UPI-C42/UPI-L42
UNIVERSAL PERIPHERAL INTERFACE
CHMOS 8-BIT SLAVE MICROCONTROLLER

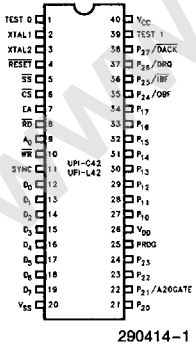
- **Pin, Software and Architecturally Compatible with all UPI-41 and UPI-42 Products**
- **Low Voltage Operation with the UPI-L42**
  — Full 3.3V Support
- **Hardware A20 Gate Support**
- **Suspend Power Down Mode**
- **Security Bit Code Protection Support**
- **8-Bit CPU plus ROM/OTP EPROM, RAM, I/O, Timer/Counter and Clock in a Single Package**
- **4096 x 8 ROM/OTP, 256 x 8 RAM 8-Bit Timer/Counter, 18 Programmable I/O Pins**
- **DMA, Interrupt, or Polled Operation Supported**
- **One 8-Bit Status and Two Data Registers for Asynchronous Slave-to-Master Interface**
- **Fully Compatible with all Intel and Most Other Microprocessor Families**
- **Interchangeable ROM and OTP EPROM Versions**
- **Expandable I/O**
- **Sync Mode Available**
- **Over 90 Instructions: 70% Single Byte**
- **Quick Pulse Programming Algorithm**
  — Fast OTP Programming
- **Available in 40-Lead Plastic, 44-Lead Plastic Leaded Chip Carrier, and 44-Lead Quad Flat Pack Packages**
  (See Packaging Spec., Order #240800, Package Type P, N, and S)

The UPI-C42 is an enhanced CHMOS version of the industry standard Intel UPI-42 family. It is fabricated on Intel's CHMOS III-E process. The UPI-C42 is pin, software, and architecturally compatible with the NMOS UPI family. The UPI-C42 has all of the same features of the NMOS family plus a larger user programmable memory array (4K), hardware A20 gate support, and lower power consumption inherent to a CHMOS product.

The UPI-L42 offers the same functionality and socket compatibility as the UPI-C42 as well as providing low voltage 3.3V operation.

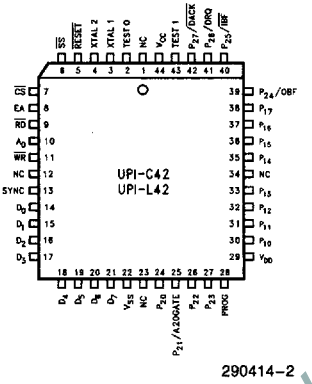
The UPI-C42 is essentially a "slave" microcontroller, or a microcontroller with a slave interface included on the chip. Interface registers are included to enable the UPI device to function as a slave peripheral controller in the MCS Modules and iAPX family, as well as other 8-, 16-, and 32-bit systems.

To allow full user flexibility, the program memory is available in ROM and One-Time Programmable EPROM (OTP).



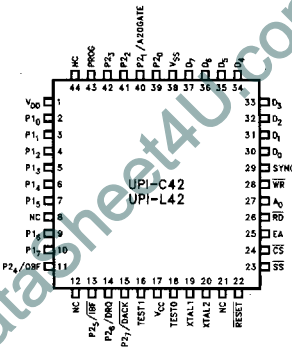
290414-1

Figure 1. DIP Pin Configuration



290414-2

Figure 2. PLCC Pin Configuration



290414-3

Figure 3. QFP Pin Configuration

Table 1. Pin Description

Symbol	DIP Pin No.	PLCC Pin No.	QFP Pin No.	Type	Name and Function
TEST 0, TEST 1	1 39	2 43	18 16	I	**TEST INPUTS:** Input pins which can be directly tested using conditional branch instructions. **FREQUENCY REFERENCE:** TEST 1 (T_1) functions as the event timer input (under software control). TEST 0 (T_0) is a multi-function pin used during PROM programming and ROM/EPROM verification, during Sync Mode to reset the instruction state to S1 and synchronize the internal clock to PH1.
XTAL 1	2	3	19	O	**OUTPUT:** Output from the oscillator amplifier.
XTAL 2	3	4	20	I	**INPUT:** Input to the oscillator amplifier and internal clock generator circuits.
$\overline{\text{RESET}}$	4	5	22	I	**RESET:** Input used to reset status flip-flops, set the program counter to zero, and force the UPI-C42 from the suspend power down mode. $\overline{\text{RESET}}$ is also used during EPROM programming and verification.
$\overline{\text{SS}}$	5	6	23	I	**SINGLE STEP:** Single step input used in conjunction with the SYNC output to step the program through each instruction (EPROM). This should be tied to +5V when not used. This pin is also used to put the device in Sync Mode by applying 12.5V to it.
$\overline{\text{CS}}$	6	7	24	I	**CHIP SELECT:** Chip select input used to select one UPI microcomputer out of several connected to a common data bus.
EA	7	8	25	I	**EXTERNAL ACCESS:** External access input which allows emulation, testing and ROM/EPROM verification. This pin should be tied low if unused.
$\overline{\text{RD}}$	8	9	26	I	**READ:** I/O read input which enables the master CPU to read data and status words from the OUTPUT DATA BUS BUFFER or status register.
A_0	9	10	27	I	**COMMAND/DATA SELECT:** Address Input used by the master processor to indicate whether byte transfer is data (A_0 = 0, F1 is reset) or command (A_0 = 1, F1 is set). A_0 = 0 during program and verify operations.
$\overline{\text{WR}}$	10	11	28	I	**WRITE:** I/O write input which enables the master CPU to write data and command words to the UPI INPUT DATA BUS BUFFER.
SYNC	11	13	29	O	**OUTPUT CLOCK:** Output signal which occurs once per UPI instruction cycle. SYNC can be used as a strobe for external circuitry; it is also used to synchronize single step operation.
D_0–D_7 (BUS)	12–19	14–21	30–37	I/O	**DATA BUS:** Three-state, bidirectional DATA BUS BUFFER lines used to interface the UPI microcomputer to an 8-bit master system data bus.
P_{10}–P_{17}	27–34	30–33 35–38	2–10	I/O	**PORT 1:** 8-bit, PORT 1 quasi-bidirectional I/O lines. P_{10}–P_{17} access the signature row and security bit.

Table 1. Pin Description (Continued)

Symbol	DIP Pin No.	PLCC Pin No.	QFP Pin No.	Type	Name and Function
P_{20}–P_{27}	21–24 35–38	24–27 39–42	39–42 11, 13–15	I/O	**PORT 2:** 8-bit, PORT 2 quasi-bidirectional I/O lines. The lower 4 bits (P_{20}–P_{23}) interface directly to the 8243 I/O expander device and contain address and data information during PORT 4–7 access. P_{21} can be programmed to provide hardware A20 gate support. The upper 4 bits (P_{24}–P_{27}) can be programmed to provide interrupt Request and DMA Handshake capability. Software control can configure P_{24} as Output Buffer Full (OBF) interrupt, P_{25} as Input Buffer Full ($\overline{IBF}$) interrupt, P_{26} as DMA Request (DRQ), and P_{27} as DMA ACKnowledge ($\overline{DACK}$).
PROG	25	28	43	I/O	**PROGRAM:** Multifunction pin used as the program pulse input during PROM programming. During I/O expander access the PROG pin acts as an address/data strobe to the 8243. This pin should be tied high if unused.
V_{CC}	40	44	17		**POWER:** +5V main power supply pin.
V_{DD}	26	29	1		**POWER:** +5V during normal operation. +12.75V during programming operation. Low power standby supply pin.
V_{SS}	20	22	38		**GROUND:** Circuit ground potential.

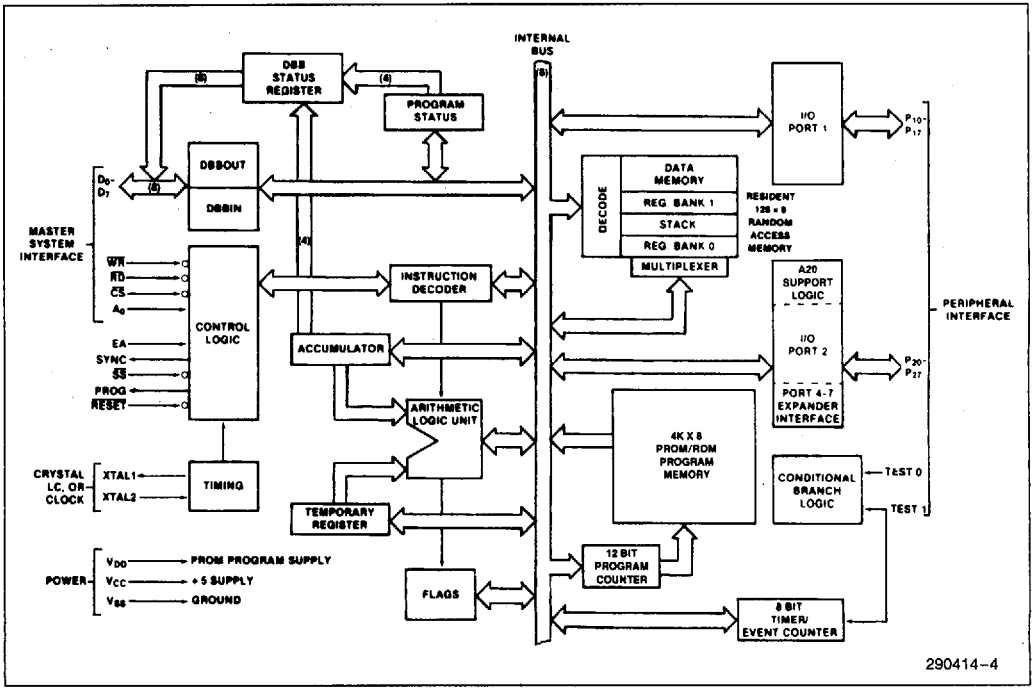


Figure 4. Block Diagram

UPI-C42/L42 PRODUCT SELECTION GUIDE

UPI-C42: Low power CHMOS version of the UPI-42.

Device	Package	ROM	OTP	Comments
80C42	N, P S	4K		ROM Device
82C42PC	N, P, S			Phoenix MultiKey/42 firmware, PS/2 style mouse support
82C42PD	N, P, S			Phoenix MultiKey/42L firmware, KBC and SCC for portable apps.
82C42PE	N, P, S			Phoenix MultiKey/42G firmware, Energy Efficient KBC solution
87C42	N, P, S		4K	One Time Programmable Version

UPI-L42: The low voltage 3.3V version of the UPI-C42.

Device	Package	ROM	OTP	Comments
80L42	N, P S	4K		ROM Device
82L42PC	N, P, S			Phoenix MultiKey/42 firmware, PS/2 style mouse support
82L42PD	N, P, S			Phoenix MultiKey/42L firmware, KBC and SCC for portable apps.
87L42	N, P, S		4K	One Time Programmable Version

N = 44 lead PLCC, P = 40 lead PDIP, S = 44 lead QFP, D = 40 lead CERDIP
KBC = Key Board Control, SCC = Scan Code Control

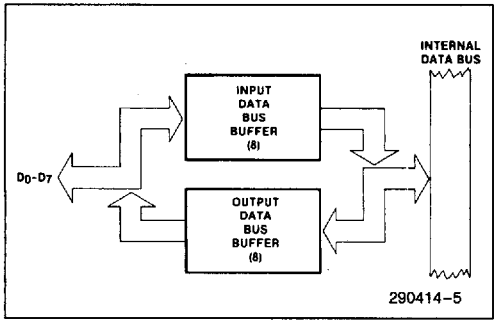
THE INTEL 82C42

As shown in the UPI-C42 product matrix, the UPI-C42 is offered as a pre-programmed 80C42 with various versions of MultiKey/42 keyboard controller firmware developed by Phoenix Technologies Ltd.

The 82C42PC provides a low powered solution for industry standard keyboard and PS/2 style mouse control. The 82C42PD provides a cost effective means for keyboard and scan code control for notebook platforms. The 82C42PE allows a quick time to market, low cost solution for energy efficient desktop designs.

UPI-42 COMPATIBLE FEATURES

1. Two Data Bus Buffers, one for input and one for output. This allows a much cleaner Master/Slave protocol.



2. 8 Bits of Status

ST_7	ST_6	ST_5	ST_4	F_1	F_0	IBF	OBF
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

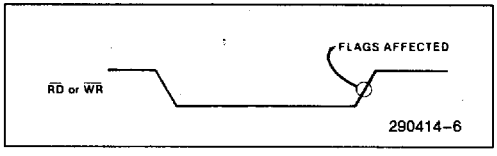
ST_4–ST_7 are user definable status bits. These bits are defined by the "MOV STS, A" single byte, single cycle instruction. Bits 4–7 of the acccumulator are moved to bits 4–7 of the status register. Bits 0–3 of the status register are not affected.

MOV STS, A Op Code: 90H

1	0	0	1	0	0	0	0
D_7							D_0

3. $\overline{RD}$ and $\overline{WR}$ are edge triggered. IBF, OBF, F_1 and INT change internally after the trailing edge of $\overline{RD}$ or $\overline{WR}$.

   During the time that the host CPU is reading the status register, the UPI is prevented from updating this register or is 'locked out.'

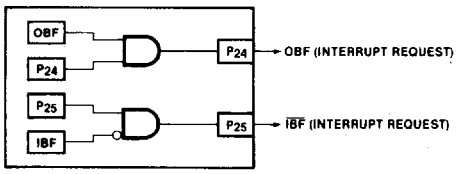


4. P_{24} and P_{25} are port pins or Buffer Flag pins which can be used to interrupt a master processor. These pins default to port pins on Reset.

   If the "EN FLAGS" instruction has been executed, P_{24} becomes the OBF (Output Buffer Full) pin. A "1" written to P_{24} enables the OBF pin (the pin outputs the OBF Status Bit). A "0" written to P_{24} disables the OBF pin (the pin remains low). This pin can be used to indicate that valid data is available from the UPI (in Output Data Bus Buffer).

   If "EN FLAGS" has been executed, P_{25} becomes the $\overline{IBF}$ (Input Buffer Full) pin. A "1" written to P_{25} enables the $\overline{IBF}$ pin (the pin outputs the inverse of the IBF Status Bit. A "0" written to P_{25} disables the $\overline{IBF}$ pin (the pin remains low). This pin can be used to indicate that the UPI is ready for data.

Data Bus Buffer Interrupt Capability



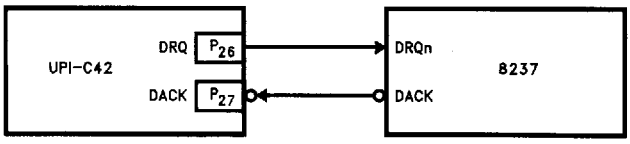
EN FLAGS Op Code: 0F5H

1	1	1	1	0	1	0	1
D_7							D_0

5. P_{26} and P_{27} are port pins or DMA handshake pins for use with a DMA controller. These pins default to port pins on Reset.

   If the "EN DMA" instruction has been executed, P_{26} becomes the DRQ (DMA Request) pin. A "1" written to P_{26} causes a DMA request (DRQ is activated). DRQ is deactivated by DACK•RD, DACK•WR, or execution of the "EN DMA" instruction.

DMA Handshake Capability



If "EN DMA" has been executed, P_{27} becomes the $\overline{DACK}$ (DMA ACKnowledge) pin. This pin acts as a chip select input for the Data Bus Buffer registers during DMA transfers.

EN DMA Op Code: 0E5H

1	1	1	0	0	1	0	1

D_7 D_0

6. When EA is enabled on the UPI, the program counter is placed on Port 1 and the lower four bits of Port 2 (MSB = P_{23}, LSB = P_{10}). On the UPI this information is multiplexed with PORT DATA (see port timing diagrams at end of this data sheet).
7. The UPI-C42 supports the Quick Pulse Programming Algorithm, but can also be programmed with the Intelligent Programming Algorithm. (See the Programming Section.)

UPI-C42 FEATURES

Programmable Memory Size Increase

The user programmable memory on the UPI-C42 will be increased from the 2K available in the NMOS product by 2X to 4K. The larger user programmable memory array will allow the user to develop more complex peripheral control micro-code. P2.3 (port 2 bit 3) has been designated as the extra address pin required to support the programming of the extra 2K of user programmable memory.

The new instruction SEL PMB1 (73h) allows for access to the upper 2K bank (locations 2048–4095). The additional memory is completely transparent to users not wishing to take advantage of the extra memory space. No new commands are required to access the lower 2K bytes. The SEL PMB0 (63h) has also been added to the UPI-C42 instruction set to allow for switching between memory banks.

Extended Memory Program Addressing (Beyond 2K)

For programs of 2K words or less, the UPI-C42 addresses program memory in the conventional manner. Addresses beyond 2047 can be reached by executing a program memory bank switch instruction (SEL PMB0, SEL PMB1) followed by a branch instruction (JMP or CALL). The bank switch feature extends the range of branch instructions beyond their normal 2K range and at the same time prevents the user from inadvertently crossing the 2K boundary.

PROGRAM MEMORY BANK SWITCH

The switching of 2K program memory banks is accomplished by directly setting or resetting the most significant bit of the program counter (bit 11); see Figure 5. Bit 11 is not altered by normal incrementing of the program counter, but is loaded with the contents of a special flip-flop each time a JMP or CALL instruction is executed. This special flip-flop is set by executing an SEL PMB1 instruction and reset by SEL PMB0. Therefore, the SEL PMB instruction may be executed at any time prior to the actual bank switch which occurs during the next branch instruction encountered. Since all twelve bits of the program counter, including bit 11, are stored in the stack, when a Call is executed, the user may jump to subroutines across the 2K boundary and the proper PC will be restored upon return. However, the bank switch flip-flop will not be altered on return.

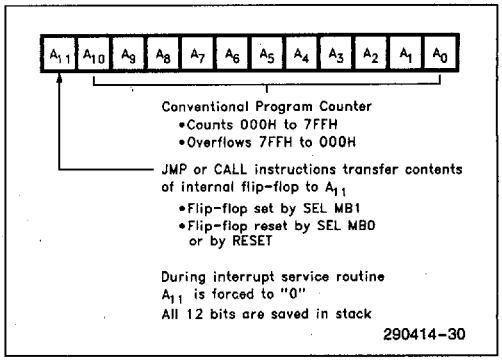


Figure 5. Program Counter

INTERRUPT ROUTINES

Interrupts always vector the program counter to location 3 or 7 in the first 2K bank, and bit 11 of the program counter is held at "0" during the interrupt service routine. The end of the service routine is signaled by the execution of an RETR instruction. Interrupt service routines should therefore be contained entirely in the lower 2K words of program memory. The execution of a SEL PMB0 or SEL PMB1 instruction within an interrupt routine is not recommended since it will not alter PC11 while in the routine, but will change the internal flip-flop.

Hardware A20 Gate Support

This feature has been provided to enhance the performance of the UPI-C42 when being used in a keyboard controller application. The UPI-C42 design has included on chip logic to support a hardware GATEA20 feature which eliminates the need to provide firmware to process A20 command sequences,

thereby providing additional user programmable memory space. This feature is enabled by the A20EN instruction and remains enabled until the device is reset. It is important to note that the execution of the A20EN instruction redefines Port 2, bit 1 as a pure output pin with read only characteristics. The state of this pin can be modified only through a valid "D1" command sequence (see Table 1). Once enabled, the A20 logic will process a "D1" command sequence (write to output port) by setting/resetting the A20 bit on port 2, bit 1 (P2.1) without requiring service from the internal CPU. The host can directly control the status of the A20 bit. At no time during this host interface transaction will the IBF flag in the status register be activated. Table 1 gives several possible GATEA20 command/data sequences and UPI-C42 responses.

Table 1. D1 Command Sequences

A0	R/W	DB Pins	IBF	A20	Comments
1	W	D1h	0	n(1)	Set A20 Sequence
0	W	DFh	0	1	Only DB1 Is Processed
1	W	FFH(2)	0	n	
1	W	D1h	0	n	Clear A20 Sequence
0	W	DDh	0	0	
1	W	FFh	0	n	
1	W	D1h	0	n	Double Trigger Set
1	W	D1h	0	n	Sequence
0	W	DFh	0	1	
1	W	FFh	0	n	
1	W	D1h	0	n	Invalid Sequence
1	W	XXh(3)	1	n	No Change in State
0	W	DDh	1	n	of A20 Bit

NOTES:
1. Indicates that P2.1 remains at the previous logic level.
2. Only FFh commands in a valid A20 sequence have no effect on IBF. An FFh issued at any other time will activate IBF.
3. Any command except D1.

The above sequences assume that the GATEA20 logic has been enabled via the A20EN instruction. As noted, only the value on DB 1 (data bus, bit 1) is processed. This bit will be directly passed through to P2.1 (port 2, bit 1).

SUSPEND

The execution of the suspend instruction (82h or E2h) causes the UPI-C42 to enter the suspend mode. In this mode of operation the oscillator is not running and the internal CPU operation is stopped. The UPI-C42 consumes $\leq 40\ \mu A$ in the suspend mode. This mode can only be exited by RESET. CPU operation will begin from PC = 000h when the UPI-C42 exits from the suspend power down mode.

Suspend Mode Summary

- Oscillator Not Running
- CPU Operation Stopped
- Ports Tristated with Weak ($\sim$2–10 μA) Pull-Up
- Micropower Mode ($I_{CC} \leq 40\ \mu A$)
- This mode is exited by RESET

5

Table 2 covers all suspend mode pin states. In addition to the suspend power down mode, the UPI-C42 will also support the NMOS power down mode as outlined in Chapter 4 of the UPI-42AH users manual.

Table 2. Suspend Mode Pin States

Pins	Suspend
Ports 1 and 2 Outputs Inputs	 Tristate Weak Pull-Up Disabled
DBB(1) Outputs Inputs	 Normal Normal
System Control (RD#, WR#, CS#, A0)	Disabled
Reset#	Enabled
Crystal Osc. (XTAL1, XTAL2)	Disabled
Test 0, Test 1	Disabled
Prog	High
Sync	High
EA	Disabled, No Pull-Up
SS#	Disabled, Weak Pull-Up
I_{CC}	<40 μA

NOTES:
1. DBB outputs are Tristate unless CS# and RD# are active. DBB inputs are disabled unless CS# and WR# are active.
2. A "disabled" input will not cause current to be drawn regardless of input level (within the supply range).
3. Weak pull-ups have current capability of typically 5 μA.

NEW UPI-C42 INSTRUCTIONS

The UPI-C42 will support several new instructions to allow for the use of new C42 features. These instructions are not necessary to the user who does not wish to take advantage of any new C42 functionality. The C42 will be completely compatible with all current NMOS code/applications. In order to use new features, however, some code modifications will be necessary. All new instructions can easily be inserted into existing code by use of the ASM-48 macro facility as shown in the following example:

```
Macname MACRO
        DB  63H
        ENDM
```

New Instructions

The following is a list of additions to the UPI-42 instruction set. These instructions apply only to the UPI-C42. These instructions must be added to existing code in order to use any new functionality.

SEL PMB0 Select Program Memory Bank 0

```
OPCODE  0110 0011 (63h)
```

PC Bit 11 is set to zero on next JMP or CALL instruction. All references to program memory fall within the range of 0–2047 (0–7FFh).

SEL PMB1 Select Program Memory Bank 1

```
OPCODE  0111 0011 (73h)
```

PC Bit 11 is set to one on next JMP or CALL instruction. All references to program memory fall within the range of 2048–4095 (800h–FFFh).

ENA20 Enables Auto A20 hardware

```
OPCODE  0011 0011 (33h)
```

Enables on chip logic to support Hardware A20 Gate feature. Will remain enabled until device is reset.

This circuitry gives the host direct control of port 2 bit 1 (P2.1) without intervention by the internal CPU. When this opcode is executed, P2.1 becomes a dedicated output pin. The status of this pin is read-able but can only be altered through a valid "D1" command sequence (see Table 1).

SUSPEND Invoke Suspend Power Down Mode

```
OPCODE  1000 0010 (82h) or 1110 0010
(E2h)
```

Enables device to enter micro power mode. In this mode the external oscillator is off, CPU operation is stopped, and the Port pins are tristated. This mode can only be exited via a RESET signal.

PROGRAMMING AND VERIFYING THE UPI-C42

The UPI-C42 programming will differ from the NMOS device in three ways. First, the C42 will have a 4K user programmable array. The UPI-C42 will also be programmed using the Intel Quick-Pulse Programming Algorithm. Finally, port 2 bit three (P2.3) will be used during program as the extra address pin required to program the upper 2K bank of additional memory. None of these differences have any effect on the full CHMOS to NMOS device compatibility. The extra memory is fully transparent to the user who does not need, or want, to use the extra memory space of the UPI-C42.

In brief, the programming process consists of: activating the program mode, applying an address, latching the address, applying data, and applying a programming pulse. Each word is programmed completely before moving on to the next and is followed by a verification step. The following is a list of the pins used for programming and a description of their functions:

Pin	Function
XTAL 2	Clock Input
$\overline{\text{Reset}}$	Initialization and Address Latching
Test 0	Selection of Program or Verify Mode
EA	Activation of Program/Verify Signature Row/Security Bit Modes
BUS	Address and Data Input Data Output During Verify
P_{20-23}	Address Input
V_{DD}	Programming Power Supply
PROG	Program Pulse Input

WARNING
An attempt to program a missocketed UPI-C42 will result in severe damage to the part. An indication of a properly socketed part is the appearance of the SYNC clock output. The lack of this clock may be used to disable the programmer.

The Program/Verify sequence is:

1. Insert 87C42 in programming socket
2. CS = 5V, V_{CC} = 5V, V_{DD} = 5V, RESET = 0V, A_0 = 0V, TEST 0 = 5V, clock applied or internal oscillator operating, BUS floating, PROG = 5V.
3. TEST 0 = 0V (select program mode)
4. EA = 12.75V (active program mode)
5. V_{CC} = 6.25V (programming supply)
6. V_{DD} = 12.75V (programming power)
7. Address applied to BUS and P_{20-23}
8. $\overline{\text{RESET}}$ = 5V (latch address)
9. Data applied to BUS
10. PROG = 5V followed by one 100 μs pulse to 0V
11. TEST 0 = 5V (verify mode)
12. Read and verify data on BUS
13. TEST 0 = 0V
14. $\overline{\text{RESET}}$ = 0V and repeat from step 6
15. Programmer should be at conditions of step 1 when the 87C42 is removed from socket

Please follow the Quick-Pulse Programming flow chart for proper programming procedure shown in Figure 6.

5

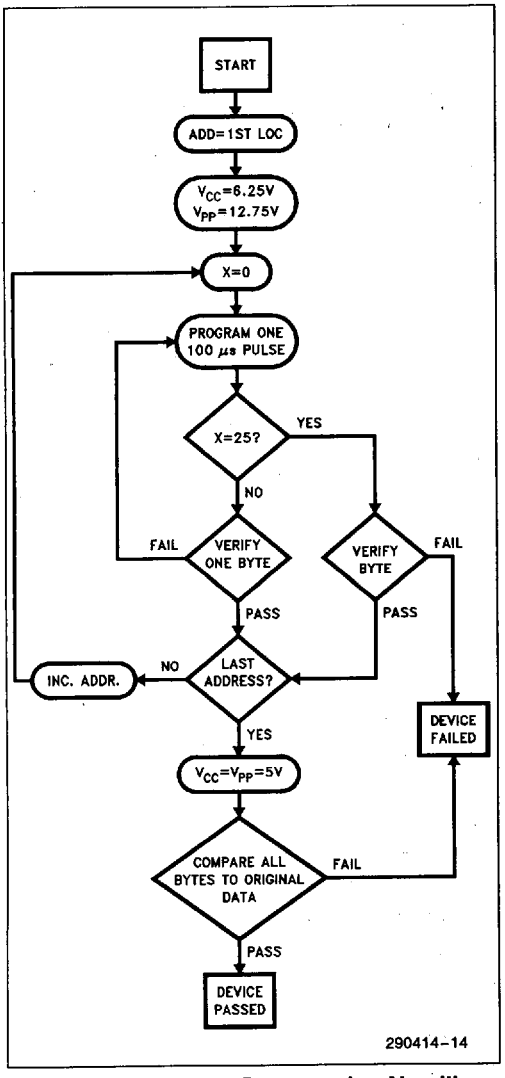


Figure 6. Quick-Pulse Programming Algorithm

Quick-Pulse Programming Algorithm

As previously stated, the UPI-C42 will be programmed using the Quick-Pulse Programming Algorithm, developed by Intel to substantially reduce the thorughput time in production programming.

The Quick-Pulse Programming Algorithm uses initial pulses of 100 μs followed by a byte verification to determine when the address byte has been successfully programmed. Up to 25 100 μs pulses per byte are provided before a failure is recognized. A flow chart of the Quick-Pulse Programming Algorithm is shown in Figure 6.

The entire sequence of program pulses and byte verifications is performed at V_{CC} = 6.25V and V_{DD} = 12.75V. When programming has been completed, all bytes should be compared to the original data with V_{CC} = V_{DD} = 5V.

A verify should be performed on the programmed bits to ensure that they have been correctly programmed. The verify is performed with T0 = 5V, V_{DD} = 5V, EA = 12.75V, SS# = 5V, PROG = 5V, A0 = 0V, and CS# = 5V.

In addition to the Quick-Pulse Programming Algorithm, the UPI-C42 OPT is also compatible with Intel's Int$_e$ligent Programming Algorithm which is used to program the NMOS UPI-42AH OTP devices.

The entire sequence of program pulses and byte verifications is performed at V_{CC} = 6.25V and V_{DD} = 12.75V. When the int$_e$ligent Programming cycle has been completed, all bytes should be compared to the original data with V_{CC} = 5.0, V_{DD} = 5V.

Verify

A verify should be performed on the programmed bits to determine that they have been correctly programmed. The verify is performed with T0 = 5V, V_{DD} = 5V, EA = 12.75V, $\overline{SS}$ = 5V, PROG = 5V, A0 = 0V, and $\overline{CS}$ = 5V..

SECURITY BIT

The security bit is a single EPROM cell outside the EPROM array. The user can program this bit with the appropriate access code and the normal programming procedure, to inhibit any external access to the EPROM contents. Thus the user's resident program is protected. There is no direct external access to this bit. However, the security byte in the signature row has the same address and can be used to check indirectly whether the security bit has been programmed or not. The security bit has no effect on the signature mode, so the security byte can always be examined.

SECURITY BIT PROGRAMMING/ VERIFICATION

Programming

a. Read the security byte of the signature mode. Make sure it is 00H.

b. Apply access code to appropriate inputs to put the device into security mode.

c. Apply high voltage to EA and V_{DD} pins.

d. Follow the programming procedure as per the Quick-Pulse Programming Algorithm with known data on the databus. Not only the security bit, but also the security byte of the signature row is programmed.

e. Verify that the security byte of the signature mode contains the same data as appeared on the data bus. (If DB0–DB7 = high, the security byte will contain FFH.)

f. Read two consecutive known bytes from the EPROM array and verify that the wrong data are retrieved in at least one verification. If the EPROM can still be read, the security bit may have not been fully programmed though the security byte in the signature mode has.

Verification

Since the security bit address overlaps the address of the security byte of the signature mode, it can be used to check indirectly whether the security bit has been programmed or not. Therefore, the security bit verification is a mere read operation of the security byte of the signature row (0FFH = security bit programmed; 00H = security bit unprogrammed). Note that during the security bit programming, the reading of the security byte does not necessarily indicate that the security bit has been successfully programmed. Thus, it is recommended that two consecutive known bytes in the EPROM array be read and the wrong data should be read at least once, because it is highly improbable that random data coincides with the correct ones twice.

SIGNATURE MODE

The UPI-C42 has an additional 64 bytes of EPROM available for Intel and user signatures and miscellaneous purposes. The 64 bytes are partitioned as follows:

A. **Test code/checksum**—This can accommodate up to 25 bytes of code for testing the internal nodes that are not testable by executing from the external memory. The test code/checksum is present on ROMs, and OTPs.

B. **Intel signature**—This allows the programmer to read from the UPI-41AH/42AH/C42 the manufacturer of the device and the exact product name. It facilitates automatic device identification and will be present in the ROM and OTP versions. Location 10H contains the manufacturer code. For Intel, it is 89H. Location 11H contains the device code.

   The code is 43H and 42H for the 8042AH/80C42 and OTP 8742AH/87C42, respectively. The code is 44H for any device with the security bit set by Intel.

C. **User signature**—The user signature memory is implemented in the EPROM and consists of 2 bytes for the customer to program his own signature code (for identification purposes and quick sorting of previously programmed materials).

D. **Test signature**—This memory is used to store testing information such as: test data, bin number, etc. (for use in quality and manufacturing control).

E. **Security byte**—This byte is used to check whether the security bit has been programmed (see the security bit section).

F. **UPI-C42 Intel Signature**—Applies only to CHMOS device. Location 20H contains the manufacturer code and location 21H contains the device code. The Intel UPI-C42 manufacturer's code is 99H. The device ID's are 82H for the OTP version and 83H for the ROM version. The device ID's are the same for the UPI-L42.

The signature mode can be accessed by setting P10 = 0, P11–P17 = 1, and then following the programming and/or verification procedures. The location of the various address partitions are as shown in Table 3.

5

SYNC MODE

The Sync Mode is provided to ease the design of multiple controller circuits by allowing the designer to force the device into known phase and state time. The Sync Mode may also be utilized by automatic test equipment (ATE) for quick, easy, and efficient synchronizing between the tester and the DUT (device under test).

Sync Mode is enabled when $\overline{SS}$ pin is raised to high voltage level of +12 volts. To begin synchronization, T0 is raised to 5 volts at least four clock cycles after $\overline{SS}$. T0 must be high for at least four X2 clock cycles to fully reset the prescaler and time state generators. T0 may then be brought down during low state of X2. Two clock cycles later, with the rising edge of X2, the device enters into Time State 1, Phase 1. $\overline{SS}$ is then brought down to 5 volts 4 clocks later after T0. RESET is allowed to go high 5 tCY (75 clocks) later for normal execution of code.

Table 3. Signature Mode Table

	Address	Device Type	No. of Bytes
Test Code/Checksum	0 0FH 16H 1EH	ROM/OTP	25
Intel Signature	10H 11H	ROM/OTP	2
User Signature	12H 13H	OTP	2
Test Signature	14H 15H	ROM/OTP	2
Security Byte	1FH or 3FH	ROM/OTP	2
UPI-C42 Intel Signature	20H 21H	ROM/OTP	2
User Defined UPI-C42 OTP EPROM Space	22H 3EH	ROM/OTP	30

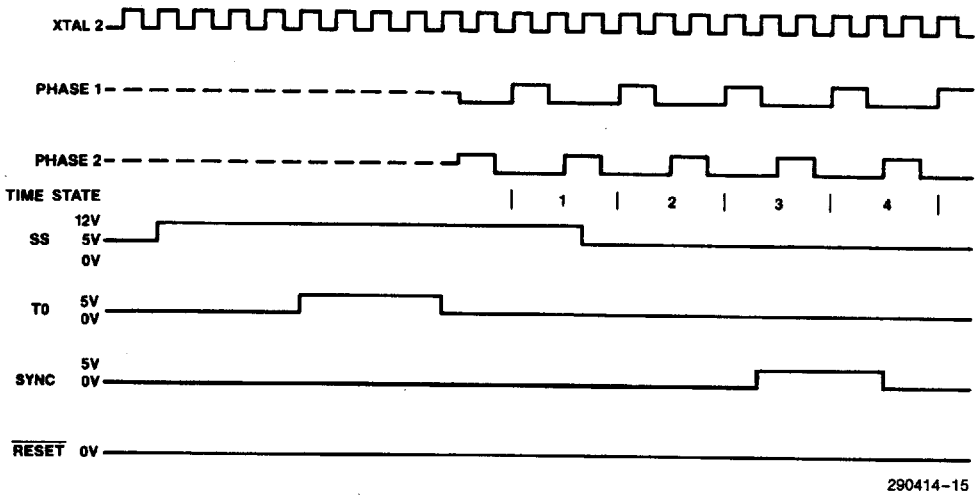
ACCESS CODE

The following table summarizes the access codes required to invoke the Sync Mode, Signature Mode, and the Security Bit, respectively. Also, the programming and verification modes are included for comparison.

Modes		Control Signals							Data Bus	Port 2	Access Code Port 1
		T0	RST	SS	EA	PROG	V_{DD}	V_{CC}	0 1 2 3 4 5 6 7	0 1 2 3	0 1 2 3 4 5 6 7
Programming Mode		0	0	1	HV	1	V_{DDH}	V_{CC}	Address	Addr	a_0 a_1 X X X X X X
		0	1	1	HV	STB	V_{DDH}	V_{CC}	Data In	Addr	
Verification Mode		0	0	1	HV	1	V_{CC}	V_{CC}	Address	Addr	a_0 a_1 X X X X X X
		1	1	1	HV	1	V_{CC}	V_{CC}	Data Out	Addr	
Sync Mode		STB High	0	HV	0	X	V_{CC}	V_{CC}	X X X X X X X X	X X X	X X X X X X X X
Signature Mode	Prog	0	0	1	HV	1	V_{DDH}	V_{CC}	Addr. (see Sig Mode Table)	0 0 0	0 1 1 1 1 X X 1
		0	1	1	HV	STB	V_{DDH}	V_{CC}	Data In	0 0 0	
	Verify	0	0	1	HV	1	V_{CC}	V_{CC}	Addr. (see Sig Mode Table)	0 0 0	
		1	1	1	HV	1	V_{CC}	V_{CC}	Data Out	0 0 0	
Security Bit/Byte	Prog	0	0	1	HV	1	V_{DDH}	V_{CC}	Address	0 0 0	
		0	1	1	HV	STB	V_{DDH}	V_{CC}	Data In	0 0 0	
	Verify	0	0	1	HV	1	V_{CC}	V_{CC}	Address	0 0 0	
		1	1	1	HV	1	V_{CC}	V_{CC}	Data Out	0 0 0	

NOTE:
1. a_0 = 0 or 1; a_1 = 0 or 1. a_0 must = a_1.

SYNC MODE TIMING DIAGRAMS



Minimum Specifications
SYNC Operation Time, t_{SYNC} = 3.5 XTAL 2 Clock cycles. Reset Time, t_{RS} = 4 t_{CY}.

NOTE:
The rising and falling edges of T0 should occur during low state of XTAL 2 clock.

APPLICATIONS

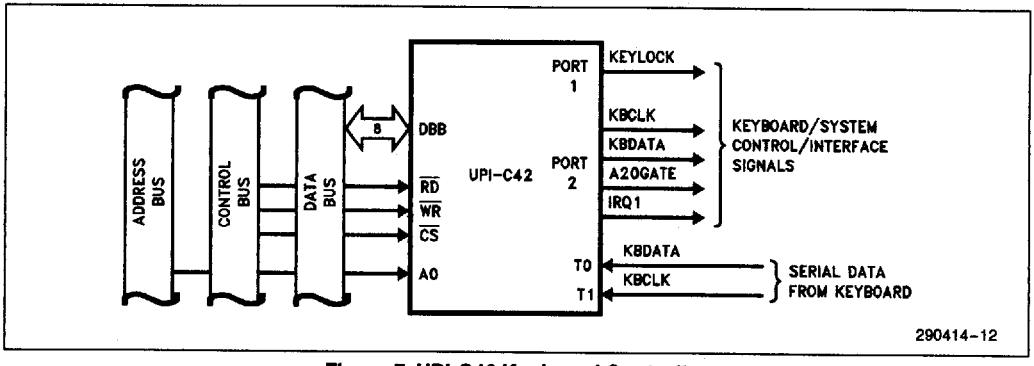


Figure 7. UPI-C42 Keyboard Controller

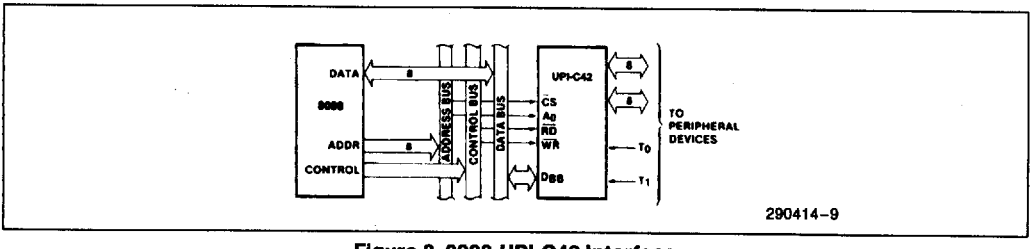


Figure 8. 8088-UPI-C42 Interface

APPLICATIONS (Continued)

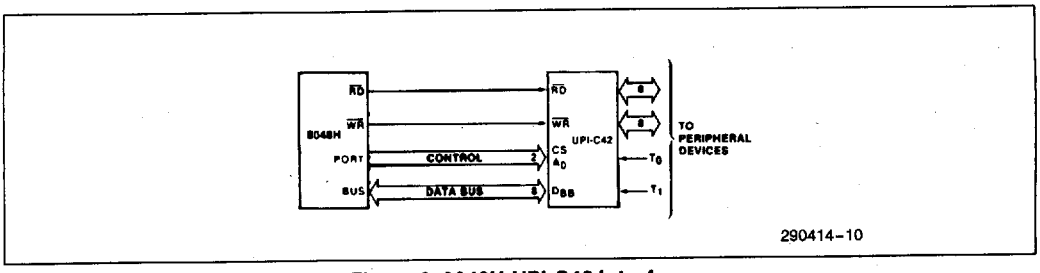


Figure 9. 8048H-UPI-C42 Interface

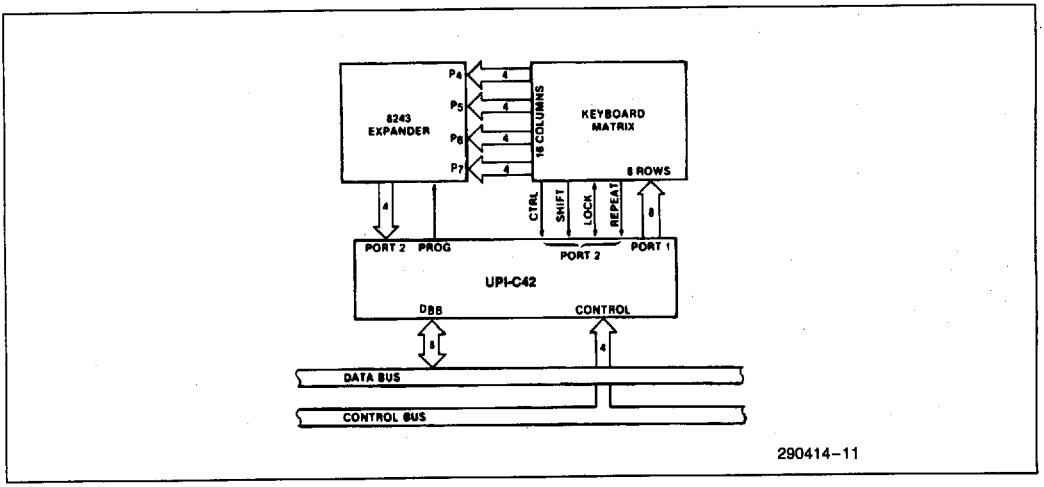


Figure 10. UPI-C42-8243 Keyboard Scanner

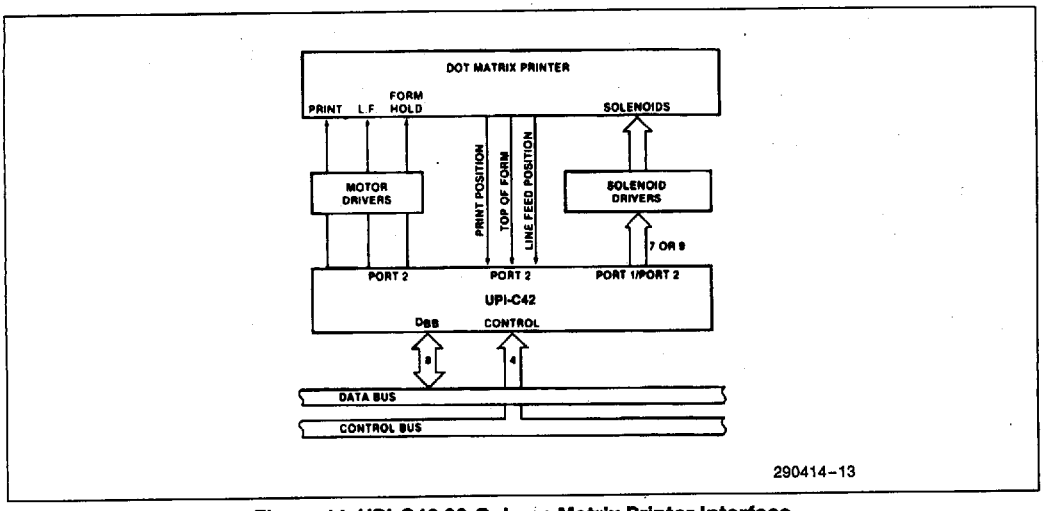


Figure 11. UPI-C42 80-Column Matrix Printer Interface

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias ....0°C to +70°C
Storage Temperature .....−65°C to +150°C
Voltage on Any Pin with
Respect to Ground.....−0.5V to +7V
Power Dissipation .....1.5 W

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

DC CHARACTERISTICS T_A = 0°C to +70°C, V_{CC} = V_{DD} = +5V ±10%; +3.3V ±10% UPI-L42

Symbol	Parameter	UPI-C42		UPI-L42		Units	Notes
		Min	Max	Min	Max		
V_{IL}	Input Low Voltage	−0.5	0.8	−0.3	+0.8	V	All Pins
V_{IH}	Input High Voltage (Except XTAL2, RESET)	2.0	V_{CC}	2.0	V_{CC} + 0.3	V	
V_{IH1}	Input High Voltage (XTAL2, RESET)	3.5	V_{CC}	2.0	V_{CC} + 0.3	V	
V_{OL}	Output Low Voltage (D_0–D_7)		0.45		0.45	V	I_{OL} = 2.0 mA UPI-C42 I_{OL} = 1.3 mA UPI-L42
V_{OL1}	Output Low Voltage ($P_{10}P_{17}$, $P_{20}P_{27}$, Sync)		0.45		0.45	V	I_{OL} = 1.6 mA UPI-C42 I_{OL} = 1 mA UPI-L42
V_{OL2}	Output Low Voltage (PROG)		0.45		0.45	V	I_{OL} = 1.0 mA UPI-C42 I_{OL} = 0.7 mA UPI-L42
V_{OH}	Output High Voltage (D_0–D_7)	2.4		2.4		V	I_{OH} = −400 μA UPI-C42 I_{OH} = −260 μA UPI-L42
V_{OH1}	Output High Voltage (All Other Outputs)	2.4		2.4			I_{OH} = −50 μA UPI-C42 I_{OH} = −25 μA UPI-L42
I_{IL}	Input Leakage Current (T_0, T_1, RD, WR, CS, A_0, EA)		±10		±10	μA	$V_{SS} \le V_{IN} \le V_{CC}$
I_{OFL}	Output Leakage Current (D_0–D_7, High Z State)		±10		±10	μA	V_{SS} + 0.45 ≤ V_{OUT} ≤ V_{CC}
I_{LI}	Low Input Load Current ($P_{10}P_{17}$, $P_{20}P_{27}$)	−50	−250	−35	−175	μA	Port Pins Min V_{IN} = 2.4V Max V_{IN} = 0.45V
I_{LI1}	Low Input Load Current (RESET, SS)		−40		−40	μA	$V_{IN} \le V_{IL}$
I_{HI}	Port Sink Current ($P_{10}P_{17}$, $P_{20}P_{27}$)				5.0	mA	V_{CC} = 3.0V V_{IH} = 5.0V
I_{DD}	V_{DD} Supply Current		4		2.5	mA	

5

DC CHARACTERISTICS

T_A = 0°C to +70°C, V_{CC} = V_{DD} = +5V ±10%; +3.3V ±10% UPI-L42 (Continued)

Symbol	Parameter	UPI-C42		UPI-L42		Units	Notes
		Min	Max	Min	Max		
$I_{CC} + I_{DD}$	Total Supply Current: Active Mode @ 12.5 MHz		30		20	mA	Typical 14 mA UPI-C42, 9 mA UPI-L42
	Suspend Mode		40		26	µA	Osc. Off(1, 4)
I_{DD} Standby	Power Down Supply Current		5		3.5	mA	NMOS Compatible Power Down Mode
I_{IH}	Input Leakage Current (P_{10}–P_{17}, P_{20}–P_{27})		100		100	µA	$V_{IN} = V_{CC}$
C_{IN}	Input Capacitance		10		10	pF	T_A = 25°C (1)
C_{IO}	I/O Capacitance		20		20	pF	T_A = 25°C (1)

NOTE:
1. Sampled, not 100% tested.

DC CHARACTERISTICS—PROGRAMMING (UPI-C42 AND UPI-L42)

T_A = 25°C ±5°C, V_{CC} = 6.25V ±0.25V, V_{DD} = 12.75V ±0.25V

Symbol	Parameter	Min	Max	Units
V_{DDH}	V_{DD} Program Voltage High Level	12.5	13	V(1)
V_{DDL}	V_{DD} Voltage Low Level	4.75	5.25	V
V_{PH}	PROG Program Voltage High Level	2.0	5.5	V
V_{PL}	PROG Voltage Low Level	−0.5	0.8	V
V_{EAH}	Input High Voltage for EA	12.0	13.0	V(2)
V_{EAL}	EA Voltage Low Level	−0.5	5.25	V
I_{DD}	V_{DD} High Voltage Supply Current		50.0	mA
I_{EA}	EA High Voltage Supply Current		1.0	mA(4)

NOTES:
1. Voltages over 13V applied to pin V_{DD} will permanently damage the device.
2. V_{EAH} must be applied to EA before V_{DDH} and removed after V_{DDL}.
3. V_{CC} must be applied simultaneously or before V_{DD} and must be removed simultaneously or after V_{DD}.
4. Sampled, not 100% tested.

AC CHARACTERISTICS

T_A = 0°C to +70°C, V_{SS} = 0V, V_{CC} = V_{DD} = +5V ±10%; +3.3V ±10% for the UPI-L42

NOTE:
All AC Characteristics apply to both the UPI-C42 and UPI-L42

DBB READ

Symbol	Parameter	Min	Max	Units
t_{AR}	CS, A_0 Setup to RD ↓	0		ns
t_{RA}	CS, A_0 Hold After RD ↑	0		ns
t_{RR}	RD Pulse Width	160		ns
t_{AD}	CS, A_0 to Data Out Delay		130	ns
t_{RD}	RD ↓ to Data Out Delay	0	130	ns
t_{DF}	RD ↑ to Data Float Delay		85	ns

DBB WRITE

Symbol	Parameter	Min	Max	Units
t_{AW}	CS, A_0 Setup to WR ↓	0		ns
t_{WA}	CS, A_0 Hold After WR ↑	0		ns
t_{WW}	WR Pulse Width	160		ns
t_{DW}	Data Setup to WR ↑	130		ns
t_{WD}	Data Hold After WR ↑	0		ns

5

AC CHARACTERISTICS

T_A = 0°C to +70°C, V_{SS} = 0V, V_{CC} = V_{DD} = +5V ±10%; +3.3V ±10% for the UPI-L42 (Continued)

CLOCK

Symbol	Parameter	Min	Max	Units
t_{CY} UPI-C42/UPI-L42	Cycle Time	1.2	9.20	μs(1)
t_{CYC} UPI-C42/UPI-L42	Clock Period	80	613	ns
t_{PWH}	Clock High Time	30		ns
t_{PWL}	Clock Low Time	30		ns
t_R	Clock Rise Time		10	ns
t_F	Clock Fall Time		10	ns

NOTE:
1. t_{CY} = 15/f(XTAL)

AC CHARACTERISTICS DMA

Symbol	Parameter	Min	Max	Units
t_{ACC}	DACK to WR or RD	0		ns
t_{CAC}	RD or WR to DACK	0		ns
t_{ACD}	DACK to Data Valid	0	130	ns
t_{CRQ}	RD or WR to DRQ Cleared		110	ns(1)

NOTE:
1. C_L = 150 pF.

AC CHARACTERISTICS PORT 2

Symbol	Parameter	f(t_{CY})(3)	Min	Max	Units
t_{CP}	Port Control Setup Before Falling Edge of PROG	1/15 t_{CY} − 28	55		ns(1)
t_{PC}	Port Control Hold After Falling Edge of PROG	1/10 t_{CY}	125		ns(2)
t_{PR}	PROG to Time P2 Input Must Be Valid	8/15 t_{CY} − 16		650	ns(1)
t_{PF}	Input Data Hold Time		0	150	ns(2)
t_{DP}	Output Data Setup Time	2/10 t_{CY}	250		ns(1)
t_{PD}	Output Data Hold Time	1/10 t_{CY} − 80	45		ns(2)
t_{PP}	PROG Pulse Width	6/10 t_{CY}	750		ns

NOTES:
1. C_L = 80 pF.
2. C_L = 20 pF.
3. t_{CY} = 1.25 μs.

AC CHARACTERISTICS—PROGRAMMING (UPI-C42 AND UPI-L42)

T_A = 25°C ±5°C, V_{CC} = 6.25V ±0.25V, V_{DDL} = +5V ±0.25V, V_{DDH} = 12.75V ±0.25V

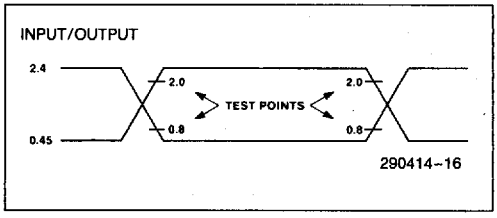
(87C42/87L42 ONLY)

Symbol	Parameter	Min	Max	Units
t_{AW}	Address Setup Time to RESET ↑	$4t_{CY}$		
t_{WA}	Address Hold Time after RESET ↑	$4t_{CY}$		
t_{DW}	Data in Setup Time to PROG ↓	$4t_{CY}$		
t_{WD}	Data in Hold Time after PROG ↑	$4t_{CY}$		
t_{PW}	Initial Program Pulse Width	95	105	μs
t_{TW}	Test 0 Setup Time for Program Mode	$4t_{CY}$		
t_{WT}	Test 0 Hold Time after Program Mode	$4t_{CY}$		
t_{DO}	Test 0 to Data Out Delay		$4t_{CY}$	
t_{WW}	RESET Pulse Width to Latch Address	$4t_{CY}$		
t_r, t_f	PROG Rise and Fall Times	0.5	100	μs
t_{CY}	CPU Operation Cycle Time	2.5	3.75	μs
t_{RE}	RESET Setup Time before EA ↑	$4t_{CY}$		
t_{OPW}	Overprogram Pulse Width	2.85	78.75	ms(1)
t_{DE}	EA High to V_{DD} High	$1t_{CY}$		

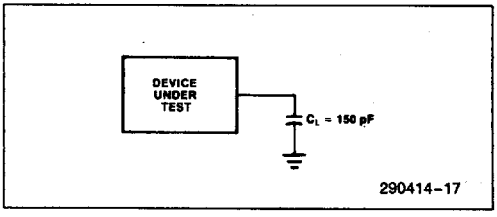
NOTES:
1. This variation is a function of the iteration counter value, X.
2. If TEST 0 is high, t_{DO} can be triggered by RESET ↑.

5

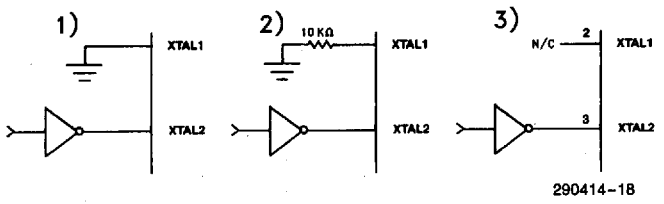
AC TESTING INPUT/OUTPUT WAVEFORM



AC TESTING LOAD CIRCUIT

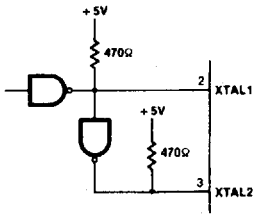


DRIVING FROM AN EXTERNAL SOURCE



290414-18

NOTE:
See XTAL1 Configuration Table.

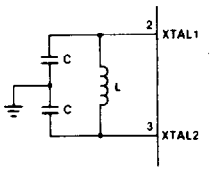


290414-19

Rise and Fall Times Should Not Exceed 10 ns. Resistors to V_{CC} are Needed to Ensure V_{IH} = 3.5V if TTL Circuitry is Used.

LC OSCILLATOR MODE

L	C	NOMINAL
45 H	20 pF	5.2 MHz
120 H	20 pF	3.2 MHz



$$f = \frac{1}{2\pi\sqrt{LC'}}$$

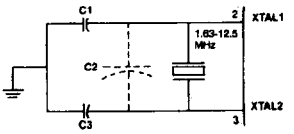
$$C' = \frac{C + 3Cpp}{2}$$

Cpp ≅ 5–10 pF
Pin-to-Pin Capacitance

290414-20

Each C Should be Approximately 20 pF, including Stray Capacitance.

CRYSTAL OSCILLATOR MODE



290414-21

C1 5 pF (STRAY 5 pF)
C2 (CRYSTAL + STRAY) 8 pF
C3 20–30 pF INCLUDING STRAY

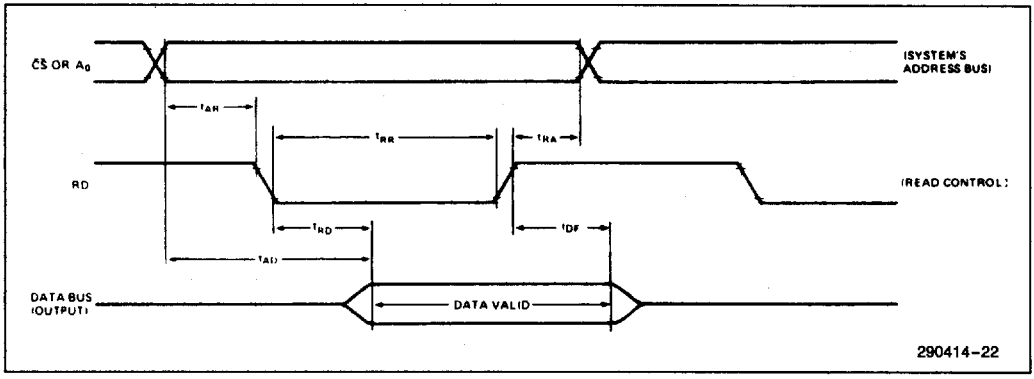
Crystal Series Resistance Should be Less Than 30Ω at 12.5 MHz.

XTAL1 Configuration Table

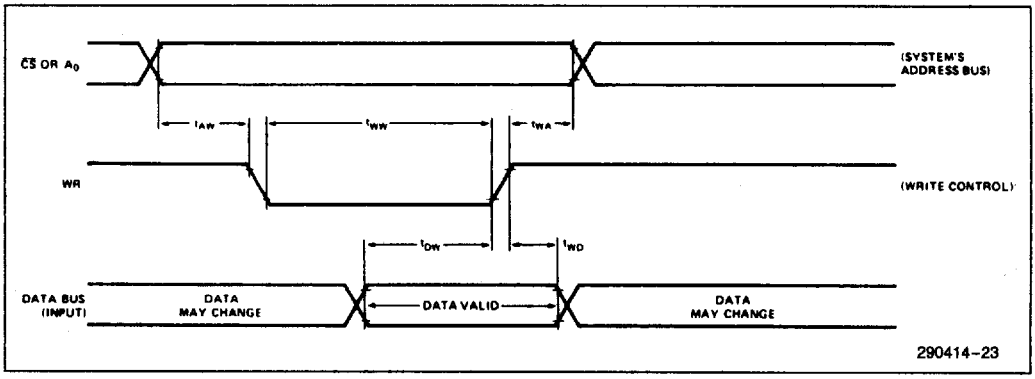
XTAL1 Connection		
1) to Ground	**2) 10 KΩ Resistor to Ground**	**3) Not Connected**
Not recommended for CHMOS designs. Causes approximately 16 mA of additional current flow through the XTAL1 pin on UPI-C42 and approximately 11 mA of additional current through XTAL1 on the UPI-L42.	Recommended configuration for designs which will use both NMOS and CHMOS parts. This configuration limits the additional current through the XTAL1 pin to approximately 1 mA, while maintaining compatibility with the NMOS device.	Low power configuration recommended for CHMOS only designs to provide lowest possible power consumption. This configuration will not work with the NMOS device.

WAVEFORMS

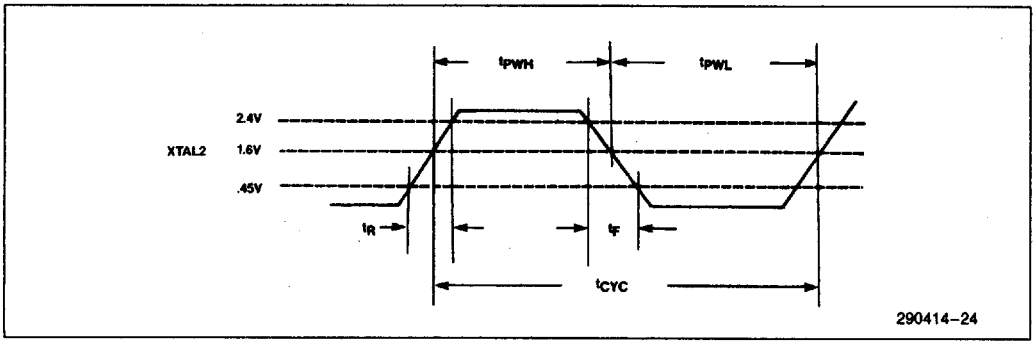
READ OPERATION—DATA BUS BUFFER REGISTER



WRITE OPERATION—DATA BUS BUFFER REGISTER



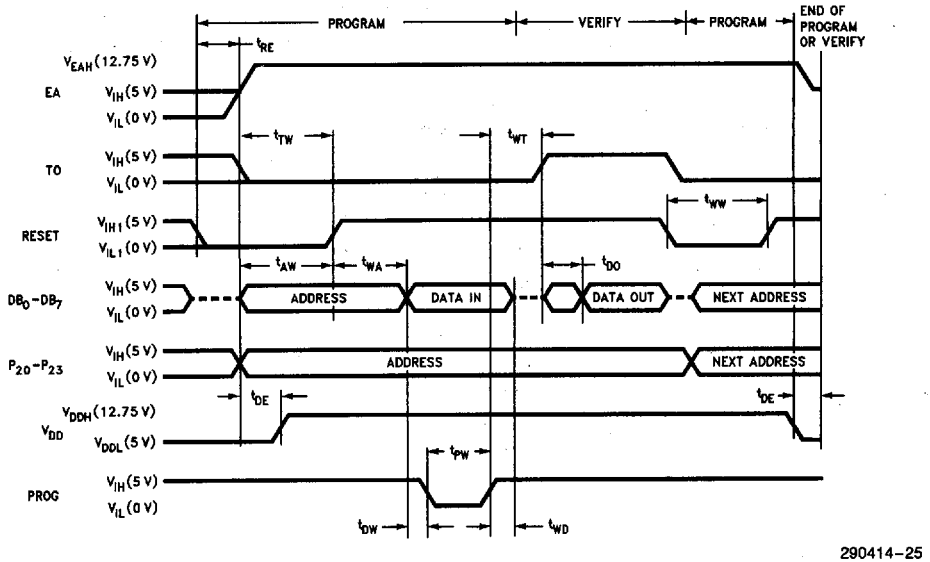
CLOCK TIMING



5

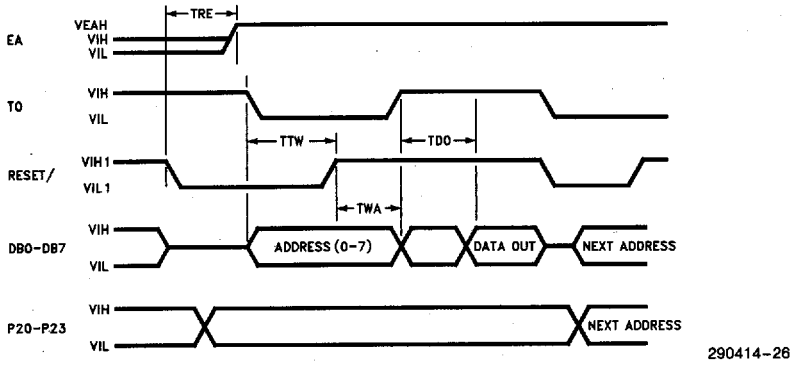
WAVEFORMS (Continued)

COMBINATION PROGRAM/VERIFY MODE



NOTES:
1. A_0 must be held low (0V) during program/verify modes.
2. For V_{IH}, V_{IH1}, V_{IL}, V_{IL1}, V_{DDH}, and V_{DDL}, please consult the D.C. Characteristics Table.
3. When programming the 87C42, a 0.1 μF capacitor is required across V_{DD} and ground to suppress spurious voltage transients which can damage the device.

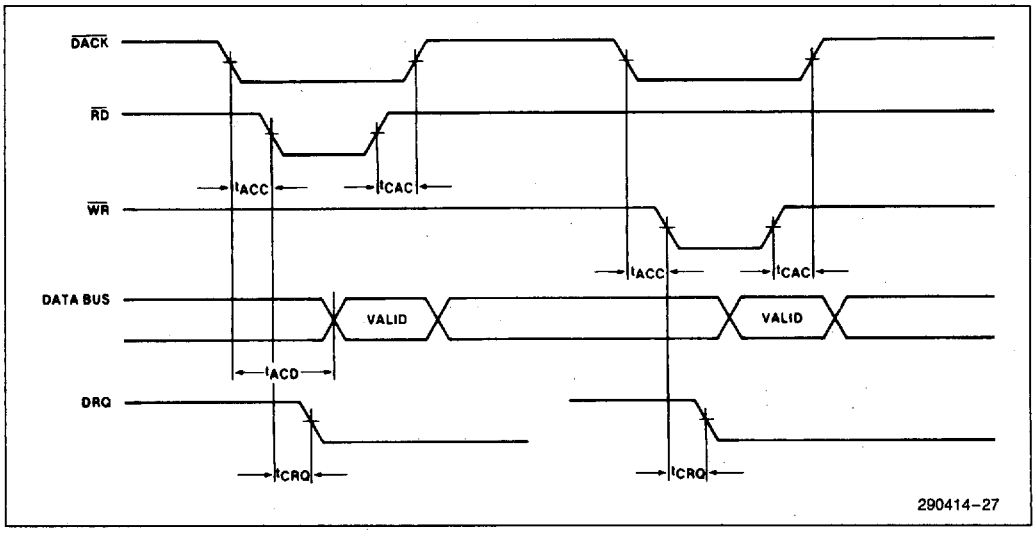
VERIFY MODE



NOTES:
1. PROG must float if EA is low.
2. PROG must float or = 5V when EA is high.
3. P_{10}-P_{17} = 5V or must float.
4. P_{24}-P_{27} = 5V or must float.
5. A_0 must be held low during programming/verify modes.

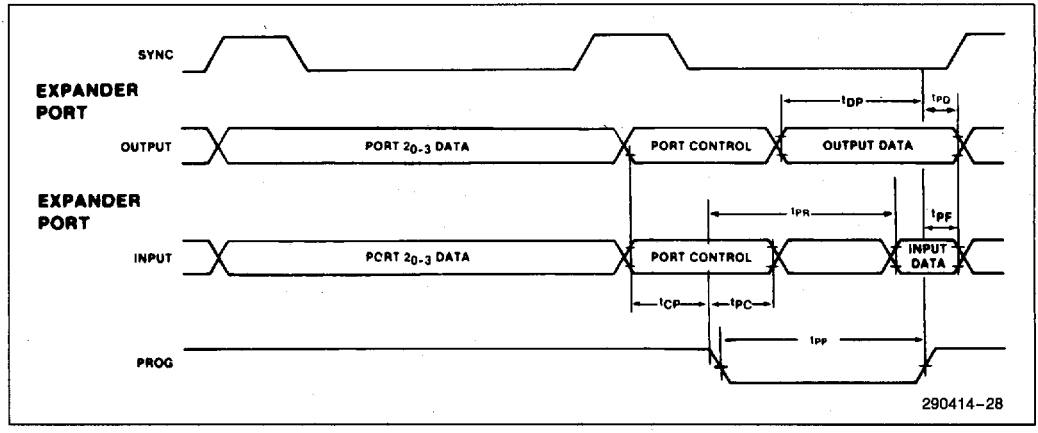
WAVEFORMS (Continued)

DMA



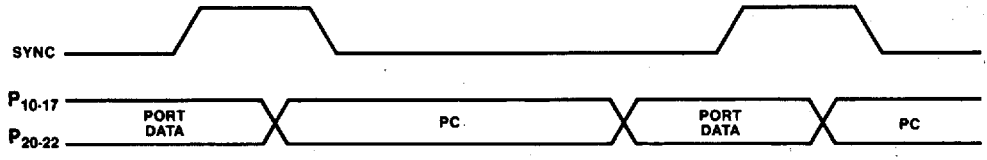
290414-27

PORT 2



290414-28

PORT TIMING DURING EXTERNAL ACCESS (EA)



290414-29

On the Rising Edge of SYNC and EA is Enabled, Port Data is Valid and can be Strobed. On the Trailing Edge of Sync the Program Counter Contents are Available.

Table 4. UPI Instruction Set

Mnemonic	Description	Bytes	Cycles
ACCUMULATOR			
ADD A, Rr	Add register to A	1	1
ADD A, @Rr	Add data memory to A	1	1
ADD A, #data	Add immediate to A	2	2
ADDC A, Rr	Add register to A with carry	1	1
ADDC A, @Rr	Add data memory to A with carry	1	1
ADDC A, #data	Add immediate to A with carry	2	2
ANL A, Rr	AND register to A	1	1
ANL, A @Rr	AND data memory to A	1	1
ANL A, #data	AND immediate to A	2	2
ORL A, Rr	OR register to A	1	1
ORL, A, @Rr	OR data memory to A	1	1
ORL A, #data	OR immediate to A	2	2
XRL A, Rr	Exclusive OR register to A	1	1
XRL A, @Rr	Exclusive OR data memory to A	1	1
XRL A, #data	Exclusive OR immediate to A	2	2
INC A	Increment A	1	1
DEC A	Decrement A	1	1
CLR A	Clear A	1	1
CPL A	Complement A	1	1
DA A	Decimal Adjust A	1	1
SWAP A	Swap nibbles of A	1	1
RL A	Rotate A left	1	1
RLC A	Rotate A left through carry	1	1
RR A	Rotate A right	1	1
RRC A	Rotate A right through carry	1	1
INPUT/OUTPUT			
IN A, Pp	Input port to A	1	2
OUTL Pp, A	Output A to port	1	2
ANL Pp, #data	AND immediate to port	2	2
ORL Pp, #data	OR immediate to port	2	2
IN A, DBB	Input DBB to A, clear IBF	1	1
OUT DBB, A	Output A to DBB, set OBF	1	1
MOV STS, A	A_4–A_7 to Bits 4–7 of Status	1	1
MOVD A, Pp	Input Expander port to A	1	2
MOVD Pp, A	Output A to Expander port	1	2
ANLD Pp, A	AND A to Expander port	1	2
ORLD Pp, A	OR A to Expander port	1	2

Mnemonic	Description	Bytes	Cycles
DATA MOVES			
MOV A, Rr	Move register to A	1	1
MOV A, @Rr	Move data memory to A	1	1
MOV A, #data	Move immediate to A	2	2
MOV Rr, A	Move A to register	1	1
MOV @Rr, A	Move A to data memory	1	1
MOV Rr, #data	Move immediate to register	2	2
MOV @Rr, #data	Move immediate to data memory	2	2
MOV A, PSW	Move PSW to A	1	1
MOV PSW, A	Move A to PSW	1	1
XCH A, Rr	Exchange A and register	1	1
XCH A, @Rr	Exchange A and data memory	1	1
XCHD A, @Rr	Exchange digit of A and register	1	1
MOVP A, @A	Move to A from current page	1	2
MOVP3, A, @A	Move to A from page 3	1	2
TIMER/COUNTER			
MOV A, T	Read Timer/Counter	1	1
MOV T, A	Load Timer/Counter	1	1
STRT T	Start Timer	1	1
STRT CNT	Start Counter	1	1
STOP TCNT	Stop Timer/Counter	1	1
EN TCNTI	Enable Timer/Counter Interrupt	1	1
DIS TCNTI	Disable Timer/Counter Interrupt	1	1
CONTROL			
*EN A20	Enable A20 Logic	1	1
EN DMA	Enable DMA Handshake Lines	1	1
EN I	Enable IBF Interrupt	1	1
DIS I	Diable IBF Interrupt	1	1
EN FLAGS	Enable Master Interrupts	1	1
*SEL PMB0	Select Program memory bank 0	1	1
*SEL PMB1	Select Program memory bank 1	1	1
SEL RB0	Select register bank 0	1	1
SEL RB1	Select register bank 1	1	1

* UPI-C42/UPI-L42 Only.

Table 4. UPI Instruction Set (Continued)

Mnemonic	Description	Bytes	Cycles
CONTROL (Continued)			
*SUSPEND	Invoke Suspend Power-down mode	1	2
NOP	No Operation	1	1
REGISTERS			
INC Rr	Increment register	1	1
INC @Rr	Increment data memory	1	1
DEC Rr	Decrement register	1	1
SUBROUTINE			
CALL addr	Jump to subroutine	2	2
RET	Return	1	2
RETR	Return and restore status	1	2
FLAGS			
CLR C	Clear Carry	1	1
CPL C	Complement Carry	1	1
CLR F0	Clear Flag 0	1	1
CPL F0	Complement Flag 0	1	1
CLR F1	Clear F1 Flag	1	1
CPL F1	Complement F1 Flag	1	1

Mnemonic	Description	Bytes	Cycles
BRANCH			
JMP addr	Jump unconditional	2	2
JMPP @A	Jump indirect	1	2
DJNZ Rr, addr	Decrement register and jump	2	2
JC addr	Jump on Carry = 1	2	2
JNC addr	Jump on Carry = 0	2	2
JZ addr	Jump on A Zero	2	2
JNZ addr	Jump on A not Zero	2	2
JT0 addr	Jump on T0 = 1	2	2
JNT0 addr	Jump on T0 = 0	2	2
JT1 addr	Jump on T1 = 1	2	2
JNT1 addr	Jump on T1 = 0	2	2
JF0 addr	Jump on F0 Flag = 1	2	2
JF1 addr	Jump on F1 Flag = 1	2	2
JTF addr	Jump on Timer Flag = 1, Clear Flag	2	2
JNIBF addr	Jump on IBF Flag = 0	2	2
JOBF addr	Jump on OBF Flag = 1	2	2
JBb addr	Jump on Accumula-for Bit	2	2

*UPI-C42/UPI-L42 Only.

REVISION SUMMARY

The following has been changed since Revision -003:

1. Delete all references to standby power down mode.

The following has been changed since Revision -002:

1. Added information on keyboard controller product family.
2. Added I_{HI} specification for the UPI-L42.

The following has been changed since Revision -001:

1. Added UPI-L42 references and specification.

5