National Semiconductor

COP8780C/COP8781C/COP8782C
Single-Chip EPROM/OTP Microcontrollers

General Description

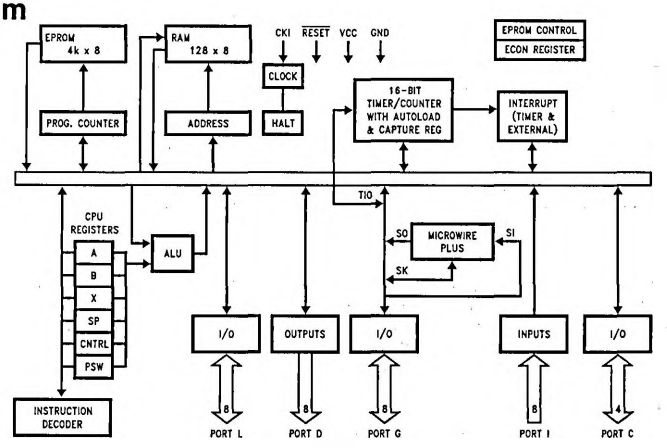
The COP8780C, COP8781C and COP8782C are members of the COPS™ 8-bit microcontroller family. They are fully static microcontrollers, fabricated using double-metal, double poly silicon gate microCMOS EPROM technology. These devices are available as UV erasable or One Time Programmable (OTP). These low cost microcontrollers are complete microcomputers containing all system timing, interrupt logic, EPROM, RAM, and I/O necessary to implement dedicated control functions in a variety of applications. Features include an 8-bit memory mapped architecture, MICROWIRE/PLUS™ serial I/O, a 16-bit timer/counter with associated 16-bit autoreload/capture register, and a multi-sourced interrupt. Each I/O pin has software selectable options to adapt the device to the specific application. These devices operate over a voltage range of 4.5V to 6.0V. An efficient, regular instruction set operating at a 1 μs instruction cycle rate provides optimal throughput.

The COP8780C, COP8781C and COP8782C can be configured to EMULATE the COP880C, COP840C and COP820C microcontrollers.

Features

- Low cost 8-bit microcontroller
- Fully static CMOS
- 4096 x 8 on-chip UV erasable or OTP EPROM
- EPROM security
- 128 or 64 bytes of on-chip RAM, user configurable
- Crystal, RC or External Oscillator, user configurable
- 1 μs instruction time (10 MHz clock)
- Low current drain
- Extra-low current static HALT mode
- Single supply operation: 4.5V to 6.0V
- 8-bit stack pointer (stack in RAM)
- 16-bit read/write timer operates in a variety of modes
  - — PWM (Pulse Width Modulation) mode with 16-bit autoreload register
  - — External Event Counter mode, with selectable edge
  - — Input Capture mode (selectable edge) with 16-bit capture register
- Multi-source interrupt
  - — External interrupt with selectable edge
  - — Timer interrupt or capture interrupt
  - — Software interrupt
- Powerful instruction set, with most instructions single byte
- Many single byte, single cycle instructions
- BCD arithmetic instructions
- MICROWIRE/PLUS serial I/O
- Software selectable I/O options (TRI-STATE, push-pull, weak pull-up)
- Temperature ranges: −40°C to +85°C
- Schmitt trigger inputs on G port
- COP8780C EPROM Programming fully supported by different sources
- Packages:
  - — 44 PLCC, OTP, Emulates COP880C, 36 I/O pins
  - — 40 DIP, OTP, Emulates COP880C, 36 I/O pins
  - — 28 DIP, OTP, Emulates COP820C/840C/881C, 24 I/O pins
  - — 20 DIP, OTP, Emulates COP822C/842C, 16 I/O pins
  - — 28 SO, 20 SO, OTP
  - — 44 LDCC, UV Erasable
  - — 40 CERDIP, 28 CERDIP, 20 CERDIP, UV Erasable

Block Diagram



TL/DD/11299-1

FIGURE 1

COP8780C/COP8781C/COP8782C

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage (V_{CC})	7V
Programming Voltage V_{PP} (RESET pin) and ME (pin G6)	13.4V
Voltage at any Pin	−0.3V to V_{CC} + 0.3V
Total Current into V_{CC} Pin (Source)	50 mA
Total Current out of GND Pin (Sink)	60 mA
Storage Temperature Range	−65°C to +150°C

Note: *Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

DC Electrical Characteristics COP87XXC; −40°C ≤ T_A ≤ +85°C unless otherwise specified

Parameter	Condition	Min	Typ	Max	Units
Operating Voltage		4.5		6.0	V
Power Supply Ripple (Note 1)	Peak to Peak			0.1 V_{CC}	V
Supply Current					
CKI = 10 MHz (Note 2)	V_{CC} = 6V, t_c = 1 μs			21	mA
HALT Current (Note 3)	V_{CC} = 6V, CKI = 0 MHz			10	μA
Input Levels					
RESET, CKI					
Logic High		0.9 V_{CC}			V
Logic Low				0.1 V_{CC}	V
All Other Inputs					
Logic High		0.7 V_{CC}			V
Logic Low				0.2 V_{CC}	V
Hi-Z Input Leakage	V_{CC} = 6.0V	−2		+2	μA
Input Pullup Current	V_{CC} = 6.0V, V_{IN} = 0V	−40		−250	μA
G Port Input Hysteresis	(Note 6)		0.05 V_{CC}		V
Output Current Levels					
D Outputs					
Source	V_{CC} = 4.5V, V_{OH} = 3.8V	−0.4			mA
Sink	V_{CC} = 4.5V, V_{OL} = 1.0V	10			mA
All Others					
Source (Weak Pull-Up)	V_{CC} = 4.5V, V_{OH} = 3.2V	−10		−110	μA
Source (Push-Pull Mode)	V_{CC} = 4.5V, V_{OH} = 3.8V	−0.4			mA
Sink (Push-Pull Mode)	V_{CC} = 4.5V, V_{OL} = 0.4V	1.6			mA
TRI-STATE Leakage		−2.0		+2.0	μA
Allowable Sink/Source Current per Pin					
D Outputs (Sink)				15	mA
All Others				3	mA
Maximum Input Current (Notes 4, 6) without Latchup (Room Temp)	Room Temp			±200	mA
RAM Retention Voltage, Vr (Note 5)		2.0			V
Input Capacitance	(Note 6)			7	pF
Load Capacitance on D2	(Note 6)			1000	pF

Note 1: Rate of voltage change must be less than 0.5V/ms.

Note 2: Supply current is measured after running 2000 cycles with a square wave CKI input, CKO open, inputs at rails and outputs open.

Note 3: The HALT mode will stop CKI from oscillating in the RC and the crystal configurations. Halt test conditions: All Inputs tied to V_{CC}. L, C, and G port I/O's configured as outputs and programmed low; D outputs programmed low; the window for UV erasable packages is completely covered with an opaque cover to prevent light from falling onto the die during HALT mode test. Parameter refers to HALT mode entered via setting bit 7 of the G Port data register.

Note 4: Pins G6 and RESET are designed with a high voltage input network for factory testing. These pins allow input voltages greater than V_{CC} and the pins will have sink current to V_{CC} when biased at voltages greater than V_{CC} (the pins do not have source current when biased at a voltage below V_{CC}). The effective resistance to V_{CC} is 750Ω (typ). These two pins will not latch up. The voltage at the pins must be limited to less than 14V.

Note 5: To maintain RAM integrity, the voltage must not be dropped or raised instantaneously.

Note 6: Parameter characterized but not tested.

COP8780C/COP8781C/COP8782C

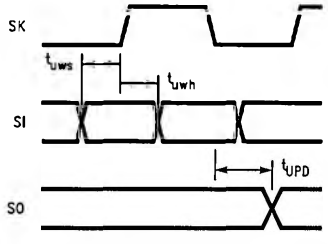
AC Electrical Characteristics $-40°C < T_A < +85°C$ unless otherwise specified

Parameter	Condition	Min	Typ	Max	Units
Instruction Cycle Time (t_c)					
Crystal/Resonator or External Clock	$V_{CC} \geq 4.5V$	1		DC	μs
R/C Oscillator Mode	$V_{CC} \geq 4.5V$	3		DC	μs
CKI Clock Duty Cycle (Note 7)	fr = Max	45		55	%
Rise Time (Note 7)	fr = 10 MHz Ext Clock			12	ns
Fall Time (Note 7)	fr = 10 MHz Ext Clock			8	ns
Inputs					
t_{SETUP}	$V_{CC} \geq 4.5V$	200			ns
t_{HOLD}	$V_{CC} \geq 4.5V$	60			ns
Output Propagation Delay	C_L = 100 pF, R_L = 2.2 kΩ				
t_{PD1}, t_{PD0}					
SO, SK	$V_{CC} \geq 4.5V$			0.7	μs
All Others	$V_{CC} \geq 4.5V$			1	μs
MICROWIRE™ Setup Time (t_{UWS})		20			ns
MICROWIRE Hold Time (t_{UWH})		56			ns
MICROWIRE Output					
Propagation Delay (t_{UPD})				220	ns
Input Pulse Width					
Interrupt Input High Time		1			t_c
Interrupt Input Low Time		1			t_c
Timer Input High Time		1			t_c
Timer Input Low Time		1			t_c
Reset Pulse Width		1.0			μs

Note 7: Parameter guaranteed by design, but not tested.

t_c = Instruction Cycle Time.

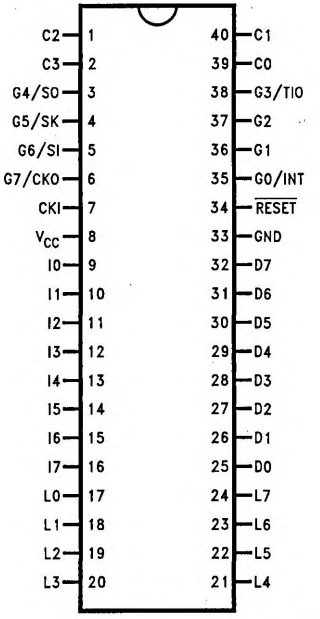
Timing Diagram



TL/DD/10802-2

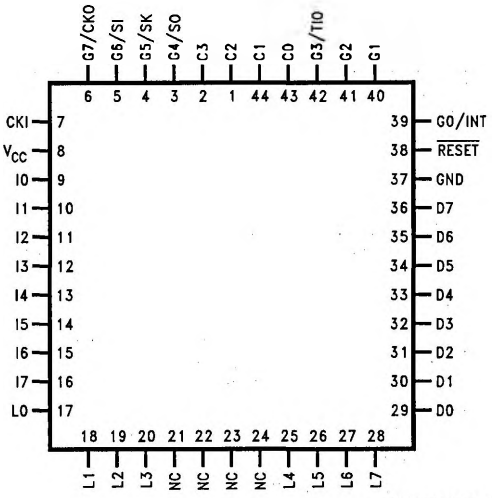
FIGURE 2. MICROWIRE/PLUS Timing

Connection Diagrams



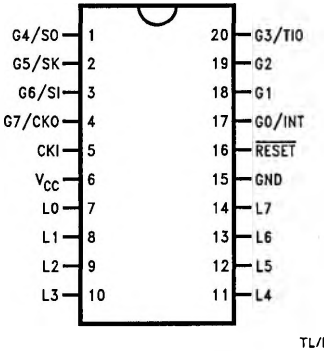
TL/DD/11299-3

Top View
COP8780CN, COP8780CJ



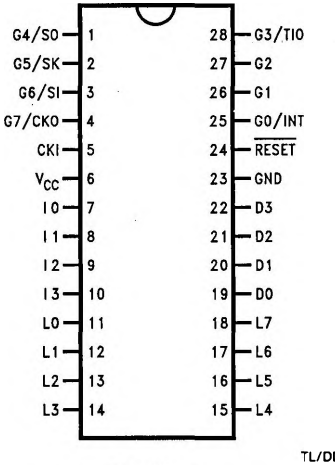
TL/DD/11299-4

Top View
COP8780CV, COP8780CEL



TL/DD/11299-5

Top View
COP8782CN, COP8782CWM,
COP8782CJ



TL/DD/11299-6

Top View
COP8781CN, COP8781CWM
COP8781CJ

FIGURE 3. Connection Diagrams

Pin Descriptions

V_{CC} and GND are the power supply pins.

CKI is the clock input. This can come from an external source, a R/C generated oscillator or a crystal (in conjunction with CKO). See Oscillator description.

$\overline{\text{RESET}}$ is the master reset input. See Reset description.

PORT I is an 8-bit Hi-Z input port. The 28-pin device does not have a full complement of PORT I pins. The unavailable pins are not terminated i.e., they are floating. A read operation for these unterminated pins will return unpredictable values. The user must ensure that the software takes this into account by either masking or restricting the accesses to bit operations. The unterminated PORT I pins will draw power only when addressed.

PORT L is an 8-bit I/O port.

PORT C is a 4-bit I/O port.

Three memory locations are allocated for the L, G and C ports, one each for data register, configuration register and the input pins. Reading bits 4–7 of the C-Configuration register, data register, and input pins returns undefined data.

There are two registers associated with the L and C ports: a data register and a configuration register. Therefore, each L and C I/O bit can be individually configured under software control as shown below:

Config.	Data	Ports L and C Setup
0	0	Hi-Z Input (TRI-STATE Output)
0	1	Input with Pull-Up (Weak One Output)
1	0	Push-Pull Zero Output
1	1	Push-Pull One Output

On the 20- and 28-pin parts, it is recommended that all bits of Port C be configured as outputs to minimize current.

PORT G is an 8-bit port with 6 I/O pins (G0–G5) and 2 input pins (G6, G7). All eight G-pins have Schmitt Triggers on the inputs.

There are two registers associated with the G port: a data register and a configuration register. Therefore, each G port bit can be individually configured under software control as shown below:

Config.	Data	Port G Setup
0	0	Hi-Z Input (TRI-STATE Output)
0	1	Input with Pull-Up (Weak One Output)
1	0	Push-Pull Zero Output
1	1	Push-Pull One Output

Since G6 and G7 are input only pins, any attempt by the user to configure them as outputs by writing a one to the configuration register will be disregarded. Reading the G6 and G7 configuration bits will return zeros. The device will be placed in the HALT mode by writing a one to the G7 bit in the G-port data register.

Six pins of Port G have alternate features:

G0 INTR (an external interrupt)

G3 TIO (timer/counter input/output)

G4 SO (MICROWIRE/PLUS serial data output)

G5 SK (MICROWIRE/PLUS clock I/O)

G6 SI (MICROWIRE/PLUS serial data input)

G7 CKO crystal oscillator output (selected by programming the ECON register) or HALT Restart/general purpose input

Pins G1 and G2 currently do not have any alternate functions.

PORT D is an 8-bit output port that is preset high when $\overline{\text{RESET}}$ goes low. Care must be exercised with the D2 pin operation. At reset, the external load on this pin must ensure that the output voltage stay above 0.7 V_{CC} to prevent the chip from entering special modes. Also, keep the external loading on D2 to less than 1000 pF.

Functional Description

Figure 1 shows the block diagram of the internal architecture. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device.

ALU AND CPU REGISTERS

The ALU can do an 8-bit addition, subtraction, logical or shift operation in one cycle time.

There are five CPU registers:

A is the 8-bit Accumulator register

PU is the upper 7 bits of the program counter (PC)

PL is the lower 8 bits of the program counter (PC)

B is the 8-bit address register, can be auto incremented or decremented.

X is the 8-bit alternate address register, can be incremented or decremented.

SP is the 8-bit stack pointer, which points to the subroutine/interrupt stack in RAM. The SP must be initialized with software (usually to RAM address 06F Hex with 128 bytes of on-chip RAM selected, or to RAM address 02F Hex with 64 bytes of on-chip RAM selected). The SP is used with the subroutine call and return instructions, and with the interrupts.

B, X and SP registers are mapped into the on-chip RAM. The B and X registers are used to address the on-chip RAM. The SP register is used to address the stack in RAM during subroutine calls and returns.

PROGRAM MEMORY

The device contains 4096 bytes of UV erasable or OTP EPROM memory. This memory is mapped in the program memory address space from 0000 to 0FFF Hex. The program memory may contain either instructions or data constants, and is addressed by the 15-bit program counter (PC). The program memory can be indirectly read by the LAID (Load Accumulator Indirect) instruction for table lookup of constant data.

All locations in the EPROM program memory will contain 0FF Hex (all 1's) after the device is erased. OTP parts are shipped with all locations already erased to 0FF Hex. Unused EPROM locations should always be programmed to 00 Hex so that the software trap can be used to halt runaway program operation.

The device can be configured to inhibit external reads of the program memory. This is done by programming the security bit in the ECON (EPROM configuration) register to zero. See the ECON REGISTER section for more details.

DATA MEMORY

The data memory address space includes on-chip RAM, I/O, and registers. Data memory is addressed directly by instructions, or indirectly by means of the B, X, or SP point-

Functional Description (Continued)

ers. The device can be configured to have either 64 or 128 bytes of RAM, depending on the value of the "RAM SIZE" bit in the ECON (EPROM CONFIGURATION) register. The sixteen bytes of RAM located at data memory address 0F0–0FF are designated as "registers". These sixteen registers can be decremented and tested with the DRSZ (Decrement Register and Skip if Zero) instruction.

The three pointers X, B, and SP are memory mapped into this register address space at addresses 0FC, 0FE, and 0FD respectively. The remaining registers are available for general usage.

Any bit of data memory can be directly set, reset or tested. All of the I/O registers and control registers (except A and PC) are memory mapped. Consequently, any of the I/O bits or control register bits can be directly and individually set, reset, or tested.

Note: RAM contents are undefined upon power-up.

ECON (EPROM CONFIGURATION) REGISTER

The ECON register is used to configure the user selectable clock, security, and RAM size options. The register can be programmed and read only in EPROM programming mode. Therefore, the register should be programmed at the same time as the program memory locations 0000 through 0FFF Hex. UV erasable parts are shipped with 0FF Hex in this register while the OTP parts are shipped with 07F Hex in this register. Erasing the EPROM program memory also erases the ECON register.

The device has a security feature which, when enabled, prevents reading of the EPROM program memory. The security bit in the ECON register determines whether security is enabled or disabled. If the security option is enabled, then any attempt to externally read the contents of the EPROM will result in the value E0 Hex being read from all program memory locations. If the security option is disabled, the contents of the internal EPROM may be read. The ECON register is readable regardless of the state of the security bit.

The format of the ECON register is as follows:

TABLE I

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
X	X	SECURITY	CKI 2	CKI 1	X	RAM SIZE	X

Bit 7 = X Don't care.

Bit 6 = X Don't care.

Bit 5 = 1 Security disabled. EPROM read and write are allowed.

= 0 Security enabled. EPROM read and write are not allowed.

Bits 4,3

= 1,1 External CKI option selected.

= 0,1 Not allowed.

= 1,0 RC oscillator option selected.

= 0,0 Crystal oscillator option selected.

Bit 2 = X Don't care.

Bit 1 = 1 Selects 128 byte RAM option. This emulates COP840 and COP880.

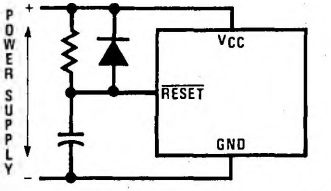
= 0 Selects 64 byte RAM option. This emulates COP820.

Bit 0 = X Don't care.

RESET

The $\overline{\text{RESET}}$ input when pulled low initializes the microcontroller. Initialization will occur whenever the $\overline{\text{RESET}}$ input is pulled low. Upon initialization, the Ports L, G and C are placed in the TRI-STATE mode and the Port D is set high. The PC, PSW and CNTRL registers are cleared. The data and configuration registers for Ports L, G and C are cleared.

The external RC network shown in *Figure 4* should be used to ensure that the $\overline{\text{RESET}}$ pin is held low until the power supply to the chip stabilizes.



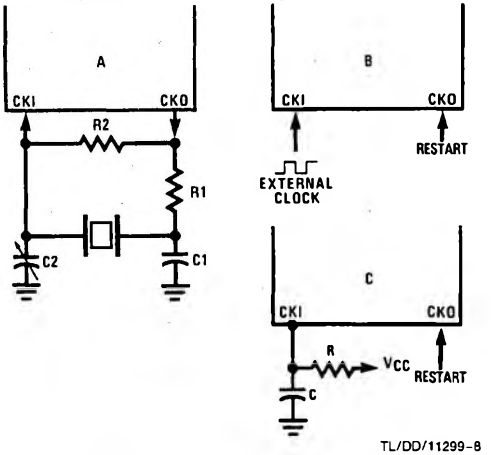
TL/DD/11299–7

RC ≥ 5X Power Supply Rise Time

FIGURE 4. Recommended Reset Circuit

OSCILLATOR CIRCUITS

Figure 5 shows the three clock oscillator configurations available for the device. The CKI 1 and CKI 2 bits in the ECON register are used to select the clock option. See the ECON REGISTER section for more details.



TL/DD/11299–8

FIGURE 5. Crystal, External and R-C Connection Diagrams

A. Crystal Oscillator

The device can be driven by a crystal clock. The crystal network is connected between the pins CKI and CKO.

Table II shows the component values required for various standard crystal frequencies.

B. External Oscillator

CKI can be driven by an external clock signal provided it meets the specified duty cycle, rise and fall times, and input levels. In External oscillator mode, G7 is available as a general purpose input and/or HALT restart control.

Functional Description (Continued)

TABLE II. Crystal Oscillator Configuration, T_A = 25°C

R1 (kΩ)	R2 (MΩ)	C1 (pF)	C2 (pF)	CKI Freq (MHz)	Conditions
0	1	30	30-36	10	V_{CC} = 5V
0	1	30	30-36	4	V_{CC} = 5V

TABLE III. RC Oscillator Configuration, T_A = 25°C

R (kΩ)	C (pF)	CKI Freq. (MHz)	Instr. Cycle (μs)	Conditions
3.3	82	2.2 to 2.7	3.7 to 4.6	V_{CC} = 5V
5.6	100	1.1 to 1.3	7.4 to 9.0	V_{CC} = 5V
6.8	100	0.9 to 1.1	8.8 to 10.8	V_{CC} = 5V

Note: (R/C Oscillator Configuration): 3k ≤ R ≤ 200k, 50 pF ≤ C ≤ 200 pF.

C. R/C Oscillator

CKI can be configured as a single pin RC controlled oscillator. In RC oscillator mode, G7 is available as a general purpose input and/or HALT restart control.

Table III shows the variation in the oscillator frequencies as functions of the component (R and C) values.

HALT MODE

The device supports a power saving mode of operation: HALT. The controller is placed in the HALT mode by setting the G7 data bit, alternatively the user can stop the clock input. (Stopping the clock input will draw more current than setting the G7 data bit.) In the HALT mode all internal processor activities including the clock oscillator are stopped. The fully static architecture freezes the state of the controller and retains all information until continuing. In the HALT mode, power requirements are minimal as it draws only leakage currents and output current. The applied voltage (V_{CC}) may be decreased down to Vr (minimum RAM retention voltage) without altering the state of the machine.

There are two ways to exit the HALT mode: via the $\overline{\text{RESET}}$ or by the G7 pin. A low on the $\overline{\text{RESET}}$ line reinitializes the microcontroller and starts execution from address 0000H. In external and RC oscillator modes, a low to high transition on the G7 pin also causes the microcontroller to come out of the HALT mode. Execution resumes at the address following the HALT instruction. Except for the G7 data bit, which gets reset, all RAM locations retain the values they had prior to execution of the "HALT" instruction. It is required that the first instruction following the "HALT" instruction be a "NOP" in order to synchronize the clock.

INTERRUPTS

The device has a sophisticated interrupt structure to allow easy interface to the real world. There are three possible interrupt sources, as shown below.

A maskable interrupt on external G0 input (positive or negative edge sensitive under software control)

A maskable interrupt on timer underflow or timer capture

A non-maskable software/error interrupt on opcode zero

INTERRUPT CONTROL

The GIE (global interrupt enable) bit enables the interrupt function. This is used in conjunction with ENI and ENTI to select one or both of the interrupt sources. This bit is reset when interrupt is acknowledged.

ENI and ENTI bits select external and timer interrupts respectively. Thus the user can select either or both sources to interrupt the microcontroller when GIE is enabled.

IEDG selects the external interrupt edge (0 = rising edge, 1 = falling edge). The user can get an interrupt on both rising and falling edges by toggling the state of IEDG bit after each interrupt.

IPND and TPND bits signal which interrupt is pending. After an interrupt is acknowledged, the user can check these two bits to determine which interrupt is pending. This permits the interrupts to be prioritized under software. The pending flags have to be cleared by the user. Setting the GIE bit high inside the interrupt subroutine allows nested interrupts.

The software interrupt does not reset the GIE bit. This means that the controller can be interrupted by other interrupt sources while servicing the software interrupt.

INTERRUPT PROCESSING

The interrupt, once acknowledged, pushes the program counter (PC) onto the stack and the stack pointer (SP) is decremented twice. The Global Interrupt Enable (GIE) bit is reset to disable further interrupts. The microcontroller then vectors to the address 00FFH and resumes execution from that address. This process takes 7 cycles to complete. At the end of the interrupt subroutine, any of the following three instructions return the processor back to the main program: RET, RETSK or RETI. Either one of the three instructions will pop the stack into the program counter (PC). The stack pointer is then incremented twice. The RETI instruction additionally sets the GIE bit to re-enable further interrupts.

Any of the three instructions can be used to return from a hardware interrupt subroutine. The RETSK instruction should be used when returning from a software interrupt subroutine to avoid entering an infinite loop.

Functional Description (Continued)

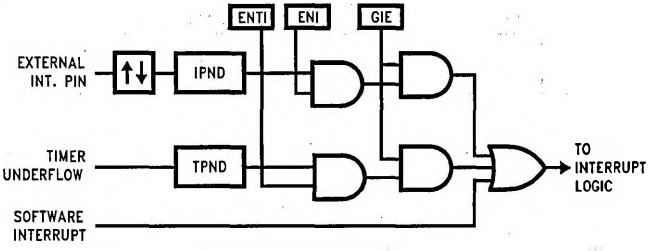


FIGURE 6. Interrupt Block Diagram

DETECTION OF ILLEGAL CONDITIONS

The device incorporates a hardware mechanism that allows it to detect illegal conditions which may occur from coding errors, noise and "brown out" voltage drop situations. Specifically, it detects cases of executing out of undefined EPROM area and unbalanced stack situations.

Reading an undefined EPROM location returns 00 (hexadecimal) as its contents. The opcode for a software interrupt is also "00". Thus a program accessing undefined EPROM will cause a software interrupt.

Reading an undefined RAM location returns an FF (hexadecimal). The subroutine stack on the device grows down for each subroutine call. By initializing the stack pointer to the top of RAM, the first unbalanced return instruction will cause the stack pointer to address undefined RAM. As a result the program will attempt to execute from FFFF (hexadecimal), which is an undefined EPROM location and will trigger a software interrupt.

MICROWIRE/PLUS

MICROWIRE/PLUS is a serial synchronous bidirectional communications interface. The MICROWIRE/PLUS capability enables the device to interface with any of National Semiconductor's MICROWIRE peripherals (i.e. A/D converters, display drivers, EEPROMS, etc.) and with other microcontrollers which support the MICROWIRE/PLUS interface. It consists of an 8-bit serial shift register (SIO) with serial data input (SI), serial data output (SO) and serial shift clock (SK). *Figure 7* shows the block diagram of the MICROWIRE/PLUS interface.

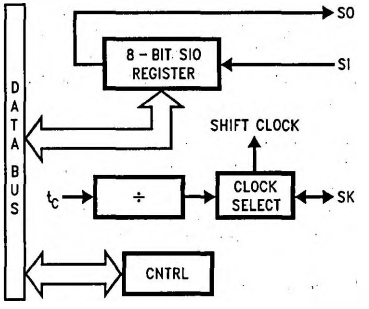


FIGURE 7. MICROWIRE/PLUS Block Diagram

The shift clock can be selected from either an internal source or an external source. Operating the MICROWIRE/PLUS interface with the internal clock source is called the Master mode of operation. Operating the MICROWIRE/PLUS interface with an external shift clock is called the Slave mode of operation.

The CNTRL register is used to configure and control the MICROWIRE/PLUS mode. To use the MICROWIRE/PLUS, the MSEL bit in the CNTRL register is set to one. The SK clock rate is selected by the two bits, SL0 and SL1, in the CNTRL register. Table IV details the different clock rates that may be selected.

TABLE IV

SL1	SL0	SK Cycle Time
0	0	$2t_C$
0	1	$4t_C$
1	x	$8t_C$

where,

t_C is the instruction cycle time.

MICROWIRE/PLUS OPERATION

Setting the BUSY bit in the PSW register causes the MICROWIRE/PLUS arrangement to start shifting the data. It gets reset when eight data bits have been shifted. The user may reset the BUSY bit by software to allow less than 8 bits to shift. The device may enter the MICROWIRE/PLUS mode either as a Master or as a Slave. *Figure 8* shows how two device microcontrollers and several peripherals may be interconnected using the MICROWIRE/PLUS arrangement.

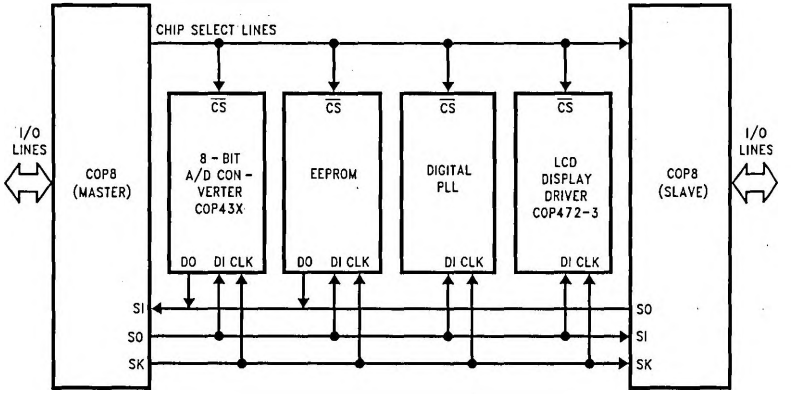
Master MICROWIRE/PLUS Operation

In the MICROWIRE/PLUS Master mode of operation the shift clock (SK) is generated internally by the device. The MICROWIRE/PLUS Master always initiates all data exchanges *(Figure 8)*. The MSEL bit in the CNTRL register must be set to enable the SO and SK functions on the G Port. The SO and SK pins must also be selected as outputs by setting appropriate bits in the Port G configuration register. Table V summarizes the bit settings required for Master mode of operation.

SLAVE MICROWIRE/PLUS OPERATION

In the MICROWIRE/PLUS Slave mode of operation the SK clock is generated by an external source. Setting the MSEL

Functional Description (Continued)



TL/DD/11299-11

FIGURE 8. MICROWIRE/PLUS Application

bit in the CNTRL register enables the SO and SK functions on the G Port. The SK pin must be selected as an input and the SO pin selected as an output pin by appropriately setting up the Port G configuration register. Table V summarizes the settings required to enter the Slave mode of operation.

The user must set the BUSY flag immediately upon entering the Slave mode. This will ensure that all data bits sent by the Master will be shifted properly. After eight clock pulses the BUSY flag will be cleared and the sequence may be repeated *(Figure 8)*.

TABLE V

G4 Config. Bit	G5 Config. Bit	G4 Fun.	G5 Fun.	G6 Fun.	Operation
1	1	SO	Int. SK	SI	MICROWIRE Master
0	1	TRI-STATE	Int. SK	SI	MICROWIRE Master
1	0	SO	Ext. SK	SI	MICROWIRE Slave
0	0	TRI-STATE	Ext. SK	SI	MICROWIRE Slave

TIMER/COUNTER

The device has a powerful 16-bit timer with an associated 16-bit register enabling it to perform extensive timer functions. The timer T1 and its register R1 are each organized as two 8-bit read/write registers. Control bits in the register CNTRL allow the timer to be started and stopped under software control. The timer-register pair can be operated in one of three possible modes. Table VI details various timer operating modes and their requisite control settings.

MODE 1. TIMER WITH AUTO-LOAD REGISTER

In this mode of operation, the timer T1 counts down at the instruction cycle rate. Upon underflow the value in the register R1 gets automatically reloaded into the timer which continues to count down. The timer underflow can be programmed to interrupt the microcontroller. A bit in the control register CNTRL enables the TIO (G3) pin to toggle upon timer underflows. This allows the generation of square-wave outputs or pulse width modulated outputs under software control *(Figure 9)*.

MODE 2. EXTERNAL COUNTER

In this mode, the timer T1 becomes a 16-bit external event counter. The counter counts down upon an edge on the TIO pin. Control bits in the register CNTRL program the counter to decrement either on a positive edge or on a negative edge. Upon underflow the contents of the register R1 are automatically copied into the counter. The underflow can also be programmed to generate an interrupt *(Figure 9)*.

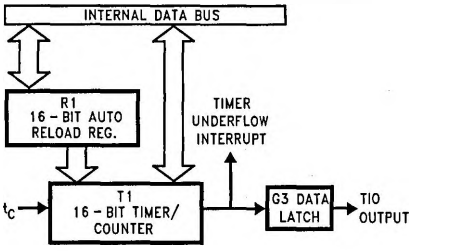
MODE 3. TIMER WITH CAPTURE REGISTER

Timer T1 can be used to precisely measure external frequencies or events in this mode of operation. The timer T1 counts down at the instruction cycle rate. Upon the occurrence of a specified edge on the TIO pin the contents of the timer T1 are copied into the register R1. Bits in the control register CNTRL allow the trigger edge to be specified either as a positive edge or as a negative edge. In this mode the user can elect to be interrupted on the specified trigger edge *(Figure 10)*.

TABLE VI. Timer Operating Modes

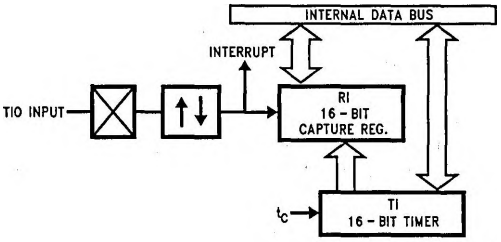
CNTRL Bits 7 6 5	Operation Mode	T Interrupt	Timer Counts On
0 0 0	External Counter w/Auto-Load Reg.	Timer Underflow	TIO Pos. Edge
0 0 1	External Counter w/Auto-Load Reg.	Timer Underflow	TIO Neg. Edge
0 1 0	Not Allowed	Not Allowed	Not Allowed
0 1 1	Not Allowed	Not Allowed	Not Allowed
1 0 0	Timer w/Auto-Load Reg.	Timer Underflow	t_C
1 0 1	Timer w/Auto-Load Reg./Toggle TIO Out	Timer Underflow	t_C
1 1 0	Timer w/Capture Register	TIO Pos. Edge	t_C
1 1 1	Timer w/Capture Register	TIO Neg. Edge	t_C

Functional Description (Continued)



TL/DD/11299-12

FIGURE 9. Timer/Counter Auto Reload Mode Block Diagram

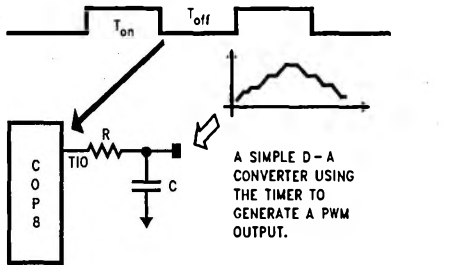


TL/DD/11299-13

FIGURE 10. Timer Capture Mode Block Diagram

TIMER PWM APPLICATION

Figure 11 shows how a minimal component D/A converter can be built out of the Timer-Register pair in the Auto-Reload mode. The timer is placed in the "Timer with auto reload" mode and the TIO pin is selected as the timer output. At the outset the TIO pin is set high, the timer T1 holds the on time and the register R1 holds the signal off time. Setting TRUN bit starts the timer which counts down at the instruction cycle rate. The underflow toggles the TIO output and copies the off time into the timer, which continues to run. By alternately loading in the on time and the off time at each successive interrupt a PWM frequency can be easily generated.



TL/DD/11299-14

FIGURE 11. Timer Application

Control Registers

CNTRL REGISTER (ADDRESS X'00EE)

The Timer and MICROWIRE/PLUS control register contains the following bits:

SL1 & SL0	Select the MICROWIRE/PLUS clock divide-by
IEDG	External interrupt edge polarity select (0 = rising edge, 1 = falling edge)
MSEL	Enable MICROWIRE/PLUS functions SO and SK
TRUN	Start/Stop the Timer/Counter (1 = run, 0 = stop)
TC3	Timer input edge polarity select (0 = rising edge, 1 = falling edge)
TC2	Selects the capture mode
TC1	Selects the timer mode

TC1	TC2	TC3	TRUN	MSEL	IEDG	S1	S0

Bit 7 Bit 0

PSW REGISTER (ADDRESS X'00EF)

The PSW register contains the following select bits:

GIE	Global interrupt enable
ENI	External interrupt enable
BUSY	MICROWIRE/PLUS busy shifting
IPND	External interrupt pending
ENTI	Timer interrupt enable
TPND	Timer interrupt pending
C	Carry Flag
HC	Half carry Flag

HC	C	TPND	ENTI	IPND	BUSY	ENI	GIE

Bit 7 Bit 0

Addressing Modes

REGISTER INDIRECT

This is the "normal" mode of addressing for the device. The operand is the memory location addressed by the B register or X register.

DIRECT

The instruction contains an 8-bit address field that directly points to the data memory location for the operand.

IMMEDIATE

The instruction contains an 8-bit immediate field as the operand.

REGISTER INDIRECT (AUTO INCREMENT AND DECREMENT)

This is a register indirect mode that automatically increments or decrements the B or X register after executing the instruction.

RELATIVE

This mode is used for the JP instruction, the instruction field is added to the program counter to get the new program location. JP has a range of −31 to +32 to allow a one byte relative jump (JP + 1 is implemented by a NOP instruction). There are no "pages" when using JP, all 15 bits of PC are used.

Memory Map

All RAM, ports and registers (except A and PC) are mapped into data memory address space.

RAM Select	Address	Contents
64 On-Chip RAM Bytes Selected by ECON reg.	00–2F 30–7F	48 On-Chip RAM Bytes Unused RAM Address Space (Reads as all 1's)
128 On-Chip RAM Bytes Selected by ECON reg.	00–6F 70–7F	112 On-chip RAM Bytes Unused RAM Address Space (Reads as all 1's)
	80 to BF	Expansion Space for On-Chip EERAM
	C0 to CF	Expansion Space for I/O and Registers
	D0 to DF D0 D1 D2 D3 D4 D5 D6 D7	On-Chip I/O and Registers Port L Data Register Port L Configuration Register Port L Input Pins (Read Only) Reserved for Port L Port G Data Register Port G Configuration Register Port G Input Pins (Read Only) Port I Input Pins (Read Only)
	D8 D9 DA DB DC DD–DF	Port C Data Register Port C Configuration Register Port C Input Pins (Read Only) Reserved for Port C Port D Data Register Reserved for Port D
	E0 to EF E0–E7 E8 E9 EA EB EC ED EE EF	On-Chip Functions and Registers Reserved for Future Parts Reserved MICROWIRE/PLUS Shift Register Timer Lower Byte Timer Upper Byte Timer Autoload Register Lower Byte Timer Autoload Register Upper Byte CNTRL Control Register PSW Register
	F0 to FF FC FD FE	On-Chip RAM Mapped as Registers X Register SP Register B Register

Reading unused memory locations below 7FH will return all ones. Reading other unused memory locations will return undefined data.

Instruction Set

REGISTER AND SYMBOL DEFINITIONS

Registers

A	8-bit Accumulator register
B	8-bit Address register
X	8-bit Address register
SP	8-bit Stack pointer register
PC	15-bit Program counter register
PU	upper 7 bits of PC
PL	lower 8 bits of PC
C	1-bit of PSW register for carry
HC	Half Carry
GIE	1-bit of PSW register for global interrupt enable

Symbols

[B]	Memory indirectly addressed by B register
[X]	Memory indirectly addressed by X register
Mem	Direct address memory or [B]
Meml	Direct address memory or [B] or Immediate data
Imm	8-bit Immediate data
Reg	Register memory: addresses F0 to FF (Includes B, X and SP)
Bit	Bit number (0 to 7)
←	Loaded with
⟷	Exchanged with

Instruction Set

ADD	add	A ← A + Meml
ADC	add with carry	A ← A + Meml + C, C ← Carry HC ← Half Carry
SUBC	subtract with carry	A ← A + $\overline{\text{Meml}}$ + C, C ← Carry HC ← Half Carry
AND	Logical AND	A ← A and Meml
OR	Logical OR	A ← A or Meml
XOR	Logical Exclusive-OR	A ← A xor Meml
IFEQ	IF equal	Compare A and Meml, Do next if A = Meml
IFGT	IF greater than	Compare A and Meml, Do next if A > Meml
IFBNE	IF B not equal	Do next if lower 4 bits of B ≠ Imm
DRSZ	Decrement Reg. ,skip if zero	Reg ← Reg − 1, skip if Reg goes to 0
SBIT	Set bit	1 to bit, Mem (bit= 0 to 7 immediate)
RBIT	Reset bit	0 to bit, Mem
IFBIT	If bit	If bit, Mem is true, do next instr.
X	Exchange A with memory	A ⟷ Mem
LD A	Load A with memory	A ← Meml
LD mem	Load Direct memory Immed.	Mem ← Imm
LD Reg	Load Register memory Immed.	Reg ← Imm
X	Exchange A with memory [B]	A ⟷ [B] (B ← B ± 1)
X	Exchange A with memory [X]	A ⟷ [X] (X ← X ± 1)
LD A	Load A with memory [B]	A ← [B] (B ← B ± 1)
LD A	Load A with memory [X]	A ← [X] (X ← X ± 1)
LD M	Load Memory Immediate	[B] ← Imm (B ← B ± 1)
CLRA	Clear A	A ← 0
INCA	Increment A	A ← A + 1
DECA	Decrement A	A ← A − 1
LAID	Load A indirect from ROM	A ← ROM(PU,A)
DCORA	DECIMAL CORRECT A	A ← BCD correction (follows ADC, SUBC)
RRCA	ROTATE A RIGHT THRU C	C → A7 → . . . → A0 → C
SWAPA	Swap nibbles of A	A7 . . . A4 ⟷ A3 . . . A0
SC	Set C	C ← 1, HC ← 1
RC	Reset C	C ← 0, HC ← 0
IFC	If C	If C is true, do next instruction
IFNC	If not C	If C is not true, do next instruction
JMPL	Jump absolute long	PC ← ii (ii = 15 bits, 0 to 32k)
JMP	Jump absolute	PC11..0 ← i (i = 12 bits)
JP	Jump relative short	PC ← PC + r (r is −31 to +32, not 1)
JSRL	Jump subroutine long	[SP] ← PL,[SP-1] ← PU,SP-2,PC ← ii
JSR	Jump subroutine	[SP] ← PL,[SP-1] ← PU,SP-2,PC11.. 0 ← i
JID	Jump indirect	PL ← ROM(PU,A)
RET	Return from subroutine	SP+2,PL ← [SP],PU ← [SP-1]
RETSK	Return and Skip	SP+2,PL ← [SP],PU ← [SP-1],Skip next instruction
RETI	Return from Interrupt	SP+2,PL ← [SP],PU ← [SP-1],GIE ← 1
INTR	Generate an interrupt	[SP] ← PL,[SP−1] ← PU,SP-2,PC ← 0FF
NOP	No operation	PC ← PC + 1

Bits 7–4

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
JP -15	JP -31	LD 0F0,#i	DRSZ 0F0	RRCA	RC	ADC A, #i	ADC A, [B]	IFBIT 0,[B]	*	LD B, 0F	IFBNE 0	JSR 0000–00FF	JMP 0000–00FF	JP + 17	INTR	0
JP -14	JP -30	LD 0F1,#i	DRSZ 0F1	*	SC	SUBC A, #i	SUBC A,[B]	IFBIT 1,[B]	*	LD B, 0E	IFBNE 1	JSR 0100–01FF	JMP 0100–01FF	JP + 18	JP + 2	1
JP -13	JP -29	LD 0F2,#i	DRSZ 0F2	X A, [X+]	X A, [B+]	IFEQ A, #i	IFEQ A,[B]	IFBIT 2,[B]	*	LD B, 0D	IFBNE 2	JSR 0200–02FF	JMP 0200–02FF	JP + 19	JP + 3	2
JP -12	JP -28	LD 0F3,#i	DRSZ 0F3	X A, [X−]	X A, [B−]	IFGT A, #i	IFGT A,[B]	IFBIT 3,[B]	*	LD B, 0C	IFBNE 3	JSR 0300–03FF	JMP 0300–03FF	JP + 20	JP + 4	3
JP -11	JP -27	LD 0F4,#i	DRSZ 0F4	*	LAID	ADD A, #i	ADD A,[B]	IFBIT 4,[B]	CLRA	LD B, 0B	IFBNE 4	JSR 0400–04FF	JMP 0400–04FF	JP + 21	JP + 5	4
JP -10	JP -26	LD 0F5,#i	DRSZ 0F5	*	JID	AND A, #i	AND A,[B]	IFBIT 5,[B]	SWAPA	LD B, 0A	IFBNE 5	JSR 0500–05FF	JMP 0500–05FF	JP + 22	JP + 6	5
JP -9	JP -25	LD 0F6,#i	DRSZ 0F6	X A, [X]	X A, [B]	XOR A, #i	XOR A,[B]	IFBIT 6,[B]	DCORA	LD B, 9	IFBNE 6	JSR 0600–06FF	JMP 0600–06FF	JP + 23	JP + 7	6
JP -8	JP -24	LD 0F7,#i	DRSZ 0F7	*	*	OR A, #i	OR A,[B]	IFBIT 7,[B]	*	LD B, 8	IFBNE 7	JSR 0700–07FF	JMP 0700–07FF	JP + 24	JP + 8	7
JP -7	JP -23	LD 0F8,#i	DRSZ 0F8	NOP	*	LD A, #i	IFC	SBIT 0,[B]	RBIT 0,[B]	LD B, 7	IFBNE 8	JSR 0800–08FF	JMP 0800–08FF	JP + 25	JP + 9	8
JP -6	JP -22	LD 0F9,#i	DRSZ 0F9	*	*	*	IFNC	SBIT 1,[B]	RBIT 1,[B]	LD B, 6	IFBNE 9	JSR 0900–09FF	JMP 0900–09FF	JP + 26	JP + 10	9
JP -5	JP -21	LD 0FA,#i	DRSZ 0FA	LD A, [X+]	LD A, [B+]	LD [B+],#i	INCA	SBIT 2,[B]	RBIT 2,[B]	LD B, 5	IFBNE 0A	JSR 0A00–0AFF	JMP 0A00–0AFF	JP + 27	JP + 11	A
JP -4	JP -20	LD 0FB,#i	DRSZ 0FB	LD A, [X−]	LD A, [B−]	LD [B−],#i	DECA	SBIT 3,[B]	RBIT 3,[B]	LD B, 4	IFBNE 0B	JSR 0B00–0BFF	JMP 0B00–0BFF	JP + 28	JP + 12	B
JP -3	JP -19	LD 0FC,#i	DRSZ 0FC	LD Md, #i	JMPL	X A,Md	*	SBIT 4,[B]	RBIT 4,[B]	LD B, 3	IFBNE 0C	JSR 0C00–0CFF	JMP 0C00–0CFF	JP + 29	JP +13	C
JP -2	JP -18	LD 0FD,#i	DRSZ 0FD	DIR	JSRL	LD A, Md	RETSK	SBIT 5,[B]	RBIT 5,[B]	LD B, 2	IFBNE 0D	JSR 0D00–0DFF	JMP 0D00–0DFF	JP + 30	JP +14	D
JP -1	JP -17	LD 0FE,#i	DRSZ 0FE	LD A, [X]	LD A, [B]	LD [B], #i	RET	SBIT 6, [B]	RBIT 6, [B]	LD B, 1	IFBNE 0E	JSR 0E00–0EFF	JMP 0E00–0EFF	JP + 31	JP +15	E
JP -0	JP -16	LD 0FF,#1	DRSZ 0FF	*	*	*	RETI	SBIT 7,[B]	RBIT 7,[B]	LD B, 0	IFBNE 0F	JSR 0F00–0FFF	JMP 0F00–0FFF	JP + 32	JP + 16	F

Bits 3–0

where, i is the immediate data Md is a directly addressed memory location * is an unused opcode (see following table)

Instruction Execution Time

Most instructions are single byte (with immediate addressing mode instruction taking two bytes).

Most single instructions take one cycle time to execute.

See the BYTES and CYCLES per INSTRUCTION table for details.

BYTES and CYCLES per INSTRUCTION

The following table shows the number of bytes and cycles for each instruction in the format of byte/cycle.

Arithmetic Instructions (Bytes/Cycles)

	[B]	Direct	Immed.
ADD	1/1	3/4	2/2
ADC	1/1	3/4	2/2
SUBC	1/1	3/4	2/2
AND	1/1	3/4	2/2
OR	1/1	3/4	2/2
XOR	1/1	3/4	2/2
IFEQ	1/1	3/4	2/2
IFGT	1/1	3/4	2/2
IFBNE	1/1		
DRSZ		1/3	
SBIT	1/1	3/4	
RBIT	1/1	3/4	
IFBIT	1/1	3/4	

Memory Transfer Instructions (Bytes/Cycles)

	Register Indirect		Direct	Immed.	Register Indirect Auto Incr & Decr		
	[B]	[X]			[B+, B−]	[X+, X−]	
X A,*	1/1	1/3	2/3		1/2	1/3	
LD A,*	1/1	1/3	2/3	2/2	1/2	1/3	
LD B,Imm				1/1			(If B < 16)
LD B,Imm				2/3			(If B > 15)
LD Mem,Imm			3/3		2/2		
LD Reg,Imm				2/3			

* = > Memory location addressed by B or X or directly.

Instructions Using A & C

Instructions	Bytes/Cycles
CLRA	1/1
INCA	1/1
DECA	1/1
LAID	1/3
DCORA	1/1
RRCA	1/1
SWAPA	1/1
SC	1/1
RC	1/1
IFC	1/1
IFNC	1/1

Transfer of Control Instructions

Instructions	Bytes/Cycles
JMPL	3/4
JMP	2/3
JP	1/3
JSRL	3/5
JSR	2/5
JID	1/3
RET	1/5
RETSK	1/5
RETI	1/5
INTR	1/7
NOP	1/1

BYTES and CYCLES per INSTRUCTION (Continued)

The following table shows the instructions assigned to unused opcodes. This table is for information only. The operations performed are subject to change without notice. Do not use these opcodes.

Unused Opcode	Instruction	Unused Opcode	Instruction
60	NOP	A9	NOP
61	NOP	AF	LD A, [B]
62	NOP	B1	C → HC
63	NOP	B4	NOP
67	NOP	B5	NOP
8C	RET	B7	X A, [X]
99	NOP	B9	NOP
9F	LD [B], #i	BF	LD A, [X]
A7	X A, [B]		
A8	NOP		

Programming Support

Programming of the single-chip emulator devices is supported by different sources. The following programmers are certified for programming the One-Time Programmable (OTP) and UV-erasable devices:

In addition to the application program, the ECON register needs to be programmed as well. The following tables provide examples of some ECON register values. For more detailed information refer to the ECON REGISTER section.

EPROM Security Disabled

RAM Memory	External CKI	RC Oscillator	Crystal Oscillator
64 Bytes	38	30	20
128 Bytes	3A	32	22

EPROM Security Enabled

RAM Memory	External CKI	RC Oscillator	Crystal Oscillator
64 Bytes	18	10	00
128 Bytes	1A	12	02

EPROM programmer manufacturers may not all calculate a "checksum" the same way. Before implementing an in-house verification by comparing checksums, need to ensure how each programming system utilized calculates a checksum. It is strongly recommended not to include the ECON register in the checksum for not all programmers include this byte in their calculated checksum.

ERASING THE COP8780C EPROM

The EPROM program memory is erased by exposing the transparent window on the UV erasable packages to an ultraviolet light source. Erasure begins to occur when exposed to light with wavelengths shorter than approximately 4000 Angstroms (Å). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000Å to 4000Å range.

After programming, "truly opaque" labels should be placed over the window of the device to prevent functional failure due to the generation of photo currents, erasure and excessive HALT current. The term "truly opaque" needs additional clarification when used in the context of covering quartz

EPROM Programmer Information

Manufacturer and Product	U.S. Phone Number	Europe Phone Number	Asia Phone Number
MetaLink- Debug Module	(602) 926-0797	Germany: +49-8141-1030	Hong Kong: 852-737-1800
Xeltek- Superpro	(408) 745-7974	Germany: +49-2041-684758	Singapore: +65-276-6433
BP Microsystems- EP-1140	(800) 225-2102	Germany: +49-89-857-6667	Hong Kong: +852-388-0629
Data I/O-Unisite; -System 29, -System 39	(800) 322-8246	Europe: +31-20-622866 Germany: +49-89-85-8020	Japan: +33-4326991
Abcom-COP8 Programmer		Europe: +89-808707	
System General Turpro-1-FX; -APRO	(408) 263-6667	Switzerland: +31-921-7844	Taiwan Taipei: +2-9173005

Programming Support (Continued)

windows on these devices. The typical white colored stickers or labels are normally used for they are easy to write on. These stickers are not opaque but translucent, they do let a certain percentage of UV-light through. The black write-protect labels supplied with 5.25" floppy disks work extremely well. If problems are encountered during programming (fails blank check) or during normal operation (intermittent functional or logical failures), need to determine first if an appropriate opaque label is being used to cover the quartz window at all times. Note that the device will also draw more current than normal (especially in HALT mode) when the window of the device is not covered with an opaque label.

The recommended erasure procedure for the device is exposure to short wave ultraviolet light which has a wavelength of 2537Å. The integrated dose (UV intensity × exposure time) for erasure should be a minimum of 30W-sec/cm^2.

The device should be placed within one inch of the lamp tubes during erasure. Some lamps have a filter on their tubes which should be removed before erasure. The following table shows the minimum erasure time for various light intensities.

Minimum Erasure Time

Light Intensity (Micro-Watts/cm^2)	Erasure Time* (Minutes)
15,000	36
10,000	50
8,500	60

*Does not include light intensity ramp up time.

An erasure system should be calibrated periodically. The distance from lamp to device should be maintained at one inch. The erasure time increases as the square of the distance. Lamps lose intensity as they age. When a lamp has aged, the system should be checked to make certain that adequate UV dosages are being applied for full erasure.

Common symptoms of insufficient erasure are:

- Inability to be programmed.
- Operational malfunctions associated with V_{CC}, temperature, or clock frequency.
- Loss of data in program memory.
- A change in configuration values in the ECON register.

Development Support

IN-CIRCUIT EMULATOR

The MetaLink iceMASTER™-COP8 Model 400 In-Circuit Emulator for the COP8 family of microcontrollers features high-performance operation, ease of use, and an extremely flexible user-interface for maximum productivity. Interchangeable probe cards, which connect to the standard common base, support the various configurations and packages of the COP8 family.

The iceMASTER provides real time, full speed emulation up to 10 MHz, 32 kbytes of emulation memory and 4k frames of trace buffer memory. The user may define as many as 32k trace and break triggers which can be enabled, disabled, set or cleared. They can be simple triggers based on code or address ranges, or complex triggers based on code address, direct address, opcode value, opcode class or immediate operand. Complex breakpoints can be ANDed and ORed together. Trace information consists of address bus values, opcodes, and user selectable probe clips status (external event lines). The trace buffer can be viewed as raw hex or as disassembled instructions. The probe clip bit values can be displayed in binary, hex or digital waveform formats.

During single-step operation the dynamically annotated code feature displays the contents of all accessed (read and write) memory locations and registers, as well as flow-of-control direction change markers next to each instruction executed.

The iceMASTER's performance analyzer offers a resolution of better than 6 μs. The user can easily monitor the time spent executing specific portions of code and find "hot spots" or "dead code". Up to 15 independent memory areas based on code address or label ranges can be defined. Analysis results can be viewed in bar graph format or as actual frequency count.

Emulator memory operations for program memory include single line assembler, disassembler, view, change and write to file. Data memory operations include fill, move, compare, dump to file, examine and modify. The contents of any memory space can be directly viewed and modified from the corresponding window.

The iceMASTER comes with an easy-to-use windowed interface. Each window can be sized, highlighted. color-controlled, added, or removed completely. Commands can be accessed via pull-down-menus and/or redefinable hot keys. A context sensitive hypertext/hyperlinked on-line help system explains clearly the options the user has from within any window.

The iceMASTER connects easily to a PC via the standard COMM port and its 115.2 kBaud serial link keeps typical program download to under 3 seconds.

The following tables list the emulator and probe cards ordering information.

Emulator Ordering Information

Part Number	Description	Current Version
IM-COP8/400/1‡	MetaLink base unit in-circuit emulator for all COP8 devices, symbolic debugger software and RS232 serial interface cable, with 110V @ 60 Hz Power Supply.	Host Software: Ver 3.3 Rev. 5, Model File Rev 3.050.
IM-COP8/400/2‡	MetaLink base unit in-circuit emulator for all COP8 devices, symbolic debugger software and RS232 serial interface cable, with 220V @ 50Hz Power Supply.	
DM-COP8/880C‡	MetaLink iceMASTER Debug Module. This is the low cost version of the MetaLink's iceMASTER. Firmware: Ver. 6.07	

‡These parts include National's COP8 Assembler/Linker/Librarian Package (COP8-DEV-IBMA).

Development Support (Continued)

Probe Card Ordering Information

Part Number	Package	Voltage Range	Emulator
MHW-880C28D5PC	28 DIP	4.5V–5.5V	COP820C, 840C, 881C, 8781C
MHW-880C28DWPC	28 DIP	2.5V–6.0V	COP820C, 840C, 881C, 8781C
MHW-880C40D5PC	40 DIP	4.5V–5.5V	COP880C, 8780C
MHW-880C40DWPC	40 DIP	2.5V–6.0V	COP880C, 8780C
MHW-880C44D5PC	44 PLCC	4.5V–5.5V	COP880C, 8780C
MHW-880C44DWPC	44 PLCC	2.5V–6.0V	COP880C, 8780C

MACRO CROSS ASSEMBLER

National Semiconductor offers a COP8 macro cross assembler. It runs on industry standard compatible PCs and supports all of the full symbolic debugging features of the MetaLink iceMASTER emulators.

Assembler Ordering Information

Part Number	Description	Manual
COP8-DEV-IBMA	COP8 Assembler/ Linker/Librarian for IBM® PC/XT®, AT® or compatible.	424410632-001

CROSS REFERENCE TABLE

The following cross reference table lists the COP800 devices which can be emulated with the COP87XXC single-chip, form fit and function emulators.

Single-Chip Emulator Selection Table

Device Number	Package	Description	Emulates
COP8780CV	44 PLCC	One Time Programmable (OTP)	COP880C
COP8780CEL	44 LDCC	UV Erasable	COP880C
COP8780CN	40 DIP	OTP	COP880C
COP8780CJ	40 DIP	UV Erasable	COP880C
COP8781CN	28 DIP	OTP	COP881C, COP840C, COP820C
COP8781CJ	28 DIP	UV Erasable	COP881C, COP840C, COP820C
COP8781CWM	28 SO	OTP	COP881C, COP840C, COP820C
COP8782CN	20 DIP	OTP	COP842C, COP822C
COP8782CJ	20 DIP	UV Erasable	COP842C, COP822C
COP8782CWM	20 SO	OTP	COP842C, COP822C

DIAL-A-HELPER

Dial-A-Helper is a service provided by the Microcontroller Applications Group. The Dial-A-Helper is an Electronic Bulletin Board information system.

INFORMATION SYSTEM

The Dial-A-Helper system provides access to an automated information storage and retrieval system that may be accessed over standard dial-up telephone lines 24 hours a day. The system capabilities include a MESSAGE SECTION (electronic mail) for communications to and from the Microcontroller Applications Group and a FILE SECTION which consists of several file areas where valuable application software and utilities could be found. The minimum requirement for accessing the Dial-A-Helper is a Hayes compatible modem.

If the user has a PC with a communications package then files from the FILE SECTION can be down-loaded to disk for later use.

FACTORY APPLICATIONS SUPPORT

Dial-A-Helper also provides immediate factory applications support. If a user has questions, he can leave messages on our electronic bulletin board.

Voice:	(800) 272-9959	
Modem:	CANADA/U.S.:	(800) NSC-MICRO (800) 672-6427
	Baud:	14.4k
	Setup:	Length: 8-Bit Parity: None Stop Bit: 1
	Operation:	24 Hrs. 7 Days