National Semiconductor

COP402/COP402M ROMless N-Channel Microcontrollers

General Description

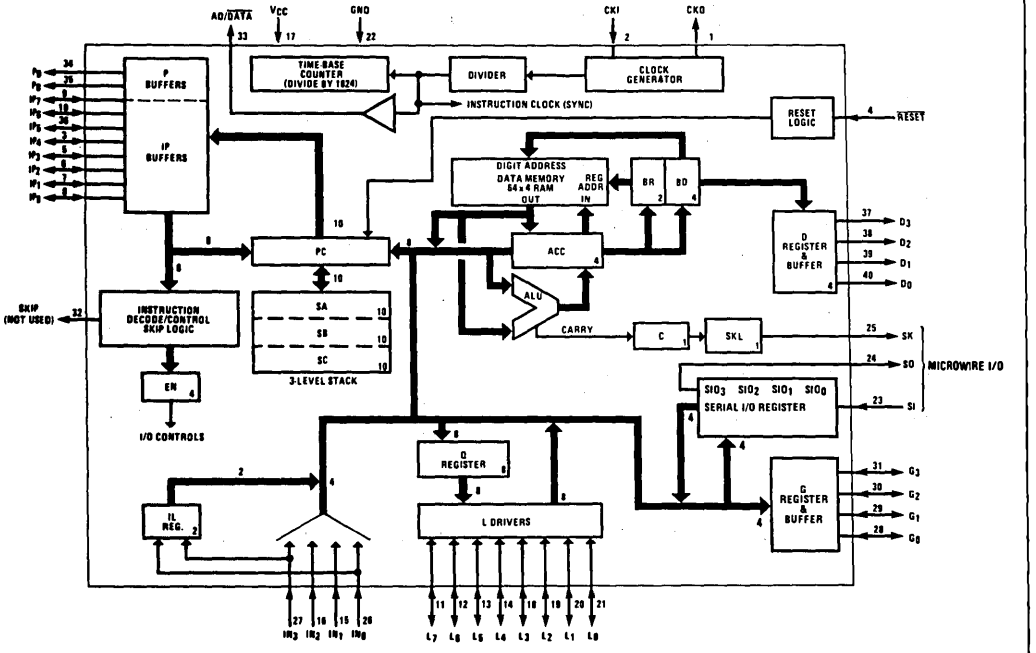
The COP402/COP402M ROMless Microcontrollers are members of the COPS™ family, fabricated using N-channel silicon gate MOS technology. Each part contains CPU, RAM, and I/O, and is identical to a COP420 device, except the ROM has been removed; pins have been added to output the ROM address and to input ROM data. In a system, the COP402 or 402M will perform exactly as the COP420; this important benefit facilitates development and debug of a COP420; this important benefit facilitates development and debug of a COP420 program prior to masking the final part. These devices are also appropriate in low volume applications, or when the program may require changing. The COP402M is identical to the COP402, except the MICROBUS™ interface option has been implemented.

The COP402 may also be used to emulate the COP410L, 411L, or 420L by appropriately reducing the clock frequency.

Features

- Extended temperature (−40°C to +85°C) COP302/COP302M, available as special order
- Low cost
- Exact circuit equivalent of COP420
- Standard 40-pin dual-in-line package
- Interfaces with standard PROM or ROM
- 64 x 4 RAM, addresses up to 1k x 8 ROM
- MICROBUS compatible (COP402M)
- Powerful instruction set
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 4.0 μs instruction time
- Single supply operation (4.5V to 6.3V)
- Internal time-base counter for real-time processing
- Internal binary counter register with MICROWIRE™ serial I/O capability
- Software/hardware compatible with other members of COP400 family

Block Diagram



TL/DD/6915-1

FIGURE 1

COP402/COP402M and COP302/COP302M

Absolute Maximum Ratings

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

Voltage at Any Pin	−0.3V to +7V
Operating Temperature Range	
COP402/COP402M	0°C to 70°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature (soldering, 10 sec.)	300°C
Package Power Dissipation	750 mW at 25°C 400 mW at 70°C 250 mW at 85°C
Total Sink Current	50 mA
Total Source Current	70 mA

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.*

COP402/COP402M
DC Electrical Characteristics $0°C \le T_A \le 70°C$, $4.5V \le V_{CC} \le 6.3V$ unless otherwise noted

Parameter	Conditions	Min	Max	Units
Operation Voltage		4.5	6.3	V
Power Supply Ripple	Peak to Peak (Note 3)		0.4	V
Supply Current	All Outputs Open V_{CC} = 5V		40	mA
Input Voltage Levels				
CKI Input Levels				
Crystal Input				
Logic High		2.4		V
Logic Low		−0.3	0.4	V
Schmitt Trigger Input				
RESET				
Logic High		0.7 V_{CC}		V
Logic Low		−0.3	0.6	V
All Other Inputs				
Logic High	V_{CC} = Max	3.0		V
Logic High	V_{CC} = 5V ±5%	2.0		V
Logic Low		−0.3	0.8	V
Input Load Source Current	V_{CC} = 5V, V_{IN} = 0V	−100	−800	μA
Input Capacitance			7	pF
Hi-Z Input Leakage	V_{CC} = 5V	−1	+1	μA
Output Voltage Levels				
D, G, L, SK, SO Outputs				
TTL Operation	V_{CC} = 5V ±10%			
Logic High	I_{OH} = −100 μA	2.4		V
Logic Low	I_{OL} = 1.6 mA	−0.3	0.4	V
IP0–IP7, P8, P9, SKIP, CKO, AD/$\overline{DATA}$				
Logic High	I_{OH} = −75 μA	2.4		V
Logic Low	I_{OL} = 400 μA	−0.3	0.4	V
CMOS Operation (Note 1)				
Logic High	I_{OH} = −10 μA	V_{CC} − 1		V
Logic Low	I_{OL} = 10 μA	−0.3	0.2	V
Output Current Levels				
LED Direct Drive (COP402)	V_{CC} = 6V			
Logic High	V_{OH} = 2.0V	2.5	14	mA
TRI-STATE® (COP402M) Leakage Current	V_{CC} = 5V	−50	+50	μA
Allowable Sink Current				
Per Pin (L, D, G)			10	mA
Per Pin (All Others)			2	mA
Per Port (L)			16	mA
Per Port (D, G)			10	mA
Allowable Source Current				
Per Pin (L)			−15	mA
Per Pin (All Others)			−1.5	mA

Note 1: TRI-STATE and LED configurations are excluded.

COP402/COP402M
AC Electrical Characteristics $0°C \le T_A \le 70°C, 4.5V \le V_{CC} \le 6.3V$ unless otherwise noted

Parameter	Conditions	Min	Max	Units
Instruction Cycle Time		4	10	μs
Operating CKI Frequency	÷16 Mode	1.6	4.0	MHz
CKI Duty Cycle (Note 1)		40	60	%
Rise Time	Frequency = 4 MHz		60	ns
Fall Time	Frequency = 4 MHz		40	ns
Inputs:				
SI				
t_{SETUP}		0.3		μs
t_{HOLD}		250		ns
All Other Inputs				
t_{SETUP}		1.7		μs
t_{HOLD}		300		ns
Output Propagation Delay	Test Conditions: $R_L = 5k, C_L = 50\,pF, V_{OUT} = 1.5V$			
SO and SK				
t_{pd1}			1.0	μs
t_{pd0}			1.0	μs
CKO				
t_{pd1}			0.25	μs
t_{pd0}			0.25	μs
AD/$\overline{DATA}$, SKIP				
t_{pd1}			0.6	μs
t_{pd0}			0.6	μs
All Other Outputs				
t_{pd1}			1.4	μs
t_{pd0}			1.4	μs
MICROBUS Timing	$C_L = 100\,pF, V_{CC} = 5V \pm 5\%$			
Read Operation *(Figure 4)*				
Chip Select Stable before $\overline{RD}$—t_{CSR}		65		ns
Chip Select Hold Time for $\overline{RD}$—t_{RCS}		20		ns
$\overline{RD}$ Pulse Width—t_{RR}		400		ns
Data Delay from $\overline{RD}$—t_{RD}			375	ns
$\overline{RD}$ to Data Floating—t_{DF}			250	ns
Write Operation *(Figure 5)*				
Chip Select Stable before $\overline{WR}$—t_{CSW}		65		ns
Chip Select Hold Time for $\overline{WR}$—t_{WCS}		20		ns
$\overline{WR}$ Pulse Width—t_{WW}		400		ns
Data Set-Up Time for $\overline{WR}$—t_{DW}		320		ns
Data Hold Time for $\overline{WR}$—t_{WD}		100		ns
INTR Transition Time from $\overline{WR}$—t_{WI}			700	ns

Note 1: Duty Cycle = $t_{W1}/(t_{W1} + t_{W0})$.

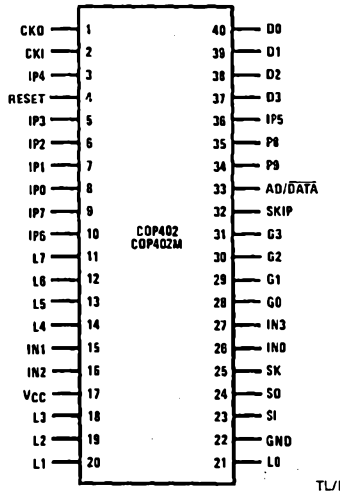
Note 2: See *Figure 9* for additional I/O characteristics.

Note 3: Voltage change must be less than 0.5V in a 1 ms period.

Note 4: Exercise great care not to exceed maximum device power dissipation limits when direct driving LEDs (or sourcing similar loads) at high temperature.

Connection Diagram

Dual-In-Line Package



TL/DD/6915-2

Top View

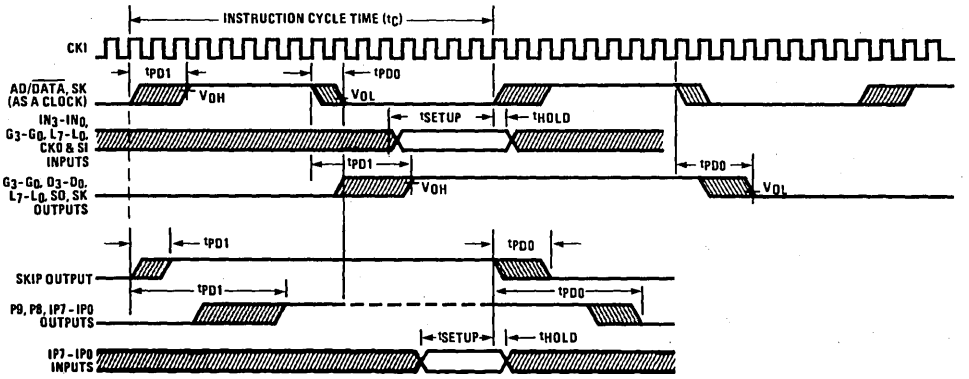
Order Number COP402N or COP402MN
See NS Package Number N40A

FIGURE 2.

Pin Descriptions

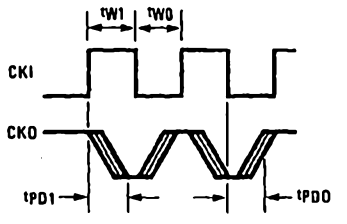
Pin	Description
L_7–L_0	8 bidirectional I/O ports with TRI-STATE
G_3–G_0	4 bidirectional I/O ports
D_3–D_0	4 general purpose outputs
IN_3–IN_0	4 general purpose inputs
SI	Serial input (or counter input)
SO	Serial output (or general purpose output)
SK	Logic-controlled clock (or general purpose output)
AD/$\overline{DATA}$	Address out/data in flag
SKIP	Instruction skip output
CKI	System oscillator input
CKO	System oscillator output
$\overline{RESET}$	System reset input
V_{CC}	Power supply
GND	Ground
IP7–IP0	8 bidirectional ROM address and data ports
P8, P9	2 most significant ROM address outputs

Timing Diagrams



TL/DD/6915-3

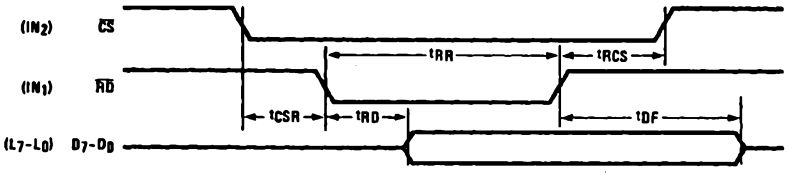
FIGURE 3a. Input/Output Timing Diagrams (Crystal ÷ 16 Mode)



TL/DD/6915-4

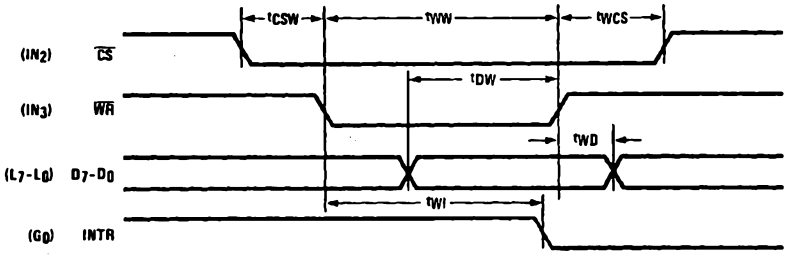
FIGURE 3b. CKO Output Timing

Timing Diagrams (Continued)



TL/DD/6915-5

FIGURE 4. MICROBUS Read Operation Timing



TL/DD/6915-6

FIGURE 5. MICROBUS Write Operation Timing

Functional Description

A block diagram of the COP402 is given in *Figure 1.* 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. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2V). When a bit is reset, it is a logic "0" (less than 0.8V).

PROGRAM MEMORY

Program Memory consists of a 1,024-byte external memory (typically PROM). Words of this memory may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit PC register. Its binay value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential **10-bit binary count** value. Three levels of subroutine nesting are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

DATA MEMORY

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instruction based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

INTERNAL LOGIC

The 4-bit **A register** (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A **4-bit adder** performs the arithmetic and logic functions of the COP402/402M, storing its results in A. It also outputs a carry bit to the 1-bit **C register**, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four **general-purpose inputs, IN_3–IN_0,** are provided; IN_1, IN_2, and IN_3 may be selected, by a mask-programmable option, as Read Strobe, Chip Select and Write Strobe inputs, respectively, for use in MICROBUS applications.

The **D register** provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd.

The **G register** contents are outputs to 4 general-purpose bidirectional I/O ports. G_0 may be mask-programmed as a "ready" output for MICROBUS applications.

The **Q register** is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.) With the MICROBUS option selected, Q can also be loaded with the 8-bit contents of the L I/O ports upon the occurrence of a write strobe from the host CPU.

Functional Description (Continued)

The **8 L drivers**, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the MICROBUS option allows L I/O port data to be latched into the Q register. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa–Sg and decimal point segments of the display.

The **SIO register** functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The **XAS instruction** copies C into the SKL latch. In the counter mode, SK is the output of SKL. In the shift register mode, SK outputs SKL ANDed with internal instruction cycle clock.

The **EN register** is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register (EN_3–EN_0).

1. The least significant bit of the enable register, EN_0, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With EN_0 set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of EN_3. With EN_0 reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With EN_1 set the IN_1 input is enabled as an interrupt input. Immediately following an interrupt, EN_1 is reset to disable further interrupts.
3. With EN_2 set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting EN_2 disables the L drivers, placing the L I/O ports in a high-impedance input state. If the MICROBUS option is being used, EN_2 does not affect the L drivers.
4. EN_3, in conjunction with EN_0, affects the SO output. With EN_0 set (binary counter option selected) SO will output the value loaded into EN_3. With EN_0 reset (serial shift register option selected), setting EN_3 enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting EN_3 with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with EN_3 and EN_0.

INTERRUPT

The following features are associated with the IN_1 interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address 0FF (the last word of page 3) and EN_1 is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1. EN_1 has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the IN_1 input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon the popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. At *this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and the LQID instruction should not be nested within the interrupt servicing routine since their popping of the stack enables any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. An LEI instruction can be put immediately before the RET to re-enable interrupts.

TABLE I. Enable Register Modes—Bits EN_3 and EN_0

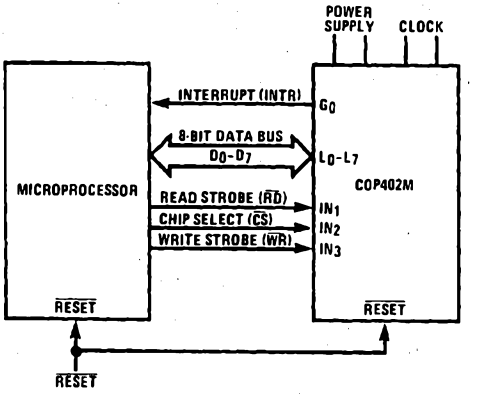
EN_3	EN_0	SIO	SI	SO	SK
0	0	Shift Register	Input to Shift Register	0	If SKL = 1, SK = SYNC If SKL = 0, SK = 0
1	0	Shift Register	Input to Shift Register	Serial Out	If SKL = 1, SK = SYNC If SKL = 0, SK = 0
0	1	Binary Counter	Input to Binary Counter	0	If SKL = 1, SK = 1 If SKL = 0, SK = 0
1	1	Binary Counter	Input to Binary Counter	1	If SKL = 1, SK = 1 If SKL = 0, SK = 0

Functional Description (Continued)

MICROBUS INTERFACE

The COP402M can be used as a peripheral microprocessor device, inputting and outputting data from and to a host microprocessor (μP). IN_1, IN_2, and IN_3 general purpose inputs become MICROBUS compatible read-strobe, chip-select, and write-strobe lines, respectively. IN_1 becomes $\overline{RD}$—a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the μP. IN_2 becomes $\overline{CS}$—a logic "0" on this line selects the COP402M as the μP peripheral device by enabling the operation of the $\overline{RD}$ and $\overline{WR}$ lines and allows for the selection of one of several peripheral components. IN_3 becomes $\overline{WR}$—a logic "0" on this line will write bus data from the L ports to the Q latches for input to the COP402M. G_0 becomes INTR, a "ready" output reset by a write pulse from the μP on the $\overline{WR}$ line, providing the "handshaking" capability necessary for asynchronous data transfer between the host CPU and the COP402M.

This option has been designed for compatibility with National's MICROBUS—a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See MICROBUS, National Publication.) The functioning and timing relationships between the COP402M signal lines affected by this option are as specified for the MICROBUS interface, and are given in the AC electrical characteristics and shown in the timing diagrams (*Figures 4* and *5*). Connection to the MICROBUS is shown in *Figure 6*.



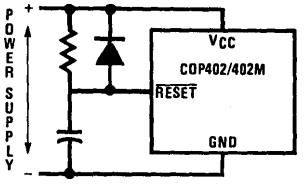
TL/DD/6915-7

FIGURE 6. MICROBUS Option Interconnect

INITIALIZATION

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1 ms and greater than 1 μs. If the power supply rise time is greater than 1 ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to V_{CC}. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least two instruction cycle times.

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, G, and SO are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) is not cleared upon initialization.* The first instruction at address 0 must be a CLRA.



RC ≥ 5 × Power Supply Rise Time TL/DD/6915-8

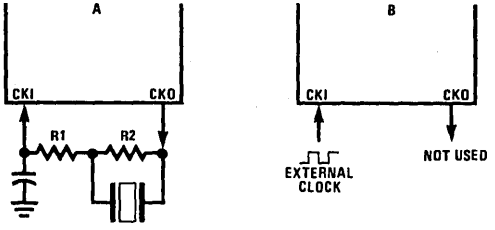
FIGURE 7. Power-Up Clear Circuit

OSCILLATOR

There are two basic clock oscillator configurations available as shown by *Figure 8*.

a. **Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 16.

b. **External Oscillator.** CKI is driven by an external clock signal. The instruction cycle time is the clock frequency divided by 16.



TL/DD/6915-9

Crystal Value	Component Values		
	R1	R2	C
4 MHz	1k	1M	27 pF
3.58 MHz	1k	1M	27 pF
2.09 MHz	1k	1M	56 pF

FIGURE 8. COP402/402M Oscillator

EXTERNAL MEMORY INTERFACE

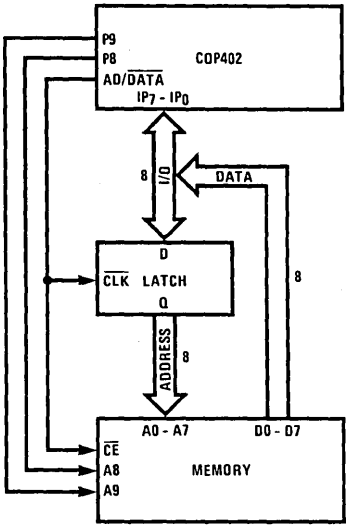
The COP402 and COP402M are designed for use with an external Program Memory. This memory may be implemented using any devices having the following characteristics:

1. random addressing
2. TTL-compatible TRI-STATE outputs
3. TTL = compatible inputs
4. access time = 1.0 μs, max.

Typically these requirements are met using bipolar or MOS PROMs.

Functional Description (Continued)

During operation, the address of the next instruction is sent out on P9, P8, and IP7 through IP0 during the time that AD/$\overline{\text{DATA}}$ is high (logic "1" = address mode). Address data on the IP lines is stored into an external latch on the high-to-low transition of the AD/$\overline{\text{DATA}}$ line; P9 and P8 are dedicated address outputs, and do not need to be latched. When AD/$\overline{\text{DATA}}$ is low (logic "0" = data mode), the output of the memory is gated onto IP7 through IP0, forming the input bus. Note that the AD/$\overline{\text{DATA}}$ output has a period of one instruction time, a duty cycle of approximately 50%, and specifies whether the IP lines are used for address output or instruction input. A simplified block diagram of the external memory interface is shown in *Figure 9*.



TL/DD/6915-10

FIGURE 9. External Memory Interface to COP402

INPUT/OUTPUT

COP402 outputs have the following configurations, illustrated in *Figure 10*.

a. **Standard**—an enhancement-mode device to ground in conjunction with a depletion-mode device to V_{CC}, compatible with TTL and CMOS input requirements.

b. **High Drive**—same as **a.** except greater current sourcing capability.

c. **Push-Pull**—an enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to V_{CC}. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads.

d. **LED Direct Drive**—an enhancement-mode device to ground and to V_{CC}, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (see Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display.

e. **TRI-STATE Push-Pull**—an enhancement-mode device to ground and V_{CC} intended to meet the requirements associated with the MICROBUS option. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers.

f. Inputs have an on-chip depletion load device to V_{CC}, as shown in *Figure 10f*.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1–6, respectively). Minimum and maximum current (I_{OUT} and V_{OUT}) curves are given in *Figure 10* for each of these devices.

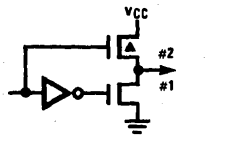
The SO, SK outputs are configured as shown in *Figure 10c*. The D and G outputs are configured as shown in *Figure 10a*.

Functional Description (Continued)

Note that when inputting data to the G ports, the G outputs should be set to "1". The L outputs are configured as in *Figure 10d* on the COP402. On the COP402M the L outputs are as in *Figure 10e*.

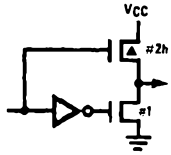
An important point to remember if using configuration d with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current. (See *Figure 11*.)

IP7 through IP0 outputs are configured as shown in *Figure 10c*; P9, P8, SKIP, and AD/$\overline{\text{DATA}}$ are configured as shown in *Figure 10b*.



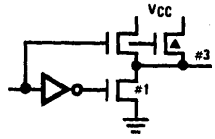
TL/DD/6915–11

a. Standard



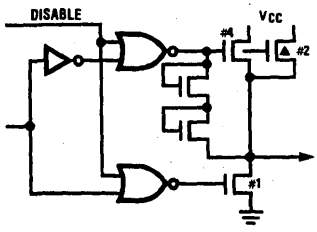
TL/DD/6915–12

b. High Drive



TL/DD/6915–13

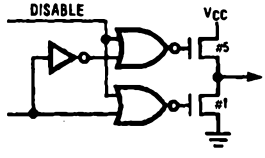
c. Push-Pull



TL/DD/6915–14

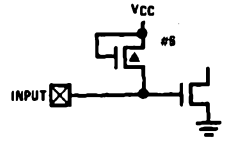
d. LED

(▲ is Depletion Device)



TL/DD/6915–15

e. TRI-STATE Push-Pull

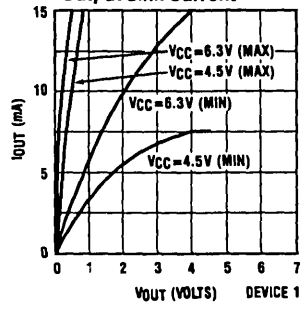


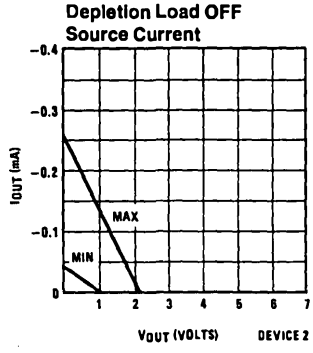
TL/DD/6915–16

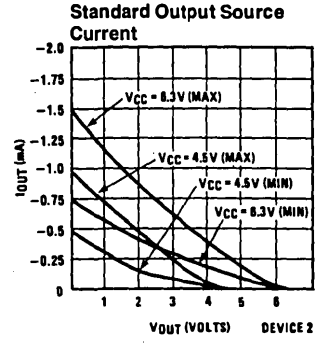
f. Input with Load

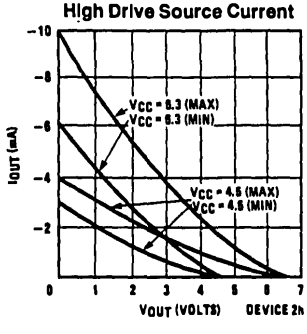
FIGURE 10. Input/Output Configurations

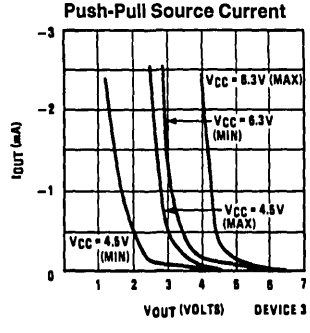
Typical Performance Characteristics

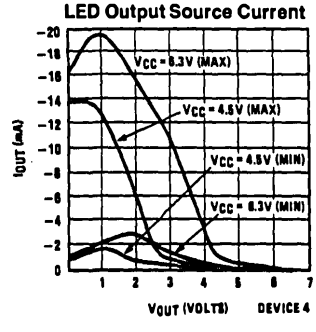


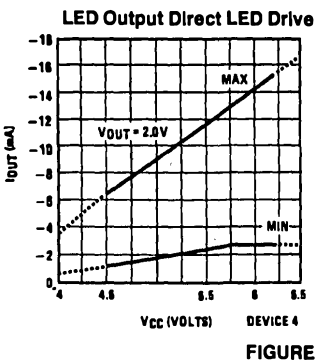


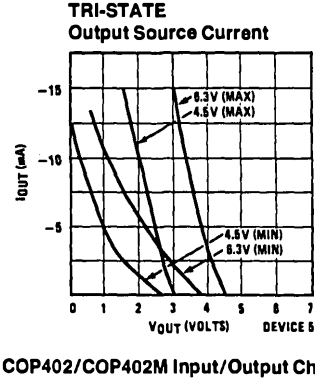












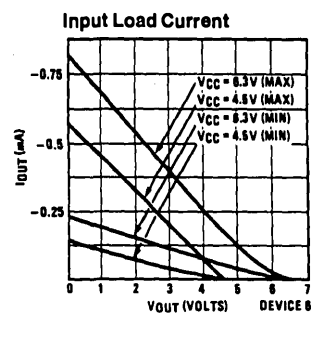
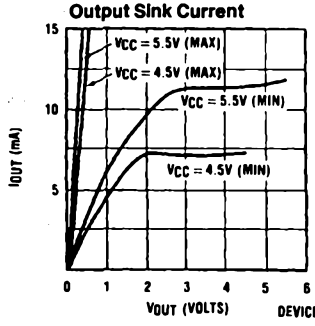


FIGURE 11. COP402/COP402M Input/Output Characteristics

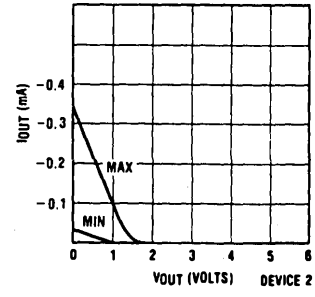
TL/DD/6915–17

Typical Performance Characteristics (Continued)

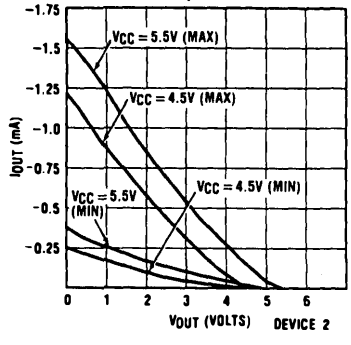
Output Sink Current



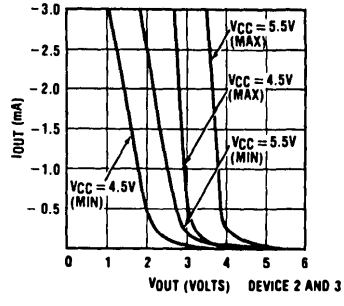
L Output Depletion Load Off Source Current



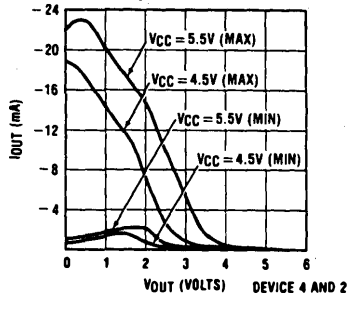
Standard Output Source Current



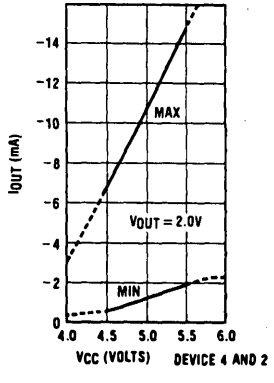
Push Pull Source Current



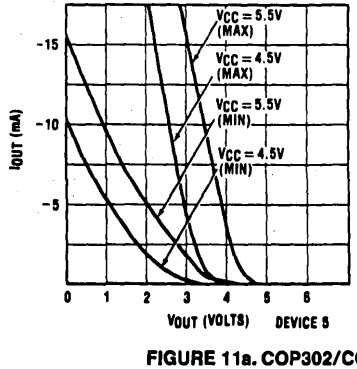
LED Output Source Current



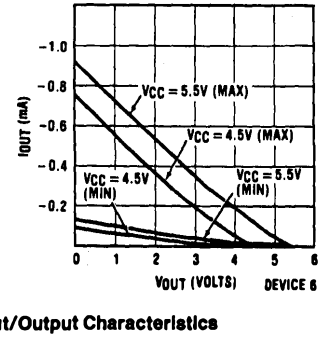
LED Output Device LED Drive



TRI-STATE Output Source Current



Input Load Source Current



TL/DD/6915-18

FIGURE 11a. COP302/COP302M Input/Output Characteristics

Instruction Set

Table II is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table III provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP402/402M instruction set.

TABLE II. COP402/COP402M Instruction Set Table Symbols

Symbol	Definition
INTERNAL ARCHITECTURE SYMBOLS	
A	4-bit Accumulator
B	6-bit RAM Address Register
Br	Upper 2 bits of B (register address)
Bd	Lower 4 bits of B (digit address)
C	1-bit Carry Register
D	4-bit Data Output Port
EN	4-bit Enable Register
G	4-bit Register to latch data for G I/O Port
IL	Two 1-bit Latches Associated with the IN_3 or IN_0 inputs
IN	4-bit Input port
L	8-bit TRI-STATE I/O Port
M	4-bit contents of RAM Memory pointed to by B Register
P	2-bit ROM Address Port
PC	10-bit ROM Address Register (program counter)
Q	8-bit Register to latch data for L I/O Port
SA	10-bit Subroutine Save Register A
SB	10-bit Subroutine Save Register B
SC	10-bit Subroutine Save Register C
SIO	4-bit Shift Register and Counter
SK	Logic-Controlled Clock Output

Symbol	Definition
INSTRUCTION OPERAND SYMBOLS	
d	4-bit Operand Field, 0–15 binary (RAM Digit Select)
r	2-bit Operand Field, 0–3 binary (RAM Register Select)
a	9-bit Operand Field, 0–511 binary (ROM Address)
y	4-bit Operand Field, 0–15 binary (Immediate Data)
RAM(s)	Contents of RAM location addressed by s
ROM(t)	Contents of ROM location addressed by t
OPERATIONAL SYMBOLS	
+	Plus
−	Minus
→	Replaces
↔	Is exchanged with
=	Is equal to
$\overline{A}$	The one's complement of A
⊕	Exclusive-OR
:	Range of values

TABLE III. COP402/COP402M Instruction Set

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
ARITHMETIC INSTRUCTIONS						
ASC		30	|0011|0000|	A + C + RAM(B) → A Carry → C	Carry	Add with Carry, Skip on Carry
ADD		31	|0011|0001|	A + RAM(B) → A	None	Add RAM to A
ADT		4A	|0100|1010|	A + 10_{10} → A	None	Add Ten to A
AISC	y	5–	|0101| y |	A + y → A	Carry	Add Immediate, Skip on Carry (y ≠ 0)
CASC		10	|0001|0000|	$\overline{A}$ + RAM(B) + C → A Carry → C	Carry	Complement and Add with Carry, Skip on Carry
CLRA		00	|0000|0000|	0 → A	None	Clear A
COMP		40	|0100|0000|	$\overline{A}$ → A	None	One's complement of A to A
NOP		44	|0100|0100|	None	None	No Operation
RC		32	|0011|0010|	"0" → C	None	Reset C
SC		22	|0010|0010|	"1" → C	None	Set C
XOR		02	|0000|0010|	A ⊕ RAM(B) → A	None	Exclusive-OR RAM with A

Instruction Set (Continued)

TABLE III. COP402/COP402M Instruction Set (Continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
TRANSFER OF CONTROL INSTRUCTIONS						
JID		FF	1111 | 1111	ROM ($PC_{9:8}$, A,M) → $PC_{7:0}$	None	Jump Indirect (Note 3)
JMP	a	6–	0110 | 00 | $a_{9:8}$	a → PC	None	Jump
		--	$a_{7:0}$			
JP	a	--	1 | $a_{6:0}$ (pages 2,3 only)	a → $PC_{6:0}$	None	Jump within Page (Note 4)
			or			
		--	11 | $a_{5:0}$ (all other pages)	a → $PC_{5:0}$		
JSRP	a	--	10 | $a_{5:0}$	PC + 1 → SA → SB → SC 0010 → $PC_{9:6}$ a → $PC_{5:0}$	None	Jump to Subroutine Page (Note 5)
JSR	a	6–	0110 | 10 | $a_{9:8}$	PC + 1 → SA → SB → SC a → PC	None	Jump to Subroutine
		--	$a_{7:0}$			
RET		48	0100 | 1000	SC → SB → SA → PC	None	Return from Subroutine
RETSK		49	0100 | 1001	SC → SB → SA → PC	Always Skip on Return	Return from Subroutine then Skip
MEMORY REFERENCE INSTRUCTIONS						
CAMQ		33	0011 | 0011	A → $Q_{7:4}$ RAM(B) → $Q_{3:0}$	None	Copy A, RAM to Q
		3C	0011 | 1100			
CQMA		33	0011 | 0011	$Q_{7:4}$ → RAM(B) $Q_{3:0}$ → A	None	Copy Q to RAM, A
		2C	0010 | 1100			
LD	r	-5	00 | r | 0101	RAM(B) → A Br ⊕ r → Br	None	Load RAM into A, Exclusive-OR Br with r
LDD	r,d	23	0010 | 0011	RAM(r,d) → A	None	Load A with RAM pointed to directly by r,d
		--	00 | r | d			
LQID		BF	1011 | 1111	ROM($PC_{9:8}$,A,M) → Q SB → SC	None	Load Q Indirect (Note 3)
RMB	0	4C	0100 | 1100	0 → $RAM(B)_0$	None	Reset RAM Bit
	1	45	0100 | 0101	0 → $RAM(B)_1$		
	2	42	0100 | 0010	0 → $RAM(B)_2$		
	3	43	0100 | 0011	0 → $RAM(B)_3$		
SMB	0	4D	0100 | 1101	1 → $RAM(B)_0$	None	Set RAM Bit
	1	47	0100 | 0111	1 → $RAM(B)_1$		
	2	46	0100 | 0110	1 → $RAM(B)_2$		
	3	4B	0100 | 1011	1 → $RAM(B)_3$		
STII	y	7–	0111 | y	y → RAM(B) Bd + 1 → Bd	None	Store Memory Immediate and Increment Bd
X	r	-6	00 | r | 0110	RAM(B) ⟷ A Br ⊕ r → Br	None	Exchange RAM with A, Exclusive-OR Br with r
XAD	r,d	23	0010 | 0011	RAM(r,d) ⟷ A	None	Exchange A with RAM pointed to directly by r,d
		--	10 | r | d			

Instruction Set (Continued)

TABLE III. COP402/COP402M Instruction Set (Continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
MEMORY REFERENCE INSTRUCTIONS (Continued)						
XDS	r	-7	|00|r|0111|	RAM(B) ⟷ A Bd − 1 → Bd Br ⊕ r → Br	Bd decrements past 0	Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r
XIS	r	-4	|00|r|0100|	RAM(B) ⟷ A Bd + 1 → Bd Br ⊕ r → Br	Bd increments past 15	Exchange RAM with A and Increment Bd, Exclusive-OR Br with r
REGISTER REFERENCE INSTRUCTIONS						
CAB		50	|0101|0000|	A → Bd	None	Copy A to Bd
CBA		4E	|0100|1110|	Bd → A	None	Copy Bd to A
LBI	r,d	--	|00|r|(d − 1)| (d=0, 9:15) or	r,d → B	Skip until not a LBI	Load B Immediate with r,d (Note 6)
		33 --	|0011|0011| |10| r | d | (any d)			
LEI	y	33 6-	|0011|0011| |0110| y |	y → EN	None	Load EN Immediate (Note 7)
XABR		12	|0001|0010|	A ⟷ Br (0,0 → A_3,A_2)	None	Exchange A with Br
TEST INSTRUCTIONS						
SKC		20	|0010|0000|		C = "1"	Skip if C is True
SKE		21	|0010|0001|		A = RAM(B)	Skip if A Equals RAM
SKGZ		33 21	|0011|0011| |0010|0001|		$G_{3:0}$ = 0	Skip if G is Zero (all 4 bits)
SKGBZ		33	|0011|0011|	1st byte		Skip if G Bit is Zero
	0	01	|0000|0001|	2nd byte	G_0 = 0	
	1	11	|0001|0001|	2nd byte	G_1 = 0	
	2	03	|0000|0011|	2nd byte	G_2 = 0	
	3	13	|0001|0011|	2nd byte	G_3 = 0	
SKMBZ	0	01	|0000|0001|		$RAM(B)_0$ = 0	Skip if RAM Bit is Zero
	1	11	|0001|0001|		$RAM(B)_1$ = 0	
	2	03	|0000|0011|		$RAM(B)_2$ = 0	
	3	13	|0001|0011|		$RAM(B)_3$ = 0	
SKT		41	|0100|0001|		A time-base counter carry has occurred since last test	Skip on Timer (Note 3)

Instruction Set (Continued)

TABLE III. COP402/COP402M Instruction Set (Continued)

Mnemonic	Operand	Hex Code	Machine Language Code (Binary)	Data Flow	Skip Conditions	Description
INPUT/OUTPUT INSTRUCTIONS						
ING		33 2A	0011 0011 0010 1010	G → A	None	Input G Ports to A
ININ		33 28	0011 0011 0010 1000	IN → A	None	Input IN Inputs to A (Notes 2 and 8)
INIL		33 29	0011 0011 0010 1001	IL_3, "0", IL_0 → A	None	Input IL Latches to A (Note 3)
INL		33 2E	0011 0011 0010 1110	$L_{7:4}$ → RAM(B) $L_{3:0}$ → A	None	Input L Ports to RAM,A
OBD		33 3E	0011 0011 0011 1110	Bd → D	None	Output Bd to D Outputs
OGI	y	33 5–	0011 0011 0101 y	y → G	None	Output to G Ports Immediate
OMG		33 3A	0011 0011 0011 1010	RAM(B) → G	None	Output RAM to G Ports
XAS		4F	0100 1111	A ⟷ SIO, C → SKL	None	Exchange A with SIO (Note 3)

Note 1: All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, A_3 indicates the most significant (left-most) bit of the 4-bit register.

Note 2: The ININ instruction is not available on the 24-pin COP421 since this device does not contain the IN inputs.

Note 3: For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

Note 4: The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

Note 5: A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

Note 6: LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 (1001_2), the lower 4 bits of the LBI instruction equal 8 (1000_2). To load 0, the lower 4 bits of the LBI instruction should equal 15 (1111_2).

Note 7: Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

Note 8: The COP402M will always read a "1" into A1 with the ININ instruction.

Description of Selected Instructions

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing programs.

XAS INSTRUCTION

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register.

The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once evey 4 instruction cycles to effect a continuous data stream.

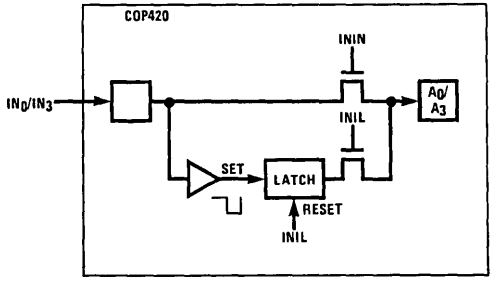
JID INSTRUCTION

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M. PC_9 and PC_8 are not affected by this instruction.

Note that JID requires 2 instruction cycles.

INIL INSTRUCTION

INIL (Input IL Latches to A) inputs 2 latches, IL_3 and IL_0 (see *Figure 12*) and CKO into A. The IL_3 and IL_0 latches are set if a low-going pulse ("1" to "0") has occurred on the IN_3 and IN_0 inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs IL_3 and IN_0 into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the IN_3 and IN_0 lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs IN_3–IN_0 are input to A upon the execution of an ININ instruction. (See Table III, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.



TL/DD/6915–19

FIGURE 12. IN_0/IN_3 Latches

LQID INSTRUCTION

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word PC_9, PC_8, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving PC_9 and PC_8 unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

SKT INSTRUCTION

The SKT (Skip on Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the controller to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 131 kHz (crystal frequency ÷ 16) and the binary counter output pulse frequency will be 128 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 128 ticks.

INSTRUCTION SET NOTES

a. The first word of a program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed, except JID and LQID. LQID and JID take two cycle times if executed and one if skipped.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is a 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, or 15 will access data in the next group of 4 pages.

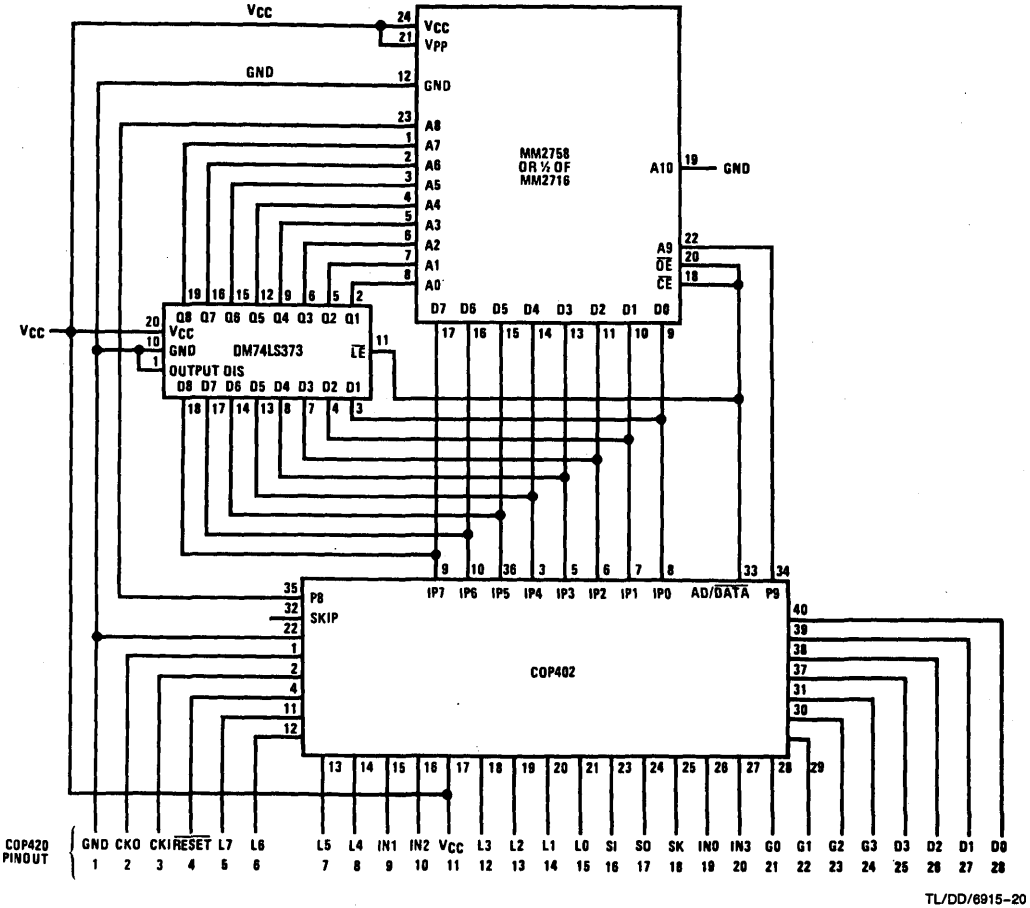
Typical Application: PROM-Based System

The COP402 may be used to exactly emulate the COP420, *Figure 13* shows the interconnect to implement a COP420 hardware emulation. This connection uses two MM5204 EPROMs as external memory. Other memory can be used such as bipolar PROM or RAM.

Pins IP7–IP0 are bidirectional inputs and outputs. When the AD/$\overline{\text{DATA}}$ clocking output turns on, the EPROM drivers are disabled and IP7–IP0 output addresses. The 8-bit latch (MM74C373) latches the addresses to drive the memory.

When AD/$\overline{\text{DATA}}$ turns off, the EPROMs are enabled and the IP7-IP0 pins will input the memory data. P8 and P9 output the most significant address bits to the memory. (SKIP output may be used for program debug if needed.)

The other 28 pins of the COP402 may be configured exactly the same as a COP420. The COP402M chip can be used if the MICROBUS feature of the COP420 is needed.



TL/DD/6915–20

FIGURE 13. COP402 Used to Emulate a COP420

Option List

COP402 MASK OPTIONS

The following COP420 options have been Implemented in this basic version of the COP402. Subsequent versions of the COP402 will implement different combinations of available options; such versions will be identified as COP402-A, COP402-B, etc.

Option Value	Comment
Option 1 = 0	Ground Pin—no option available
Option 2 = 0	CKO is clock generator output to crystal
Option 3 = 0	CKI is crystal input ÷ 16 (may be overridden externally)
Option 4 = 0	RESET pin has load device to V_{CC}
Option 5 = 2 (402) = 3 (402M)	L7 has LED direct-drive output L7 has TRI-STATE push-pull output
Option 6 = 2, 3	L6 same as L7
Option 7 = 2, 3	L5 same as L7
Option 8 = 2, 3	L4 same as L7
Option 9 = 0 (402) = 1 (402M)	IN1 has load device to V_{CC} Hi Z
Option 10 = 0 (402) = 1 (402M)	IN2 has load device to V_{CC} Hi Z
Option 11 = 0	V_{CC} pin—no option available
Option 12 = 2, 3	L3 same as L7
Option 13 = 2, 3	L2 same as L7
Option 14 = 2, 3	L1 same as L7
Option 15 = 2, 3	L0 same as L7
Option 16 = 0	SI has load device to V_{CC}
Option 17 = 2	SO has push-pull output
Option 18 = 2	SK has push-pull output
Option 19 = 0	IN0 has load device to V_{CC}
Option 20 = 0 (402) = 1 (402M)	IN3 has load device to V_{CC} Hi Z
Option 21 = 0	G0 has standard output
Option 22 = 0	G1 same as G0
Option 23 = 0	G2 same as G0
Option 24 = 0	G3 same as G0
Option 25 = 0	D3 has standard output
Option 26 = 0	D2 same as D3
Option 27 = 0	D1 same as D3
Option 28 = 0	D0 same as D3
Option 29 = 0 (402) = 1 (402M)	normal operation MICROBUS operation
Option 30 = N/A	40-pin package