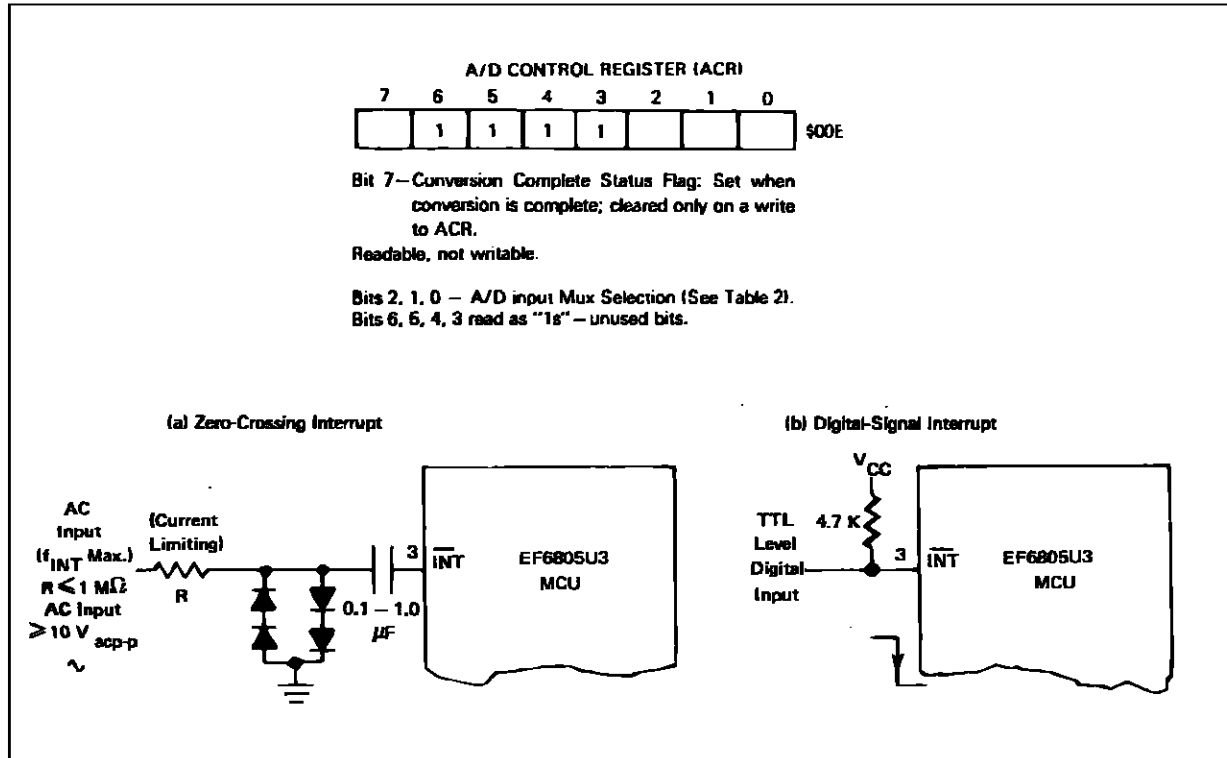


**INPUT/OUTPUT CIRCUITRY**

There are 32 input/output pins. The INT pin may be polled with branch instructions to provide an additional input pin. All pins on ports A, B, and C are programmable as either inputs or outputs under software control of the corresponding data direction register (DDR). See below I/O port control registers configuration. The port I/O programming is accomplished by writing the corresponding bit in the port DDR to a logic one for output or a logic zero for input. On reset all the DDRs are initialized to a logic zero state, placing the ports in the input mode. The port output registers are not initialized on reset and should be initialized by software before changing the DDRs from input to output. A read operation on a port programmed as an output will read the contents of the output latch regardless of the logic levels at the output pin, due to output loading. Refer to figure 20.

**PORT DATA REGISTER**

7                                                      0

Port A Addr = $000
Port B Addr = $001
Port C Addr = $002
Port D Addr = $003

**PORT DATA DIRECTION REGISTER (DDR)**

7                                                      0

(1) Write only ; reads as all "1s"
(2) 1 = Output. 0 = input Cleared to 0 by Reset
(3) Port A Addr = $004
    Port B Addr = $005
    Port C Addr = $006
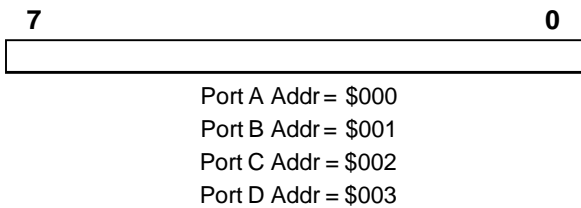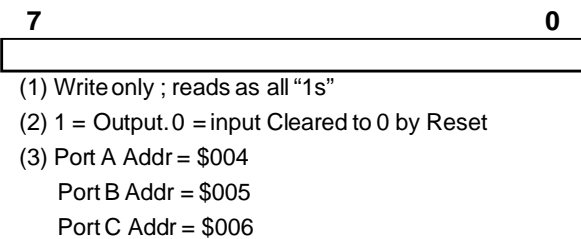
**Figure 20 :** Typical Port I/O Circuitry.



| Data Direction Register Bit | Latched Output Data Bit | Output State | Input to MCU |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |
| 0 | X | High-Z** | Pin |

* DDR is a write-only register and reads as all "1s".
** Ports B and C are three-state ports.

Port A has optional internal pull-up devices to provide CMOS data drive capability. See Electrical Characteristic tables for complete information.
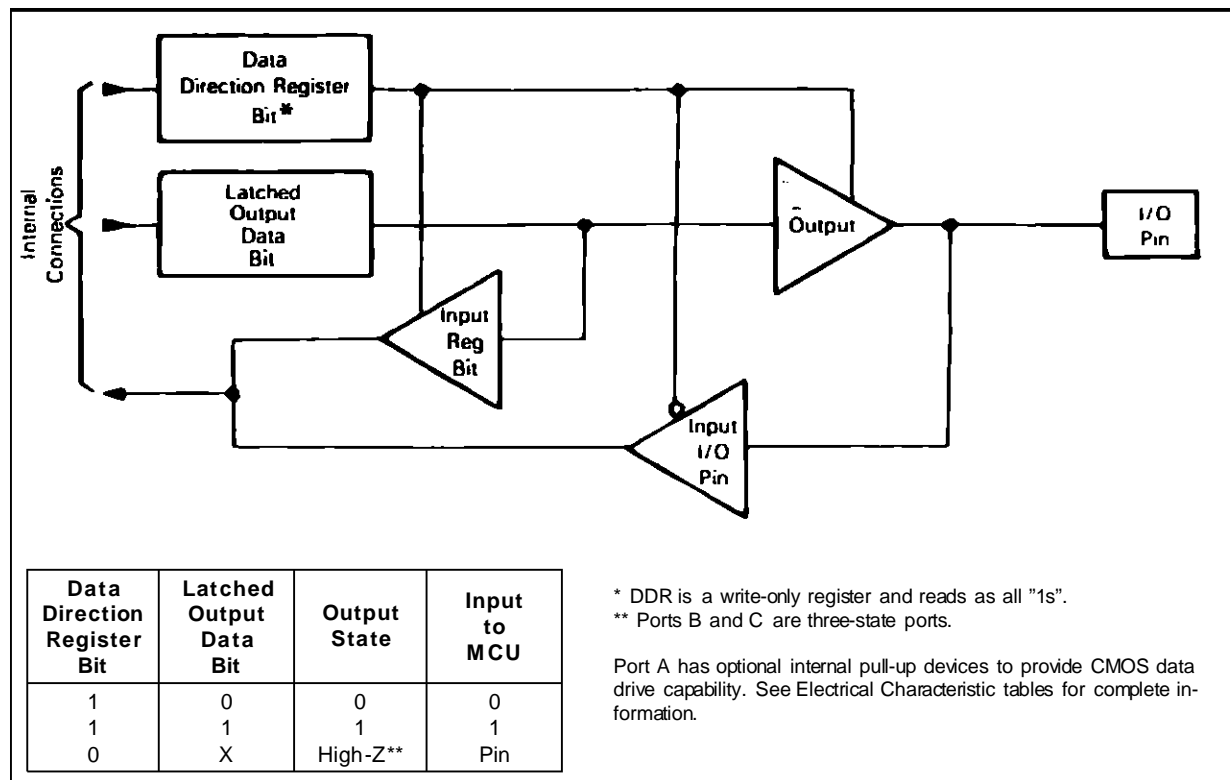
All input/output lines are TTL compatible as both inputs and outputs. Port A lines are CMOS compatible as outputs (mask option) while port B, C, and D lines are CMOS compatible as inputs. Port D lines are input only ; thus, there is no corresponding DDR. When programmed as outputs, port B is capable of sinking 10 milliamperes and sourcing 1 milliampere on each pin.

**The address map (figure 6) gives the addresses of data registers and data direction registers. Figure 21 provides some examples of port connections.**

### CAUTION

The corresponding DDRs for ports A, B, and C are write-only registers (registers at **$004, $005, $006**).

A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write in function, they cannot be used to set or clear a single DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

The latched output data bit (see figure 20) must always be written. Therefore, any write to a port writes all of its data bits even though the port DDR is set to input. This may be used to initialize the data register and avoid undefined outputs ; however, care must be exercised when using read-modify-write instructions, since the data read corresponds to the pin level if the DDR is an input (zero) and corresponds to the latched output data when the DDR is an output (one).

**Figure 21 :** Typical Port Connections.



Port A, bit 7 and bit 4 programmed as output. Bit 7 driving CMOS loads and bit 4 driving one TTL load directly using CMOS output option.

Port B, bit 5 programmed as output, driving Darlington-base directly.

Port B, bit 0 and bit 1 programmed as output, driving LEDs directly.

Port C, bits 0-3 programmed as output, driving CMOS loads, using external pullup resistors (required if Port C is open-drain).

SN74LS04 (Typical)

TTL driving Port A directly.

SN74LS04 or MC14069 (Typical)

CMOS or TTL driving Port B directly

MC14069 (Typical)

SN74LS04 (Typical)

CMOS and TTL driving Port C directly.

SN 74LS04 or MC14069 (typical)

CMOS or TTL driving Port D directly

## SOFTWARE

### BIT MANIPULATION

The EF6805U3 MCU has the ability to set or clear any single random access memory or input/output bit (except the data direction register, see Caution below), with a single instruction (BSET, BCLR). Any bit in page zero including ROM, except the DDRs, can be tested, using the BRSET and BRCLR instructions, and the program branches as a result of its state. The carry bit equals the value of the bit referenced by BRSET or BRCLR. A rotate instruction may then be used to accumulate serial input data in a RAM location or register. The capability to work with any bit in RAM, ROM, or I/O allows the user to have individual flags in RAM or to handle I/O bits as control lines.

The coding example in figure 21 illustrates the usefulness of the bit manipulation and test instructions.

**Figure 21 :** Bit Manipulation Example.

Assume that the MCU is to communicate with an external serial device.

The external device has a data ready signal, a data output line, and a clock line to clock data one bit at a time. LSB first, out of the device. The MCU waits until the data is ready, clocks the external device, picks up the data in the carry flag (C bit), clears the clock line, and finally accumulates the data bit in a RAM location.

**Caution**
The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.



## ADDRESSING MODES

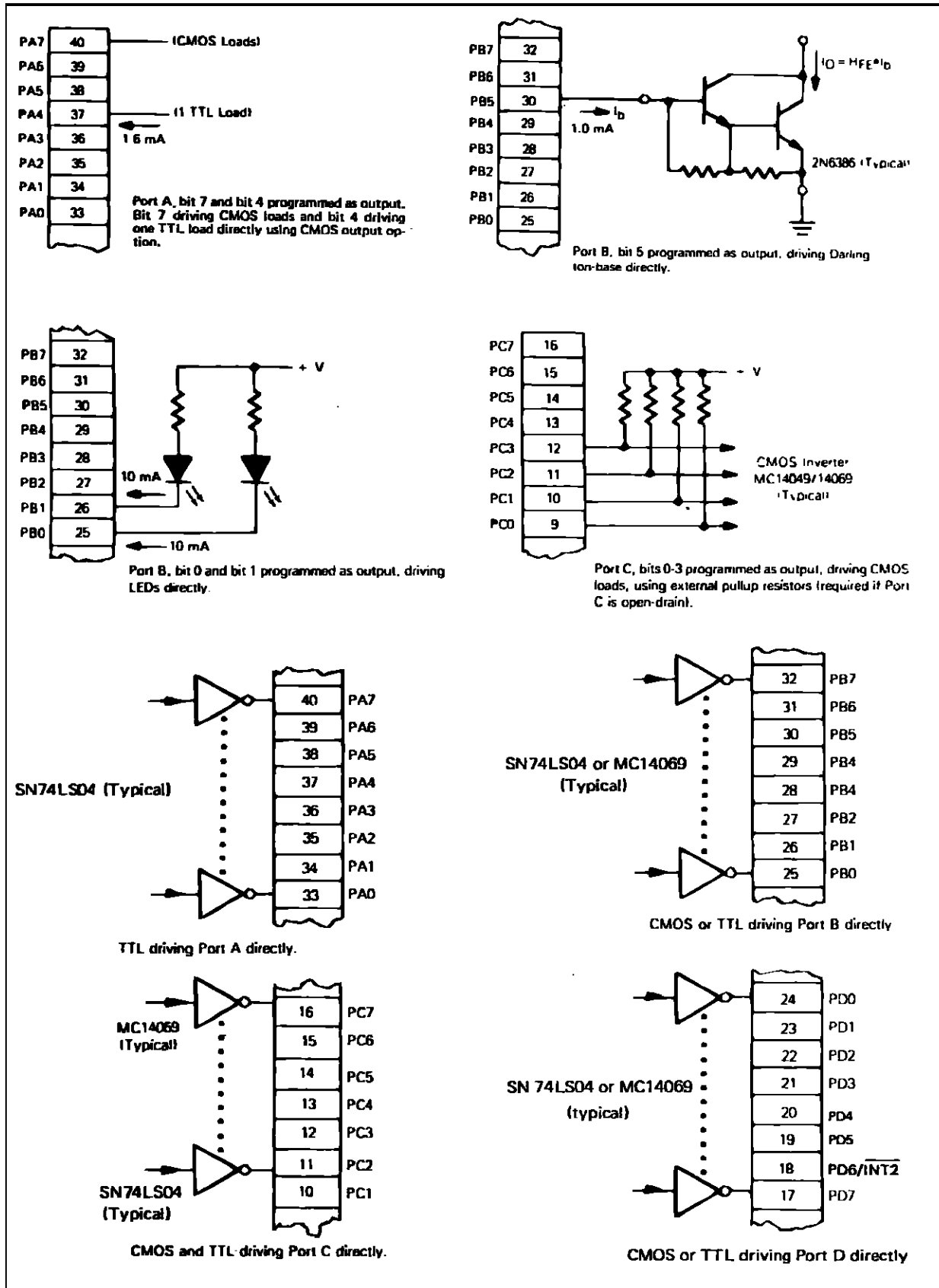The EF6805P2 MCU has 10 addressing modes which are explained briefly in the following paragraphs. For additional details and graphical illustrations, refer to the 6805 Family User's Manual.

The term "effective address" (EA) is used in describing the address modes. EA is defined as the address from which the argument for an instruction is fetched or stored.

**IMMEDIATE** - In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants which do not change during program execution (e.g;, a constant used to initialize a loop counter).

**DIRECT** - In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addresing al-

lows the user to directly address the lowest 256 bytes in memory with a single 2-byte instruction. This includes the on-chip RAM and I/O registers and 128 bytes of ROM. Direct addressing is an effective use of both memory and time.

**EXTENDED** - In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions using extended addressing are capable of referencing arguments anywhere in memory with a single 3-byte instruction. When using the Motorola assembler, the programmer need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest for of the instruction.

**RELATIVE** - The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte following the opcode (the offset) is added to the PC if and only if the

branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from - 126 to + 129 from the opcode address. The programmer need not worry about calculating the correct offset when using the Motorola assembler since it calculates the proper offset and checks to see if it is within the span of the branch.

**INDEXED, NO OFFSET** - In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

**INDEXED, 8-BIT OFFSET** - In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. This addressing mode is useful in selecting the kth element in an n element table. With this 2-byte instruction, k would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

**INDEXED, 16-BIT OFFSET** - In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed, 8-bit offset, except that this 3-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

**BIT SET/CLEAR** - In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct address of the byte in which the specified bit is to be set or cleared. Thus, any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single 2-byte instruction.

**Caution**
The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would

be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

**BIT TEST AND BRANCH** - The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit and condition (set or clear) which is to be tested is included in the opcode, and the address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset is in the third byte and is added to the value of the PC if the branch condition is true. This single 3-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from - 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

**Caution**
The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

**INHERENT** - In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, as well as control instruction with no other arguments, are included in this mode. These instructions are one byte long.

## INSTRUCTION SET

The EF6805U3 MCU has a set of 59 basic instructions, which when combined with the 10 addressing modes produce 207 usable opcodes. They can be divided into five different types : register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

**REGISTER/MEMORY INSTRUCTIONS** - Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operands. Refer to table 1.

**READ-MODIFY-WRITE MODIFICATIONS** - These instructions read a memory location or a register,

modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is included in read-modify-write instructions through it does not perform the write. Rfer to table 2.

**Caution**
The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

**BRANCH INSTRUCTIONS** - The branch instructions cause a branch from the program when a certain condition is met. Refer to table 3.

**BIT MANIPULATION INSTRUCTIONS** - These instructions are used on any bit in the first 256 bytes of the memory. One group either sets or clears. The other group performs the bit test branch operations. Refer to table 4.

**Caution**
The corresponding DDRs for ports A, B, and C are write-only registers (registers at $004, $005, and $006). A read operation on these registers is undefined. Since BSET and BCLR are read-modify-write functions, they cannot be used to set or clear a DDR bit (all "unaffected" bits would be set). It is recommended that all DDR bits in a port be written using a single-store instruction.

**CONTROL INSTRUCTIONS** - The control instructions control the MCU operations during program execution. Refer to table 5.

**ALPHABETICAL LISTING** - The complete instruction set is given in alphabetical order in table 6.

**OPCODE MAP SUMMARY** - Table 7 is an opcode map for the instructions used on the MCU.

**Table 1 :** Register/Memory Instructions.

| Function | Mnemonic | Immediate Op Code | Immediate # Bytes | Immediate # Cycles | Direct Op Code | Direct # Bytes | Direct # Cycles | Extended Op Code | Extended # Bytes | Extended # Cycles | Indexed (No Offset) Op Code | Indexed (No Offset) # Bytes | Indexed (No Offset) # Cycles | Indexed (8-Bit Offset) Op Code | Indexed (8-Bit Offset) # Bytes | Indexed (8-Bit Offset) # Cycles | Indexed (16-Bit Offset) Op Code | Indexed (16-Bit Offset) # Bytes | Indexed (16-Bit Offset) # Cycles |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Load A from Memory | LDA | A6 | 2 | 2 | B6 | 2 | 4 | C6 | 3 | 5 | F6 | 1 | 4 | E6 | 2 | 5 | D6 | 3 | 6 |
| Load X from Memory | LDX | AE | 2 | 2 | BE | 2 | 4 | CE | 3 | 5 | FE | 1 | 4 | EE | 2 | 5 | DE | 3 | 6 |
| Store A in Memory | STA | — | — |  | B7 | 2 | 5 | C7 | 3 | 6 | F7 | 1 | 5 | E7 | 2 | 6 | D7 | 3 | 7 |
| Store X in Memory | STX | — | i |  | BF | 2 | 5 | CF | 3 | 6 | FF | 1 | 5 | EF | 2 | 6 | DF | 3 | 7 |
| Add Memory to A | ADD | AB | 2 | 2 | BB | 2 | 4 | CB | 3 | 5 | FB | 1 | 4 | EB | 2 | 5 | DB | 3 | 6 |
| Add Memory and Carry to A | ADC | A9 | 2 | 2 | B9 | 2 | 4 | C9 | 3 | 5 | F9 | 1 | 4 | E9 | 2 | 5 | D9 | 3 | 6 |
| Subtract Memory | SUB | A0 | 2 | 2 | B0 | 2 | 4 | C0 | 3 | 5 | F0 | 1 | 4 | E0 | 2 | 5 | D0 | 3 | 6 |
| Subtract Memory from A with Borrow | SBC | A2 | 2 | 2 | B2 | 2 | 4 | C2 | 3 | 5 | F2 | 1 | 4 | E2 | 2 | 5 | D2 | 3 | 6 |
| AND Memory to A | AND | A4 | 2 | 2 | B4 | 2 | 4 | C4 | 3 | 5 | F4 | 1 | 4 | E4 | 2 | 5 | D4 | 3 | 6 |
| OR Memory with A | ORA | AA | 2 | 2 | BA | 2 | 4 | CA | 3 | 5 | FA | 1 | 4 | EA | 2 | 5 | DA | 3 | 6 |
| Exclusive OR Memory with A | EOR | A8 | 2 | 2 | BB | 2 | 4 | C8 | 3 | 5 | F8 | 1 | 4 | E8 | 2 | 5 | D8 | 3 | 6 |
| Arithmetic Compare A with Memory | CMP | A1 | 2 | 2 | B1 | 2 | 4 | C1 | 3 | 5 | F1 | 1 | 4 | E1 | 2 | 5 | D1 | 3 | 6 |
| Arithmetic Compare X with Memory | CPX | A3 | 2 | 2 | B3 | 2 | 4 | C3 | 3 | 5 | F3 | 1 | 4 | E3 | 2 | 5 | D3 | 3 | 6 |
| Bit Test Memory with A (Logical Compare) | BIT | A5 | 2 | 2 | B5 | 2 | 4 | C5 | 3 | 5 | F5 | 1 | 4 | E5 | 2 | 5 | D5 | 3 | 6 |
| Jump Unconditional | JMP | : |  |  | BC | 2 | 3 | CC | 3 | 4 | FC | 1 | 3 | EC | 2 | 4 | DC | 3 | 5 |
| Jump to Subroutine | JSR | — | . |  | BD | 2 | 7 | CD | 3 | 8 | FD | 1 | 7 | ED | 2 | 8 | DD | 3 | 9 |

SGS-THOMSON
MICROELECTRONICS

**Table 2 :** Read-Modify-Write Instructions.

Addressing Modes

| Function | Mnemonic | Inherent (A) | | | Inherent (X) | | | Direct | | | Indexed (No Offset) | | | Indexed (8 Bit Offset) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Op Code | # Bytes | # Cycles | Op Code | # Bytes | # Cycles | Op Code | # Bytes | # Cycles | Op Code | # Bytes | # Cycles | Op Code | # Bytes | # Cycles |
| Increment | INC | 4C | 1 | 4 | 5C | 1 | 4 | 3C | 2 | 6 | 7C | 1 | 6 | 6C | 2 | 7 |
| Decrement | DEC | 4A | 1 | 4 | 5A | 1 | 4 | 3A | 2 | 6 | 7A | 1 | 6 | 6A | 2 | 7 |
| Clear | CLR | 4F | 1 | 4 | 5F | 1 | 4 | 3F | 2 | 6 | 7F | 1 | 6 | 6F | 2 | 7 |
| Complement | COM | 43 | 1 | 4 | 53 | 1 | 4 | 33 | 2 | 6 | 73 | 1 | 6 | 63 | 2 | 7 |
| Negate (2's Complement) | NEG | 40 | 1 | 4 | 50 | 1 | 4 | 30 | 2 | 6 | 70 | 1 | 6 | 60 | 2 | 7 |
| Rotate Left Thru Carry | ROL | 49 | 1 | 4 | 59 | 1 | 4 | 39 | 2 | 6 | 79 | 1 | 6 | 69 | 2 | 7 |
| Rotate Right Thru Carry | ROR | 46 | 1 | 4 | 56 | 1 | 4 | 36 | 2 | 6 | 76 | 1 | 6 | 66 | 2 | 7 |
| Logical Shift Left | LSL | 48 | 1 | 4 | 58 | 1 | 4 | 38 | 2 | 6 | 78 | 1 | 6 | 68 | 2 | 7 |
| Logical Shift Right | LSR | 44 | 1 | 4 | 54 | 1 | 4 | 34 | 2 | 6 | 74 | 1 | 6 | 64 | 2 | 7 |
| Arithmetic Shift Right | ASR | 47 | 1 | 4 | 57 | 1 | 4 | 37 | 2 | 6 | 77 | 1 | 6 | 67 | 2 | 7 |
| Test for Negative or Zero | TST | 4D | 1 | 4 | 5D | 1 | 4 | 3D | 2 | 6 | 7D | 1 | 6 | 6D | 2 | 7 |

**Table 3 :** Branch Instructions.

| Function | Mnemonic | Relative Addressing Mode | | |
|---|---|---|---|---|
| | | Op Code | # Bytes | # Cycles |
| Branch Always | BRA | 20 | 2 | 4 |
| Branch Never | BRN | 21 | 2 | 4 |
| Branch IFF Higher | BHI | 22 | 2 | 4 |
| Branch IFF Lower or Same | BLS | 23 | 2 | 4 |
| Branch IFF Carry Clear | BCC | 24 | 2 | 4 |
| (branch IFF higher or same) | (BHS) | 24 | 2 | 4 |
| Branch IFF Carry Set | BCS | 25 | 2 | 4 |
| (branch IFF lower) | (BLO) | 25 | 2 | 4 |
| Branch IFF Not Equal | BNE | 26 | 2 | 4 |
| Branch IFF Equal | BEQ | 27 | 2 | 4 |
| Branch IFF Half Carry Clear | BHCC | 28 | 2 | 4 |
| Branch IFF Half Carry Set | BHCS | 29 | 2 | 4 |
| Branch IFF Plus | BPL | 2A | 2 | 4 |
| Branch IFF Minus | BMI | 2B | 2 | 4 |
| Branch IFF interrupt mask bit is clear. | BMC | 2C | 2 | 4 |
| Branch IFF interrupt mask bit is set. | BMS | 2D | 2 | 4 |
| Branch IFF interrupt line is low. | BIL | 2E | 2 | 4 |
| Branch IFF interrupt line is high. | BIH | 2F | 2 | 4 |
| Branch to Subroutine | BSR | AD | 2 | 8 |

**Table 4** : Bit Manipulation Instructions.

| Function | Mnemonic | Addressing Modes | | | | | |
|---|---|---|---|---|---|---|---|
| | | Bit Set/clear | | | Bit Test and Branch | | |
| | | Op Code | # Bytes | # Cycles | Op Code | # Bytes | # Cycles |
| Branch IFF Bit n is set | BRSET n (n = 0… 7) | | | | $2 \bullet n$ | 3 | 10 |
| Branch IFF Bit n is clear | BRCLR n (n = 0… 7) | | | | $01 + 2 \bullet n$ | 3 | 10 |
| Set Bit n | BSET n (n = 0… 7) | $10 + 2 \bullet n$ | 2 | 7 | | | |
| Clear Bit n | BCLR n (n = 0… 7) | $11 + 2 \bullet n$ | 2 | 7 | | | |

**SGS-THOMSON**
**MICROELECTRONICS**

**Table 5 :** Control Instructions.

| Function | Mnemonic | Inherent | | |
|---|---|---|---|---|
| | | Op Code | # Bytes | # Cycles |
| Transfer A to X | TAX | 97 | 1 | 2 |
| Transfer X to A | TXA | 9F | 1 | 2 |
| Set Carry Bit | SEC | 99 | 1 | 2 |
| Clear Carry Bit | CLC | 98 | 1 | 2 |
| Set Interrupt Mask Bit | SEI | 9B | 1 | 2 |
| Clear Interrupt Mask Bit | CLI | 9A | 1 | 2 |
| Software Interrupt | SWI | 83 | 1 | 11 |
| Return from Subroutine | RTS | 81 | 1 | 6 |
| Return from Interrupt | RTI | 80 | 1 | 9 |
| Reset Stack Pointer | RSP | 9C | 1 | 2 |
| No-operation | NOP | 9D | 1 | 2 |

**Table 6** : Instruction Set.

| Mnem | Addressing Modes | | | | | | | | | | Condition Code | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Inherent | Immediate | Direct | Extended | Relative | Indexed (no offset) | Indexed (8 Bits) | Indexed (16 Bits) | Bit Set/Clear | Bit Test & Branch | H | I | N | Z | C |
| ADC | | X | X | X | | X | X | X | | | ∧ | ● | ∧ | ∧ | ∧ |
| ADD | | X | X | X | | X | X | X | | | ∧ | ● | ∧ | ∧ | ∧ |
| AND | | X | X | X | | X | X | X | | | ● | ● | ∧ | ∧ | ● |
| ASL | X | | X | | | X | X | | | | ● | ● | ∧ | ∧ | ∧ |
| ASR | X | | X | | | X | X | | | | ● | ● | ∧ | ∧ | ∧ |
| BCC | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BCLR | | | | | | | | | X | | ● | ● | ● | ● | ● |
| BCS | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BEQ | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BHCC | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BHCS | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BHI | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BHS | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BIH | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BIL | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BIT | | X | X | X | | X | X | X | | | ● | ● | ∧ | ∧ | ● |
| BLO | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BLS | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BMC | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BMI | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BMS | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BNE | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BPL | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BRA | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BRN | | | | | X | | | | | | ● | ● | ● | ● | ● |
| BRCLR | | | | | | | | | | X | ● | ● | ● | ● | ∧ |
| BRSET | | | | | | | | | | X | ● | ● | ● | ● | ∧ |
| BSET | | | | | | | | | X | | ● | ● | ● | ● | ● |
| BSR | | | | | X | | | | | | ● | ● | ● | ● | ● |
| CLL | X | | | | | | | | | | ● | ● | ● | ● | 0 |

**Table 6** : Instruction Set (continued).

| Mnem | Inherent | Immediate | Direct | Extended | Relative | Indexed (no offset) | Indexed (8 Bits) | Indexed (16 Bits) | Bit Set/clear | Bit Test & Branch | H | I | N | Z | C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CLI | X | | | | | | | | | | ● | 0 | ● | ● | ● |
| CLR | X | | X | | | X | X | | | | ● | ● | 0 | 1 | ● |
| CMP | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ^ |
| COM | X | | X | | | X | X | | | | ● | ● | ^ | ^ | 1 |
| CPX | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ^ |
| DEC | X | | X | | | X | X | | | | ● | ● | ^ | ^ | ● |
| EOR | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ● |
| INC | X | | X | | | X | X | | | | ● | ● | ^ | ^ | ● |
| JMP | | | X | X | | X | X | X | | | ● | ● | ● | ● | ● |
| JSR | | | X | X | | X | X | X | | | ● | ● | ● | ● | ● |
| LDA | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ● |
| LDX | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ● |
| LSL | X | | X | | | X | X | | | | ● | ● | ^ | ^ | ^ |
| LSR | X | | X | | | X | X | | | | ● | ● | 0 | ^ | ^ |
| NEQ | X | | X | | | X | X | | | | ● | ● | ^ | ^ | ^ |
| NOP | X | | | | | | | | | | ● | ● | ● | ● | ● |
| ORA | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ● |
| ROL | X | | X | | | X | X | | | | ● | ● | ^ | ^ | ^ |
| RSP | X | | | | | | | | | | ● | ● | ● | ● | ● |
| RTI | X | | | | | | | | | | ? | ? | ? | ? | ? |
| RTS | X | | | | | | | | | | ● | ● | ● | ● | ● |
| SBC | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ^ |
| SEC | X | | | | | | | | | | ● | ● | ● | ● | 1 |
| SEI | X | | | | | | | | | | ● | 1 | ● | ● | ● |
| STA | | | X | X | | X | X | X | | | ● | ● | ^ | ^ | ● |
| STX | | | X | X | | X | X | X | | | ● | ● | ^ | ^ | ● |
| SUB | | X | X | X | | X | X | X | | | ● | ● | ^ | ^ | ^ |
| SWI | X | | | | | | | | | | ● | 1 | ● | ● | ● |
| TAX | X | | | | | | | | | | ● | ● | ● | ● | ● |
| TST | X | | X | | | X | X | | | | ● | ● | ^ | ^ | ● |
| TXA | X | | | | | | | | | | ● | ● | ● | ● | ● |

Condition Code Symbols :
H
I  Interrupt Mask
N

Z
C        Half Carry (from bit 3)
^  Test and Set if True, Cleared Otherwise
● Not Affected  Negative (sign bit)

**SGS-THOMSON**
**MICROELECTRONICS**

**HMOS 6805 FAMILY**

| Features | EF6805P2 | EF6805P6 | EF6805R2 | EF6805R3 | EF6805U2 | EF6805U3 |
|---|---|---|---|---|---|---|
| Technology | HMOS | HMOS | HMOS | HMOS | HMOS | HMOS |
| Number of Pins | 28 | 28 | 40 | 40 | 40 | 40 |
| On-chip RAM (bytes) | 64 | 64 | 64 | 112 | 64 | 112 |
| On-chip User ROM (bytes) | 1100 | 1796 | 2048 | 3776 | 2048 | 3776 |
| External Bus | None | None | None | None | None | None |
| Bidirectional I/O Lines | 20 | 20 | 24 | 24 | 24 | 24 |
| Unidirectional I/O Lines | None | None | 6 Inputs | 6 Inputs | 8 Inputs | 8 Inputs |
| Other I/O Features | Timer | Timer | Timer, A/D | Timer, A/D | Timer | Timer |
| External Interrupt Inputs | 1 | 1 | 2 | 2 | 2 | 2 |
| STOP and WAIT | No | No | No | No | No | No |

**Table 7 :** 6805 HMOS Family Opcode MAP.

| Low \ Hi | Bit Manip. BTB (0) | Bit Manip. BSC (1) | Branch REL (2) | R-M-W DIR (3) | R-M-W INH (4) | R-M-W INH (5) | R-M-W IX1 (6) | R-M-W IX (7) | Control INH (8) | Control INH (9) | Reg/Mem IMM (A) | Reg/Mem DIR (B) | Reg/Mem EXT (C) | Reg/Mem IX2 (D) | Reg/Mem IX1 (E) | Reg/Mem IX (F) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | BRSET0 BTB | BSET0 BSC | BRA REL | NEG DIR | NEGA INH | NEGX INH | NEG IX1 | NEG IX | RTI INH | | SUB IMM | SUB DIR | SUB EXT | SUB IX2 | SUB IX1 | SUB IX |
| 1 | BRCLR0 BTB | BCLR0 BSC | BRN REL | | | | | | RTS INH | | CMP IMM | CMP DIR | CMP EXT | CMP IX2 | CMP IX1 | CMP IX |
| 2 | BRSET1 BTB | BSET1 BSC | BHI REL | | | | | | | | SBC IMM | SBC DIR | SBC EXT | SBC IX2 | SBC IX1 | SBC IX |
| 3 | BRCLR1 BTB | BCLR1 BSC | BLS REL | COM DIR | COMA INH | COMX INH | COM IX1 | COM IX | SWI INH | | CPX IMM | CPX DIR | CPX EXT | CPX IX2 | CPX IX1 | CPX IX |
| 4 | BRSET2 BTB | BSET2 BSC | BCC REL | LSR DIR | LSRA INH | LSRX INH | LSR IX1 | LSR IX | | | AND IMM | AND DIR | AND EXT | AND IX2 | AND IX1 | AND IX |
| 5 | BRCLR2 BTB | BCLR2 BSC | BCS REL | | | | | | | | BIT IMM | BIT DIR | BIT EXT | BIT IX2 | BIT IX1 | BIT IX |
| 6 | BRSET3 BTB | BSET3 BSC | BNE REL | ROR DIR | RORA INH | RORX INH | ROR IX1 | ROR IX | | | LDA IMM | LDA DIR | LDA EXT | LDA IX2 | LDA IX1 | LDA IX |
| 7 | BRCLR3 BTB | BCLR3 BSC | BEQ REL | ASR DIR | ASRA INH | ASRX INH | ASR IX1 | ASR IX | | TAX INH | | STA DIR | STA EXT | STA IX2 | STA IX1 | STA IX |
| 8 | BRSET4 BTB | BSET4 BSC | BHCC REL | LSL DIR | LSLA INH | LSLX INH | LSL IX1 | LSL IX | | CLC INH | EOR IMM | EOR DIR | EOR EXT | EOR IX2 | EOR IX1 | EOR IX |
| 9 | BRCLR4 BTB | BCLR4 BSC | BHCS REL | ROL DIR | ROLA INH | ROLX INH | ROL IX1 | ROL IX | | SEC INH | ADC IMM | ADC DIR | ADC EXT | ADC IX2 | ADC IX1 | ADC IX |
| A | BRSET5 BTB | BSET5 BSC | BPL REL | DEC DIR | DECA INH | DECX INH | DEC IX1 | DEC IX | | CLI INH | ORA IMM | ORA DIR | ORA EXT | ORA IX2 | ORA IX1 | ORA IX |
| B | BRCLR5 BTB | BCLR5 BSC | BMI REL | | | | | | | SEI INH | ADD IMM | ADD DIR | ADD EXT | ADD IX2 | ADD IX1 | ADD IX |
| C | BRSET6 BTB | BSET6 BSC | BMC REL | INC DIR | INCA INH | INCX INH | INC IX1 | INC IX | | RSP INH | | JMP DIR | JMP EXT | JMP IX2 | JMP IX1 | JMP IX |
| D | BRCLR6 BTB | BCLR6 BSC | BMS REL | TST DIR | TSTA INH | TSTX INH | TST IX1 | TST IX | | NOP INH | BSR REL | JSR DIR | JSR EXT | JSR IX2 | JSR IX1 | JSR IX |
| E | BRSET7 BTB | BSET7 BSC | BIL REL | | | | | | | | LDX IMM | LDX DIR | LDX EXT | LDX IX2 | LDX IX1 | LDX IX |
| F | BRCLR7 BTB | BCLR7 BSC | BIH REL | CLR DIR | CLRA INH | CLRX INH | CLR IX1 | CLR IX | | TXA INH | | STX DIR | STX EXT | STX IX2 | STX IX1 | STX IX |

**LEGEND**

- Opcode in Hexadecimal
- Opcode in Binary
- Address Mode

```
      F          0
     1111       0000
   4  SUB  1     IX
```

- # of Cycles
- Mnemonic
- Bytes

**Abbreviations for Address Modes**

| | |
|---|---|
| INH | Inherent |
| IMM | Immediate |
| DIR | Direct |
| EXT | Extended |
| REL | Relative |
| BSC | Bit Set/Clear |
| BTB | Bit Test and Branch |
| IX | Indexed (No Offset) |
| IX1 | Indexed, 1 Byte (8-Bit) Offset |
| IX2 | Indexed, 2 Byte (16-Bit) Offset |

**SGS-THOMSON MICROELECTRONICS**

## PACKAGE MECHANICAL DATA

40 Pin Plastic Dual In Line Package (PDIP)



| Dim. | mm | | | inches | | |
|------|-----|-----|-------|-------|-------|-------|
| | Min | Typ | Max | Min | Typ | Max |
| A | 2.2 | | 4.8 | 0.086 | | 0.189 |
| A1 | 0.51 | | 1.77 | 0.010 | | 0.069 |
| B | 0.38 | | 0.58 | 0.015 | | 0.023 |
| B1 | 0.97 | | 1.52 | 0.055 | | 0.065 |
| C | 0.2 | | 0.3 | 0.008 | | 0.009 |
| D | 50.30 | | 52.22 | 1.980 | | 20.56 |
| D1 | - | - | - | - | - | - |
| E | | | 16.3 | | | 0.641 |
| E1 | | 12.9 | | | 0.508 | |
| K1 | – | – | – | – | – | – |
| K2 | – | – | – | – | – | – |
| L | 3.18 | | 4.44 | 1.25 | | 0.174 |
| e1 | | 2.54 | | | 0.10 | |
| **Number of Pins** | | | | | | |
| N | 40 | | | | | |

VR001725
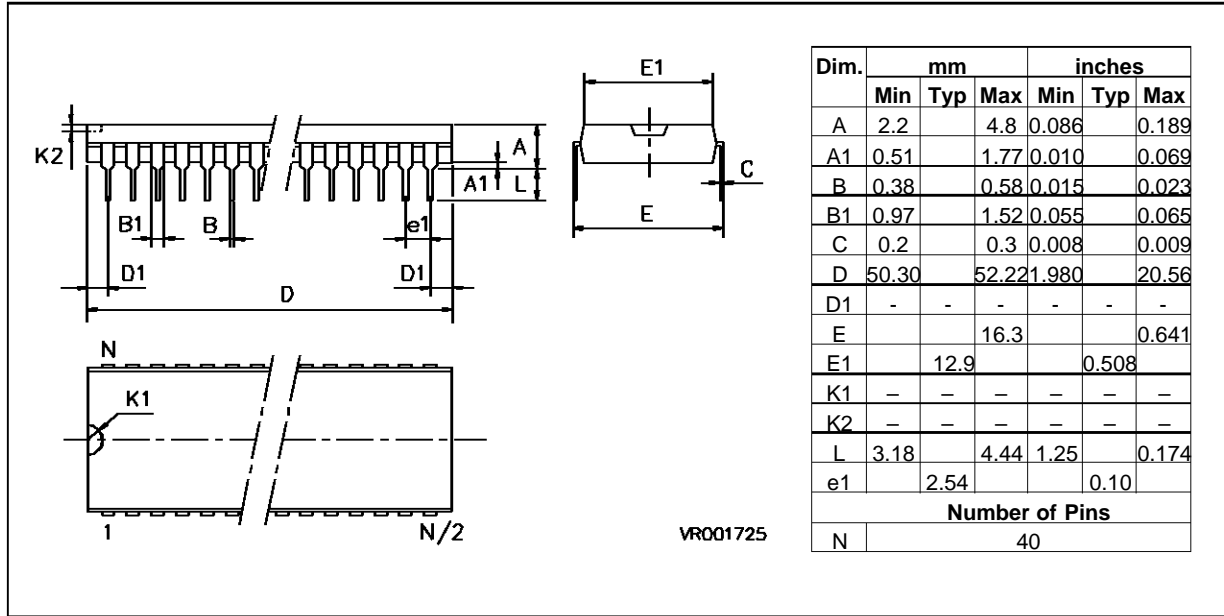
44 Pin Plastic Quad Package (PLCC)



| Dim. | mm | | | inches | | |
|------|-------|-------|-------|-------|-------|-------|
| | Min | Typ | Max | Min | Typ | Max |
| A | 4.2 | | 5.08 | 0.165 | | 0.200 |
| A1 | 0.64 | | | 0.020 | | |
| A3 | 2.29 | | 3.30 | 0.090 | | 0.130 |
| B | 0.331 | - | - | - | - | - |
| B1 | 0.661 | - | - | - | - | - |
| D | 17.40 | | 17.65 | 0.685 | | 0.695 |
| D1 | 16.51 | | 16.66 | 0.650 | | 0.656 |
| D3 | | 12.70 | | | 0.500 | |
| E | 17.40 | | 17.65 | 0.685 | | 0.695 |
| E1 | 16.51 | | 16.66 | 0.650 | | 0.656 |
| E3 | | 12.70 | | | 0.500 | |
| K1 | - | - | - | - | - | - |
| e | | 1.27 | | | 0.050 | |
| **Number of Pins** | | | | | | |
| N | 44 | | | | | |
| ND | 11 | | | | | |

VR001534
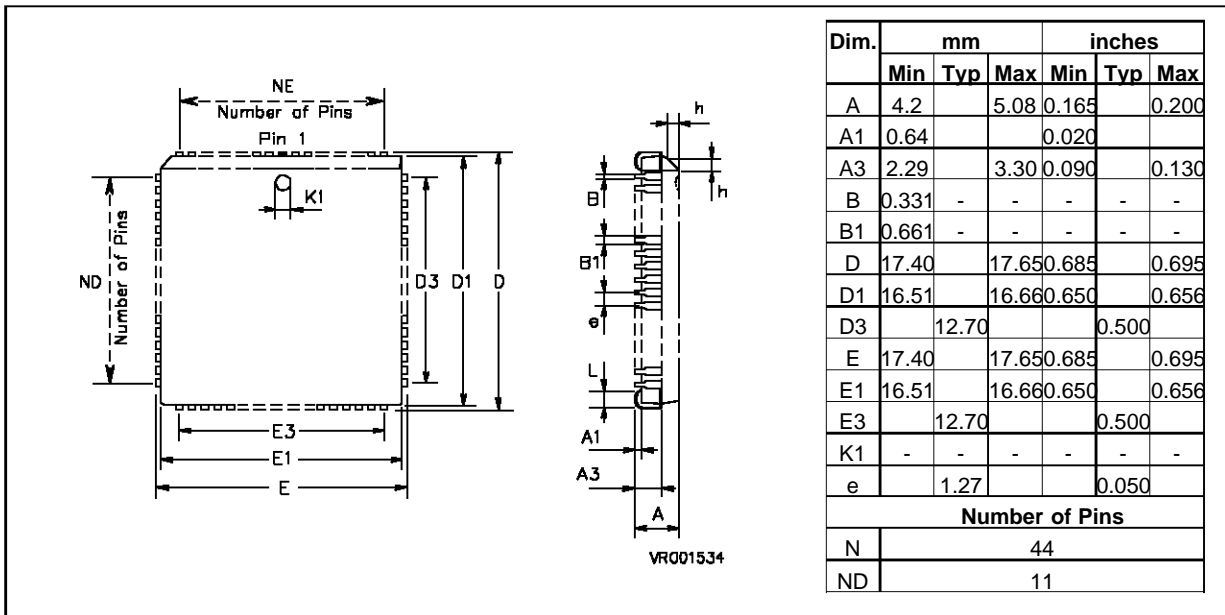
## ORDERING INFORMATION

The information required when ordering a custom MCU is listed below. The ROM program may be transmitted to SGS-THOMSON on EPROM(s) or an EFDOS/MDOS* disk file.

To initiate a ROM pattern for the MCU, it is necessary to first contact your local SGS-THOMSON representative or distributor.

### EPROMs

One 2716 or 2732 type EPROMs, programmed with the customer program (positive logic sense for ad-
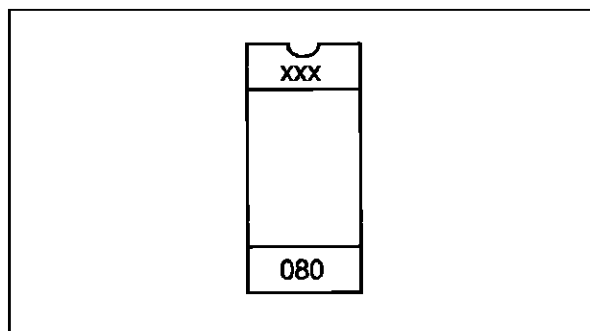


XXX = Customer ID)

dress and data), may be submitted for pattern generation.

After the EPROM is marked, it should be placed in conductive IC carriers and securely packed. Do not use styrofoam.

## VERIFICATION MEDIA

All original pattern media (EPROMs or floppy disk) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked and the verification form completed, signed, and returned to SGS-THOMSON. The signed verification form constitutes the contractual agreement for creation of the customer mask. If desired, SGS-THOM-

SON will program on blank EPROM from the data file used to create the custom mask and aid in the verification process.

### ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask but are for the purpose of ROM verification only. For expediency they are usually unmarked, packaged in ceramic, and tested only at room temperature and 5 volts. These RVUs are included in the mask change and are not production parts. The RVUs are thus not guaranteed by SGS THOMSON. Quality Assurance, and should be discarded after verification is completed.

## FLEXIBLE DISKS

The disk media submitted must be single-sided, EF-DOS/MDOS* compatible floppies.

The customer must write the binary file name and company name on the disk with a felt-tip-pen. The minimum EFDOS/MDOS* system files, as well as the absolute binary object file (Filename .LO type of file) from the 6805 cross assembler, must be on the disk. An object file made from a memory dump using the ROLLOUT command is also acceptable. Consider submitting a source listing as well as the following files : filename .LX (DEVICE/EXORciser loadable format) and filename .SA (ASCII Source Code). These files will of course be kept confidential and are used 1) to speed up the process in-house if any problems arise, and 2) to speed up the user-to-factory interface if the user finds any software errors and needs assistance quickly from SGS-THOMSON factory representatives.

EFDOS is SGS-THOMSON Disk Operating System available on development systems such as DE-VICE...

MDOS* is MOTOROLA's Disk Operating System available on development systems such as EXORciser...

* Requires prior factory approval.

Whenever ordering a custom MCU is required, please contact your local SGS-THOMSON representative or SGS-THOMSON distributor and/or complete and send the attached "MCU customer ordering sheet" to your local SGS-THOMSON Microelectronics representative.

## ORDER CODES

```
EF6805U3 │ P │ V │
```

Device

Package

Screen level
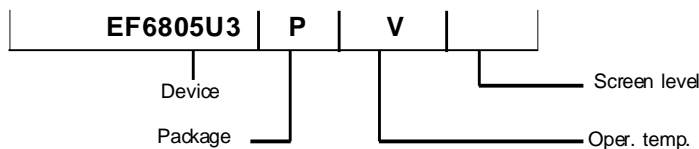
Oper. temp.

The table below horizontally shows all available suffix combinations for package, operating temperature and screening level. Other possibilities on request.

| Device | Package | | | | | Oper. Temp. | | | Screening Level | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C | J | P | E | FN | L* | V | T | Std | D | G/B | B/B |
| **EF6805U3** | | | X | | X | X | X | X | X | X | | |
| | | | | | | | | | | | | |
| | | | | | | | | | | | | |
| **Examples :** EF6805U3P, EF6805U3FN, EF6805U3PV, EF6805U3FNV | | | | | | | | | | | | |

**Package : C :** Ceramic DIL, **J :** Cerdip DIL, **P :** Plastic DIL, **E :** LCCC, **FN :** PLCC
**Oper. temp. : L* :** 0°C to + 70°C, **V :** – 40 °C to + 85°C, **T :** – 40°C to + 105°C, *** :** may be omitted.
**Screening level : Std :** (no-end suffix), **D :** NFC 96883 level D,
EXORciser is a registered trademark of MOTOROLA Inc.