National Semiconductor

HPC16064/26064/36064/46064/16004/26004/36004/46004 High-Performance microController

General Description

The HPC46064 and HPC46004 are members of the HPC™ family of High Performance microControllers. Each member of the family has the same core CPU with a unique memory and I/O configuration to suit specific applications. The HPC46064 has 16k bytes of on-chip ROM. The HPC46004 has no on-chip ROM and is intended for use with external memory. Each part is fabricated in National's advanced microCMOS technology. This process combined with an advanced architecture provides fast, flexible I/O control, efficient data manipulation, and high speed computation.

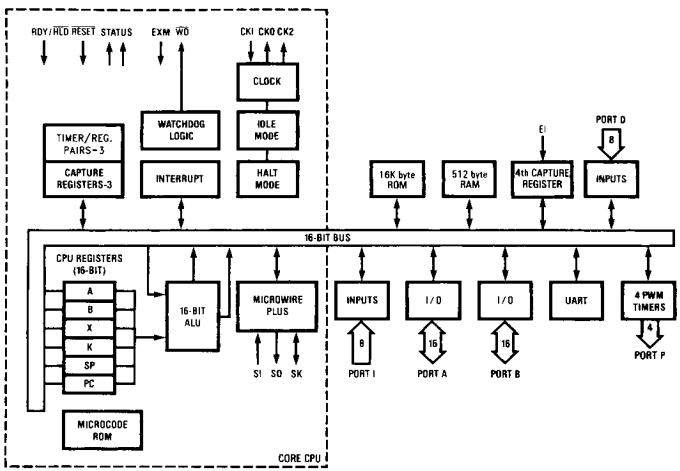
The HPC devices are complete microcomputers on a single chip. All system timing, internal logic, ROM, RAM, and I/O are provided on the chip to produce a cost effective solution for high performance applications. On-chip functions such as UART, up to eight 16-bit timers with 4 input capture registers, vectored interrupts, WATCHDOG™ logic and MICROWIRE/PLUS™ provide a high level of system integration. The ability to address up to 64k bytes of external memory enables the HPC to be used in powerful applications typically performed by microprocessors and expensive peripheral chips. The term "HPC46064" is used throughout this datasheet to refer to the HPC46064 and HPC46004 devices unless otherwise specified.

The microCMOS process results in very low current drain and enables the user to select the optimum speed/power product for his system. The IDLE and HALT modes provide further current savings. The HPC is available in 68-pin PLCC, LDCC, PGA and 80-pin PQFP package.

Features

- HPC family—core features:
  - — 16-bit architecture, both byte and word
  - — 16-bit data bus, ALU, and registers
  - — 64k bytes of external direct memory addressing
  - — FAST—200 ns for fastest instruction when using 20.0 MHz clock, 134 ns at 30.0 MHz
  - — High code efficiency—most instructions are single byte
  - — 16 x 16 multiply and 32 x 16 divide
  - — Eight vectored interrupt sources
  - — Four 16-bit timer/counters with 4 synchronous outputs and WATCHDOG logic
  - — MICROWIRE/PLUS serial I/O interface
  - — CMOS—very low power with two power save modes: IDLE and HALT
- UART—full duplex, programmable baud rate
- Four additional 16-bit timer/counters with pulse width modulated outputs
- Four input capture registers
- 52 general purpose I/O lines (memory mapped)
- 16k bytes of ROM, 512 bytes of RAM on-chip
- ROMless version available (HPC46004)
- Commercial (0°C to +70°C), industrial (−40°C to +85°C), automotive (−40°C to +105°C) and military (−55°C to +125°C) temperature ranges

Block Diagram (HPC46064 with 16k ROM shown)



TL/DD/11372-1

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Absolute Maximum Ratings

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

Total Allowable Source or Sink Current	100 mA
Storage Temperature Range	−65°C to +150°C
Lead Temperature (Soldering, 10 sec.)	300°C
V_{CC} with Respect to GND	−0.5V to 7.0V
All Other Pins	(V_{CC} + 0.5)V to (GND − 0.5)V

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

DC Electrical Characteristics

V_{CC} = 5V ±10%, T_A = 0°C to +70°C for HPC46064/46004, −40°C to +85°C for HPC36064/36004, −40°C to +105°C for HPC26064/26004, −55°C to +125°C for HPC16064/16004

Symbol	Parameter	Test Conditions	Min	Max	Units
I_{CC1}	Supply Current	V_{CC} = 5.5V, f_{in} = 30 MHz (Note 1)		65	mA
		V_{CC} = 5.5V, f_{in} = 20 MHz (Note 1)		47	mA
		V_{CC} = 5.5V, f_{in} = 2.0 MHz (Note 1)		10	mA
I_{CC2}	IDLE Mode Current	V_{CC} = 5.5V, f_{in} = 30 MHz (Note 1)		5	mA
		V_{CC} = 5.5V, f_{in} = 20 MHz (Note 1)		3.0	mA
		V_{CC} = 5.5V, f_{in} = 2.0 MHz (Note 1)		1	mA
I_{CC3}	HALT Mode Current	V_{CC} = 5.5V, f_{in} = 0 kHz (Note 1)		300	μA
		V_{CC} = 2.5V, f_{in} = 0 kHz (Note 1)		100	μA
INPUT VOLTAGE LEVELS FOR SCHMITT TRIGGERED INPUTS, $\overline{RESET}$, NMI, AND $\overline{WO}$; AND ALSO CKI					
V_{IH1}	Logic High		0.9 V_{CC}		V
V_{IL1}	Logic Low			0.1 V_{CC}	V
ALL OTHER INPUTS					
V_{IH2}	Logic High		0.7 V_{CC}		V
V_{IL2}	Logic Low			0.2 V_{CC}	V
I_{LI1}	Input Leakage Current	V_{IN} = 0 and V_{IN} = V_{CC}		±2	μA
I_{LI2}	Input Leakage Current RDY/$\overline{HLD}$, $\overline{EXUI}$	V_{IN} = 0	−3	−50	μA
I_{LI3}	Input Leakage Current B12	$\overline{RESET}$ = 0, V_{IN} = V_{CC}	0.5	7	μA
C_I	Input Capacitance	(Note 2)		10	pF
C_{IO}	I/O Capacitance	(Note 2)		20	pF
OUTPUT VOLTAGE LEVELS					
V_{OH1}	Logic High (CMOS)	I_{OH} = −10 μA (Note 2)	V_{CC} − 0.1		V
V_{OL1}	Logic Low (CMOS)	I_{OH} = 10 μA (Note 2)		0.1	V
V_{OH2}	Port A/B Drive, CK2 (A_0–A_{15}, B_{10}, B_{11}, B_{12}, B_{15})	I_{OH} = −7 mA	2.4		V
V_{OL2}		I_{OL} = 3 mA		0.4	V
V_{OH3}	Other Port Pin Drive, $\overline{WO}$ (open drain) (B_0–B_9, B_{13}, B_{14}, P_0–P_3)	I_{OH} = −1.6 mA (except $\overline{WO}$)	2.4		V
V_{OL3}		I_{OL} = 0.5 mA		0.4	V
V_{OH4}	ST1 and ST2 Drive	I_{OH} = −6 mA	2.4		V
V_{OL4}		I_{OL} = 1.6 mA		0.4	V
V_{OH5}	Port A/B Drive (A_0–A_{15}, B_{10}, B_{11}, B_{12}, B_{15}) When Used as External Address/Data Bus	I_{OH} = −1 mA	2.4		V
V_{OL5}		I_{OL} = 3 mA		0.4	V
V_{RAM}	RAM Keep-Alive Voltage	(Note 3)	2.5	V_{CC}	V
I_{OZ}	TRI-STATE® Leakage Current	V_{IN} = 0 and V_{IN} = V_{CC}		±5	μA

Note 1: I_{CC1}, I_{CC2}, I_{CC3} measured with no external drive (I_{OH} and I_{OL} = 0, I_{IH} and I_{IL} = 0). I_{CC1} is measured with $\overline{RESET}$ = V_{SS}. I_{CC3} is measured with NMI = V_{CC}. CKI driven to V_{IH1} and V_{IL1} with rise and fall times less than 10 ns.

Note 2: This is guaranteed by design and not tested.

Note 3: Test duration is 100 ms.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

20 MHz

AC Electrical Characteristics

(See Notes 1 and 4 and *Figure 1* through *Figure 5*). V_{CC} = 5V ±10%, T_A = 0°C to +70°C for HPC46064/46004, −40°C to +85°C for HPC36064/36004, −40°C to +105°C for HPC26064/26004, −55°C to +125°C for HPC16064/16004

	Symbol and Formula	Parameter and Notes	Min	Max	Units	Notes
Clocks	f_C	CKI Operating Frequency	2	20	MHz	
	$t_{C1} = 1/f_C$	CKI Clock Period	50	500	ns	
	t_{CKIH}	CKI High Time	22.5		ns	
	t_{CKIL}	CKI Low Time	22.5		ns	
	$t_C = 2/f_C$	CPU Timing Cycle	100		ns	
	$t_{WAIT} = t_C$	CPU Wait State Period	100		ns	
	t_{DC1C2R}	Delay of CK2 Rising Edge after CKI Falling Edge	0	55	ns	(Note 1)
	t_{DC1C2F}	Delay of CK2 Falling Edge after CKI Falling Edge	0	55	ns	(Note 1)
	$f_U = f_C/8$ f_{MW}	External UART Clock Input Frequency External MICROWIRE/PLUS Clock Input Frequency		2.5* 1.25	MHz MHz	
Timers	$f_{XIN} = f_C/22$ $t_{XIN} = t_C$	External Timer Input Frequency Pulse Width for Timer Inputs	100	0.91	MHz ns	
MICROWIRE/PLUS	t_{UWS}	MICROWIRE Setup Time Master Slave	 100 20		ns	
	t_{UWH}	MICROWIRE Hold Time Master Slave	 20 50		ns	
	t_{UWV}	MICROWIRE Output Valid Time Master Slave		 50 150	ns	
External Hold	$t_{SALE} = \frac{3}{4}t_C + 40$	$\overline{HLD}$ Falling Edge before ALE Rising Edge	115		ns	
	$t_{HWP} = t_C + 10$	$\overline{HLD}$ Pulse Width	110		ns	
	$t_{HAE} = t_C + 100$	$\overline{HLDA}$ Falling Edge after $\overline{HLD}$ Falling Edge		200	ns	(Note 3)
	$t_{HAD} = \frac{3}{4}t_C + 85$	$\overline{HLDA}$ Rising Edge after $\overline{HLD}$ Rising Edge		160	ns	
	$t_{BF} = \frac{1}{2}t_C + 66$	Bus Float after $\overline{HLDA}$ Falling Edge		116	ns	(Note 5)
	$t_{BE} = \frac{1}{2}t_C + 66$	Bus Enable after $\overline{HLDA}$ Rising Edge	116		ns	(Note 5)
UPI Timing	t_{UAS}	Address Setup Time to Falling Edge of $\overline{URD}$	10		ns	
	t_{UAH}	Address Hold Time from Rising Edge of $\overline{URD}$	10		ns	
	t_{RPW}	$\overline{URD}$ Pulse Width	100		ns	
	t_{OE}	$\overline{URD}$ Falling Edge to Output Data Valid	0	60	ns	
	t_{OD}	Rising Edge of $\overline{URD}$ to Output Data Invalid	5	35	ns	(Note 6)
	t_{DRDY}	$\overline{RDRDY}$ Delay from Rising Edge of $\overline{URD}$		70	ns	
	t_{WDW}	$\overline{UWR}$ Pulse Width	40		ns	
	t_{UDS}	Input Data Valid before Rising Edge of $\overline{UWR}$	10		ns	
	t_{UDH}	Input Data Hold after Rising Edge of $\overline{UWR}$	20		ns	
	t_A	$\overline{WRRDY}$ Delay from Rising Edge of $\overline{UWR}$		70	ns	

*This maximum frequency is attainable provided that this external baud clock has a duty cycle such that the high period includes two (2) falling edges of the CK2 clock.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

20 MHz (Continued)

AC Electrical Characteristics

(See Notes 1 and 4 and *Figure 1* through *Figure 5*). V_{CC} = 5V ±10%, T_A = 0°C to +70°C for HPC46064/46004, −40°C to +85°C for HPC36064/36004, −40°C to +105°C for HPC26064/26004, −55°C to +125°C for HPC16064/16004

	Symbol and Formula	Parameter and Notes	Min	Max	Units	Notes
Address Cycles	$t_{DC1ALER}$	Delay from CKI Rising Edge to ALE Rising Edge	0	35	ns	(Notes 1, 2)
	$t_{DC1ALEF}$	Delay from CKI Rising Edge to ALE Falling Edge	0	35	ns	(Notes 1, 2)
	$t_{DC2ALER} = 1/4\ t_C + 20$	Delay from CK2 Rising Edge to ALE Rising Edge		45	ns	(Note 2)
	$t_{DC2ALEF} = 1/4\ t_C + 20$	Delay from CK2 Falling Edge to ALE Falling Edge		45	ns	(Note 2)
	$t_{LL} = 1/2\ t_C - 9$	ALE Pulse Width	41		ns	
	$t_{ST} = 1/4\ t_C - 7$	Setup of Address Valid before ALE Falling Edge	18		ns	
	$t_{VP} = 1/4\ t_C - 5$	Hold of Address Valid after ALE Falling Edge	20		ns	
Read Cycles	$t_{ARR} = 1/4\ t_C - 5$	ALE Falling Edge to $\overline{RD}$ Falling Edge	20		ns	
	$t_{ACC} = t_C + WS - 55$	Data Input Valid after Address Output Valid		145	ns	(Note 6)
	$t_{RD} = 1/2\ t_C + WS - 65$	Data Input Valid after $\overline{RD}$ Falling Edge		85	ns	
	$t_{RW} = 1/2\ t_C + WS - 10$	$\overline{RD}$ Pulse Width	140		ns	
	$t_{DR} = 3/4\ t_C - 15$	Hold of Data Input Valid after $\overline{RD}$ Rising Edge	0	60	ns	
	$t_{RDA} = t_C - 15$	Bus Enable after $\overline{RD}$ Rising Edge	85		ns	
Write Cycles	$t_{ARW} = 1/2\ t_C - 5$	ALE Falling Edge to $\overline{WR}$ Falling Edge	45		ns	
	$t_{WW} = 3/4\ t_C + WS - 15$	$\overline{WR}$ Pulse Width	160		ns	
	$t_V = 1/2\ t_C + WS - 5$	Data Output Valid before $\overline{WR}$ Rising Edge	145		ns	
	$t_{HW} = 1/4\ t_C - 5$	Hold of Data Valid after $\overline{WR}$ Rising Edge	20		ns	
Ready Input	$t_{DAR} = 1/4\ t_C + WS - 50$	Falling Edge of ALE to Falling Edge of RDY		75	ns	
	$t_{RWP} = t_C$	RDY Pulse Width	100		ns	

Note: C_L = 40 pF.

Note 1: These AC characteristics are guaranteed with external clock drive on CKI having 50% duty cycle and with less than 15 pF load on CKO with rise and fall times (t_{CKIR} and t_{CKIL}) on CKI input less than 2.5 ns.

Note 2: Do not design with these parameters unless CKI is driven with an active signal. When using a passive crystal circuit, its stability is not guaranteed if either CKI or CKO is connected to any external logic other than the passive components of the crystal circuit.

Note 3: t_{HAE} is spec'd for case with $\overline{HLD}$ falling edge occurring at the latest time it can be accepted during the present CPU cycle being executed. If $\overline{HLD}$ falling edge occurs later, t_{HAE} may be as long as ($3\ t_C + 4WS + 72\ t_C + 100$) may occur depending on the following CPU instruction cycles, its wait states and ready input.

Note 4: WS (t_{WAIT}) × (number of preprogrammed wait states). Minimum and maximum values are calculated at maximum operating frequency, t_C = 20 MHz, with one wait state programmed.

Note 5: Due to emulation restrictions—actual limits will be better.

Note 6: This is guaranteed by design and not tested.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

30 MHz

AC Electrical Characteristics

(See Notes 1 and 4 and *Figure 1* through *Figure 5.*) V_{CC} = 5V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46064/46004, −40°C to +85°C for HPC36064/36004, −40°C to +105°C for HPC26064/26004, −55°C to +125°C for HPC16064/16004

	Symbol and Formula	Parameter and Notes	Min	Max	Units	Notes
Clocks	f_C	CKI Operating Frequency	2	30	MHz	
	$t_{C1} = 1/f_C$	CKI Clock Period	33	500	ns	
	t_{CKIH}	CKI High Time	15		ns	
	t_{CKIL}	CKI Low Time	16.6		ns	
	$t_C = 2/f_C$	CPU Timing Cycle	66		ns	
	$t_{WAIT} = t_C$	CPU Wait State Period	66		ns	
	t_{DC1C2R}	Delay of CK2 Rising Edge after CKI Falling Edge	0	55	ns	(Note 1)
	t_{DC1C2F}	Delay of CK2 Falling Edge after CKI Falling Edge	0	55	ns	(Note 1)
	$f_U = f_C/8$ f_{MW}	External UART Clock Input Frequency External MICROWIRE/PLUS Clock Input Frequency		3.75* 1.875	MHz MHz	
Timers	$f_{XIN} = f_C/22$ $t_{XIN} = t_C$	External Timer Input Frequency Pulse Width for Timer Inputs	 66	1.36	MHz ns	
MICROWIRE/PLUS	t_{UWS}	MICROWIRE Setup Time Master Slave	 100 20		 ns	
	t_{UWH}	MICROWIRE Hold Time Master Slave	 20 50		 ns	
	t_{UWV}	MICROWIRE Output Valid Time Master Slave		 50 150	 ns	
External Hold	$t_{SALE} = \frac{3}{4}t_C + 40$	$\overline{HLD}$ Falling Edge before ALE Rising Edge	90		ns	
	$t_{HWP} = t_C + 10$	$\overline{HLD}$ Pulse Width	76		ns	
	$t_{HAE} = t_C + 100$	$\overline{HLDA}$ Falling Edge after $\overline{HLD}$ Falling Edge		151	ns	(Note 3)
	$t_{HAD} = \frac{3}{4}t_C + 85$	$\overline{HLDA}$ Rising Edge after $\overline{HLD}$ Rising Edge		135	ns	
	$t_{BF} = \frac{1}{2}t_C + 66$	Bus Float after $\overline{HLDA}$ Falling Edge		99	ns	(Note 5)
	$t_{BE} = \frac{1}{2}t_C + 66$	Bus Enable after $\overline{HLDA}$ Rising Edge	99		ns	(Note 5)
UPI Timing	t_{UAS}	Address Setup Time to Falling Edge of $\overline{URD}$	10		ns	
	t_{UAH}	Address Hold Time from Rising Edge of $\overline{URD}$	10		ns	
	t_{RPW}	$\overline{URD}$ Pulse Width	100		ns	
	t_{OE}	$\overline{URD}$ Falling Edge to Output Data Valid	0	60	ns	
	t_{OD}	Rising Edge of $\overline{URD}$ to Output Data Invalid	5	35	ns	(Note 6)
	t_{DRDY}	$\overline{RDRDY}$ Delay from Rising Edge of $\overline{URD}$		70	ns	
	t_{WDW}	$\overline{UWR}$ Pulse Width	40		ns	
	t_{UDS}	Input Data Valid before Rising Edge of $\overline{UWR}$	10		ns	
	t_{UDH}	Input Data Hold after Rising Edge of $\overline{UWR}$	20		ns	
	t_A	$\overline{WRRDY}$ Delay from Rising Edge of $\overline{UWR}$		70	ns	

*This maximum frequency is attainable provided that this external baud clock has a duty cycle such that the high period includes two (2) falling edges of the CK2 clock.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

30 MHz (Continued)

AC Electrical Characteristics

(See Notes 1 and 4 and *Figure 1* through *Figure 5*.) V_{CC} = 5V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46064/46004, −40°C to +85°C for HPC36064/36004, −40°C to +105°C for HPC26064/26004, −55°C to +125°C for HPC16064/16004

	Symbol and Formula	Parameter and Notes	Min	Max	Units	Notes
Address Cycles	$t_{DC1ALER}$	Delay from CKI Rising Edge to ALE Rising Edge	0	35	ns	(Notes 1, 2)
	$t_{DC1ALEF}$	Delay from CKI Rising Edge to ALE Falling Edge	0	35	ns	(Notes 1, 2)
	$t_{DC2ALER} = 1/4\ t_C + 20$	Delay from CK2 Rising Edge to ALE Rising Edge		37	ns	(Note 2)
	$t_{DC2ALEF} = 1/4\ t_C + 20$	Delay from CK2 Falling Edge to ALE Falling Edge		37	ns	(Note 2)
	$t_{LL} = 1/2\ t_C - 9$	ALE Pulse Width	24		ns	
	$t_{ST} = 1/4\ t_C - 7$	Setup of Address Valid before ALE Falling Edge	9		ns	
	$t_{VP} = 1/4\ t_C - 5$	Hold of Address Valid after ALE Falling Edge	11		ns	
Read Cycles	$t_{ARR} = 1/4\ t_C - 5$	ALE Falling Edge to $\overline{RD}$ Falling Edge	11		ns	
	$t_{ACC} = t_C + WS - 32$	Data Input Valid after Address Output Valid		100	ns	(Note 6)
	$t_{RD} = 1/2\ t_C + WS - 39$	Data Input Valid after $\overline{RD}$ Falling Edge		60	ns	
	$t_{RW} = 1/2\ t_C + WS - 14$	$\overline{RD}$ Pulse Width	85		ns	
	$t_{DR} = 3/4\ t_C - 15$	Hold of Data Input Valid after $\overline{RD}$ Rising Edge	0	35	ns	
	$t_{RDA} = t_C - 15$	Bus Enable after $\overline{RD}$ Rising Edge	51		ns	
Write Cycles	$t_{ARW} = 1/2\ t_C - 5$	ALE Falling Edge to $\overline{WR}$ Falling Edge	28		ns	
	$t_{WW} = 3/4\ t_C + WS - 15$	$\overline{WR}$ Pulse Width	101		ns	
	$t_V = 1/2\ t_C + WS - 5$	Data Output Valid before $\overline{WR}$ Rising Edge	94		ns	
	$t_{HW} = 1/4\ t_C - 10$	Hold of Data Valid after $\overline{WR}$ Rising Edge	7		ns	
Ready Input	$t_{DAR} = 1/4\ t_C + WS - 50$	Falling Edge of ALE to Falling Edge of RDY		33	ns	
	$t_{RWP} = t_C$	RDY Pulse Width	66		ns	

Note: C_L = 40 pF.

Note 1: These AC characteristics are guaranteed with external clock drive on CKI having 50% duty cycle and with less than 15 pF load on CKO with rise and fall times (t_{CKIR} and t_{CKIL}) on CKI input less than 2.5 ns.

Note 2: Do not design with these parameters unless CKI is driven with an active signal. When using a passive crystal circuit, its stability is not guaranteed if either CKI or CKO is connected to any external logic other than the passive components of the crystal circuit.

Note 3: t_{HAE} is spec'd for case with $\overline{HLD}$ falling edge occurring at the latest time it can be accepted during the present CPU cycle being executed. If $\overline{HLD}$ falling edge occurs later, t_{HAE} may be as long as (3 t_C + 4WS + 72 t_C + 100) may occur depending on the following CPU instruction cycles, its wait states and ready input.

Note 4: WS (t_{WAIT}) × (number of preprogrammed wait states). Minimum and maximum values are calculated at maximum operating frequency, t_C = 20 MHz, with one wait state programmed.

Note 5: Due to emulation restrictions—actual limits will be better.

Note 6: This is guaranteed by design and not tested.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

CKI Input Signal Characteristics

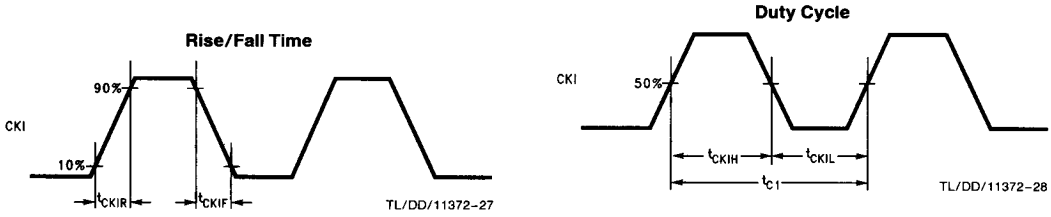


FIGURE 1. CKI Input Signal

Note: AC testing inputs are driven at V_{IH} for a logic "1" and V_{IL} for a logic "0". Output timing measurements are made at V_{OH} for a logic "1" and V_{OL} for a logic "0".

FIGURE 2. Input and Output for AC Tests

Timing Waveforms

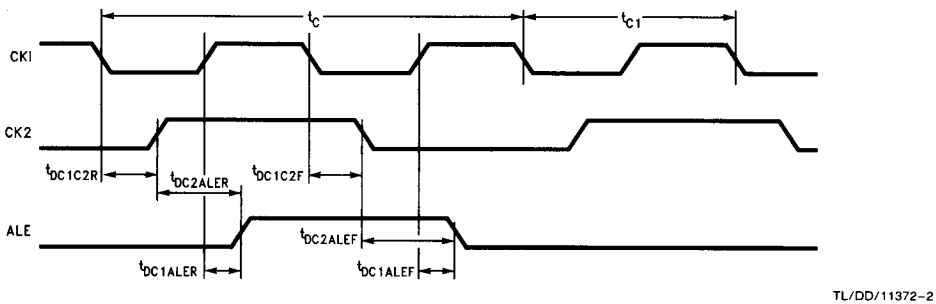
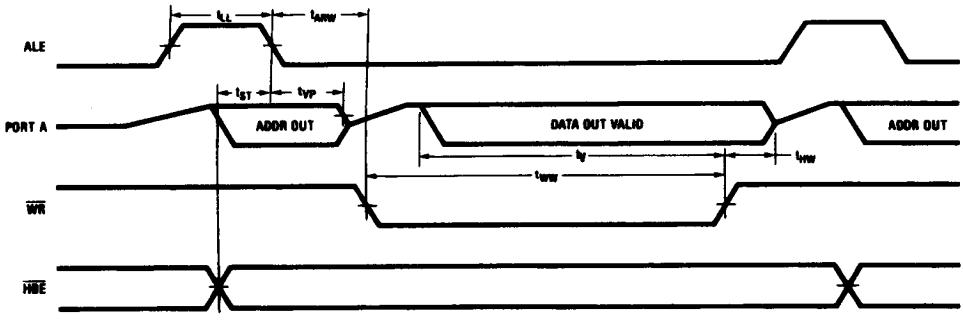


FIGURE 3. CKI, CK2, ALE Timing Diagram

4

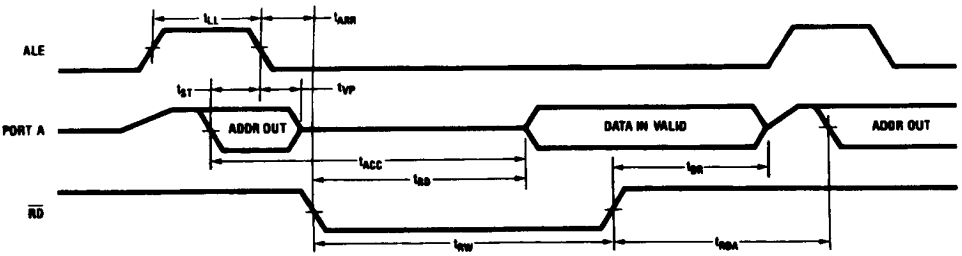
Powered by ICminer.com Electronic-Library Service CopyRight 2003

Timing Waveforms (Continued)



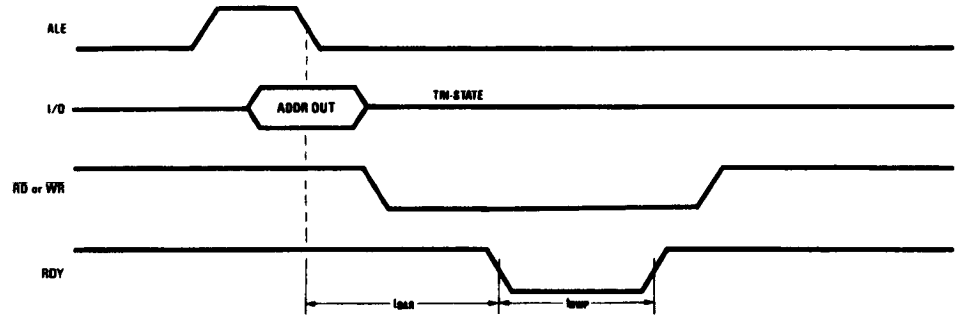
TL/DD/11372–3

FIGURE 4. Write Cycle



TL/DD/11372–4

FIGURE 5. Read Cycle

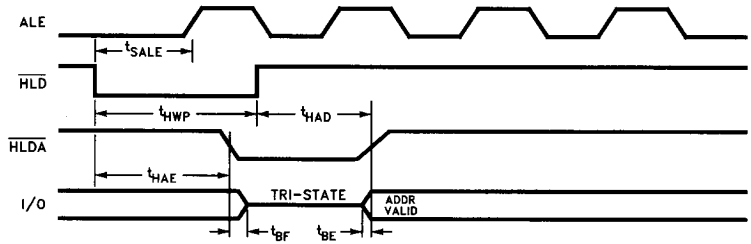


TL/DD/11372–5

FIGURE 6. Ready Mode Timing

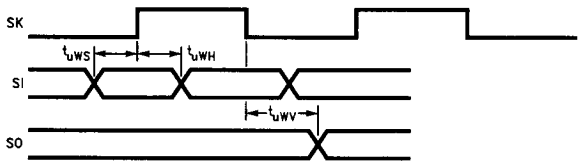
Powered by ICminer.com Electronic-Library Service CopyRight 2003

Timing Waveforms (Continued)



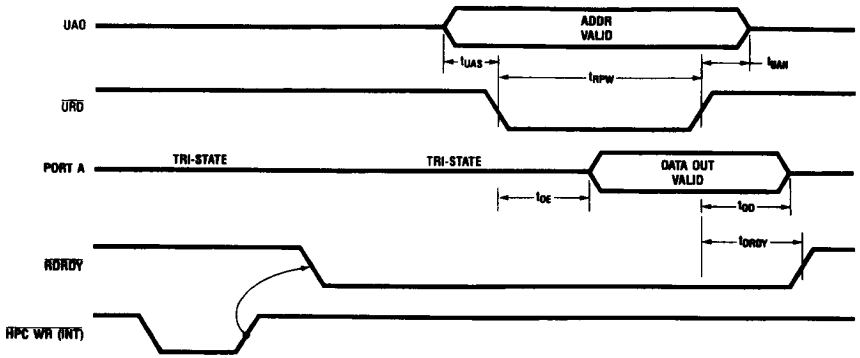
TL/DD/11372-6

FIGURE 7. Hold Mode Timing



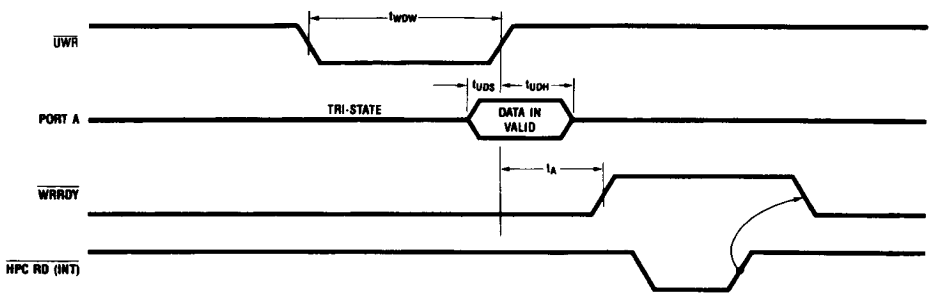
TL/DD/11372-29

FIGURE 8. MICROWIRE Setup/Hold Timing



TL/DD/11372-7

FIGURE 9. UPI Read Timing



TL/DD/11372-8

FIGURE 10. UPI Write Timing

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Pin Descriptions

The HPC46064 is available only in 68-pin PLCC, LDCC, PGA, and 80-pin PQFP packages.

I/O PORTS

Port A is a 16-bit bidirectional I/O port with a data direction register to enable each separate pin to be individually defined as an input or output. When accessing external memory, port A is used as the multiplexed address/data bus.

Port B is a 16-bit port with 12 bits of bidirectional I/O similar in structure to Port A. Pins B10, B11, B12 and B15 are general purpose outputs only in this mode. Port B may also be configured via a 16-bit function register BFUN to individually allow each pin to have an alternate function.

B0:	TDX	UART Data Output
B1:		
B2:	CKX	UART Clock (Input or Output)
B3:	T2IO	Timer2 I/O Pin
B4:	T3IO	Timer3 I/O Pin
B5:	SO	MICROWIRE/PLUS Output
B6:	SK	MICROWIRE/PLUS Clock (Input or Output)
B7:	$\overline{\text{HLDA}}$	Hold Acknowledge Output
B8:	TS0	Timer Synchronous Output
B9:	TS1	Timer Synchronous Output
B10:	UA0	Address 0 Input for UPI Mode
B11:	$\overline{\text{WRRDY}}$	Write Ready Output for UPI Mode
B12:		
B13:	TS2	Timer Synchronous Output
B14:	TS3	Timer Synchronous Output
B15:	$\overline{\text{RDRDY}}$	Read Ready Output for UPI Mode

When accessing external memory, four bits of port B are used as follows:

B10:	ALE	Address Latch Enable Output
B11:	$\overline{\text{WR}}$	Write Output
B12:	$\overline{\text{HBE}}$	High Byte Enable Output/Input (sampled at reset)
B15:	$\overline{\text{RD}}$	Read Output

Port I is an 8-bit input port that can be read as general purpose inputs and is also used for the following functions:

I0:		
I1:	NMI	Nonmaskable Interrupt Input
I2:	INT2	Maskable Interrupt/Input Capture/$\overline{\text{URD}}$
I3:	INT3	Maskable Interrupt/Input Capture/$\overline{\text{UWR}}$
I4:	INT4	Maskable Interrupt/Input Capture
I5:	SI	MICROWIRE/PLUS Data Input
I6:	RDX	UART Data Input
I7:		

Port D is an 8-bit input port that can be used as general purpose digital inputs.

Port P is a 4-bit output port that can be used as general purpose data, or selected to be controlled by timers 4 through 7 in order to generate frequency, duty cycle and pulse width modulated outputs.

POWER SUPPLY PINS

V_{CC1} and V_{CC2}	Positive Power Supply
GND	Ground for On-Chip Logic
DGND	Ground for Output Buffers

Note: There are two electrically connected V_{CC} pins on the chip, GND and DGND are electrically isolated. Both V_{CC} pins and both ground pins must be used.

CLOCK PINS

CKI	The Chip System Clock Input
CKO	The Chip System Clock Output (inversion of CKI)

Pins CKI and CKO are usually connected across an external crystal.

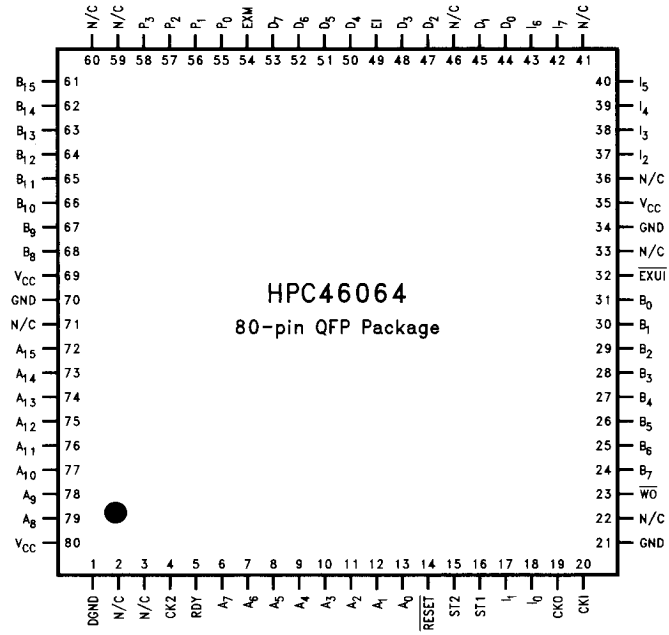
CK2	Clock Output (CKI divided by 2)

OTHER PINS

$\overline{\text{WO}}$	This is an active low open drain output that signals an illegal situation has been detected by the WATCHDOG logic.
ST1	Bus Cycle Status Output: indicates first opcode fetch.
ST2	Bus Cycle Status Output: indicates machine states (skip, interrupt and first instruction cycle).
$\overline{\text{RESET}}$	Active low input that forces the chip to restart and sets the ports in a TRI-STATE mode.
RDY/$\overline{\text{HLD}}$	Selected by a software bit. It's either a READY input to extend the bus cycle for slower memories, or a HOLD request input to put the bus in a high impedance state for DMA purposes.
N/C	(No connection) do not connect anything to this pin.
EXM	External memory enable (active high) disables internal ROM and maps it to external memory.
EI	External interrupt with vector address FFF1:FFF0. (Rising/falling edge or high/low level sensitive). Alternately can be configured as 4th input capture.
$\overline{\text{EXUI}}$	External active low interrupt which is internally OR'ed with the UART interrupt with vector address FFF3:FFF2.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Connection Diagrams

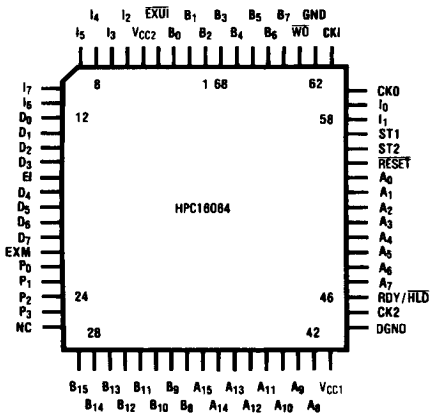


TL/DD/11372-32

Top View

Order Number HPC46064XXX/F20, HPC46064XXX/F30, HPC46004VF20 or HPC46004VF30
See NS Package Number VF80B

Plastic and Ceramic Leaded Chip Carriers



TL/DD/11372-33

Top View

Order Number HPC16064XXX/L20, HPC16064XXX/L30, HPC16004EL20 or HPC16004EL30
See NS Package Number EL68A

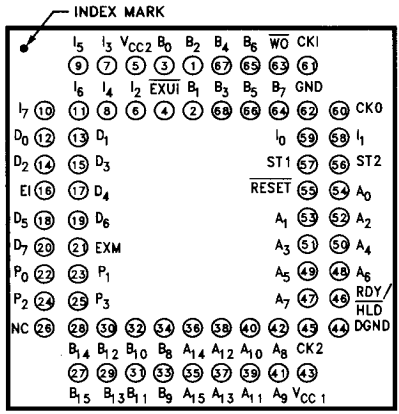
Order Number HPC16064XXX/V20, HPC26064XXX/V20, HPC36064XXX/V20, HPC46064XXX/V20, HPC16064XXX/V30, HPC26064XXX/V30, HPC360864XXX/V30, HPC16004V20, HPC26004V20, HCP36004V20, HPC46003V20, HPC16004V30, HPV26004V30, HPC36004V30 or HPC46004V30
See NS Package Number V68A

Note: XXX designates the unique ROM cocde of a masked device.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Connection Diagrams (Continued)

Pin Grid Array Pinout



TL/DD/11372–34

Top View
(looking down on component side of PC Board)

Order Number HPC16064XXX/U20, HPC16064XXX/U30, HPC16004U20 or HPC16004U30
See NS Package Number U68A

Note: XXX designates the unique ROM code of a masked device.

Ports A & B

The highly flexible A and B ports are similarly structured. The Port A (see *Figure 11*) consists of a data register and a direction register. Port B (see *Figures 12, 13* and *14*) has an alternate function register in addition to the data and direction registers. All the control registers are read/write registers.

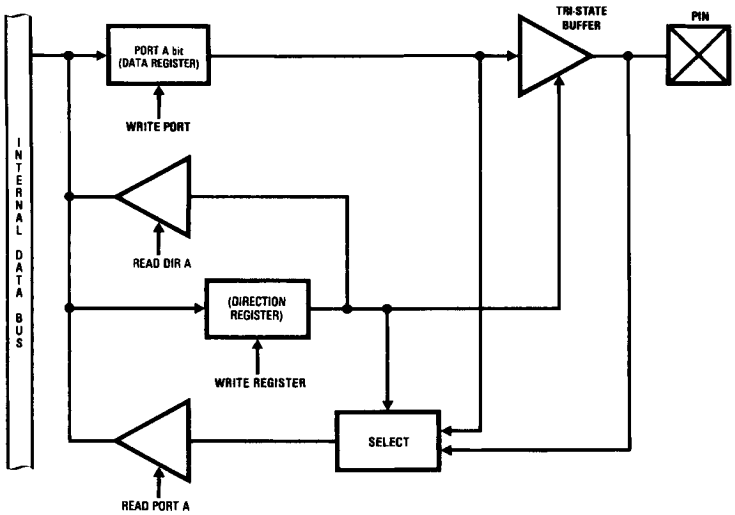
The associated direction registers allow the port pins to be individually programmed as inputs or outputs. Port pins selected as inputs, are placed in a TRI-STATE mode by resetting corresponding bits in the direction register.

A write operation to a port pin configured as an input causes the value to be written into the data register, a read operation returns the value of the pin. Writing to port pins configured as outputs causes the pins to have the same value, reading the pins returns the value of the data register.

Primary and secondary functions are multiplexed onto Port B through the alternate function register (BFUN). The secondary functions are enabled by setting the corresponding bits in the BFUN register.

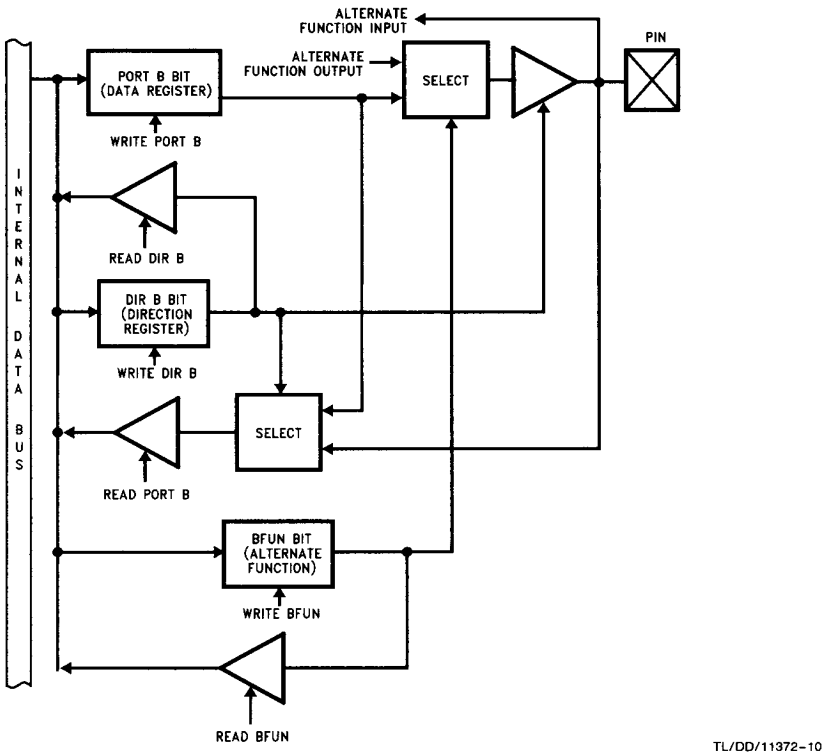
Powered by ICminer.com Electronic-Library Service CopyRight 2003

Ports A & B (Continued)



TL/DD/11372-9

FIGURE 11. Port A: I/O Structure



TL/DD/11372-10

FIGURE 12. Structure of Port B Pins B0, B1, B2, B5, B6 and B7 (Typical Pins)

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Ports A & B (Continued)

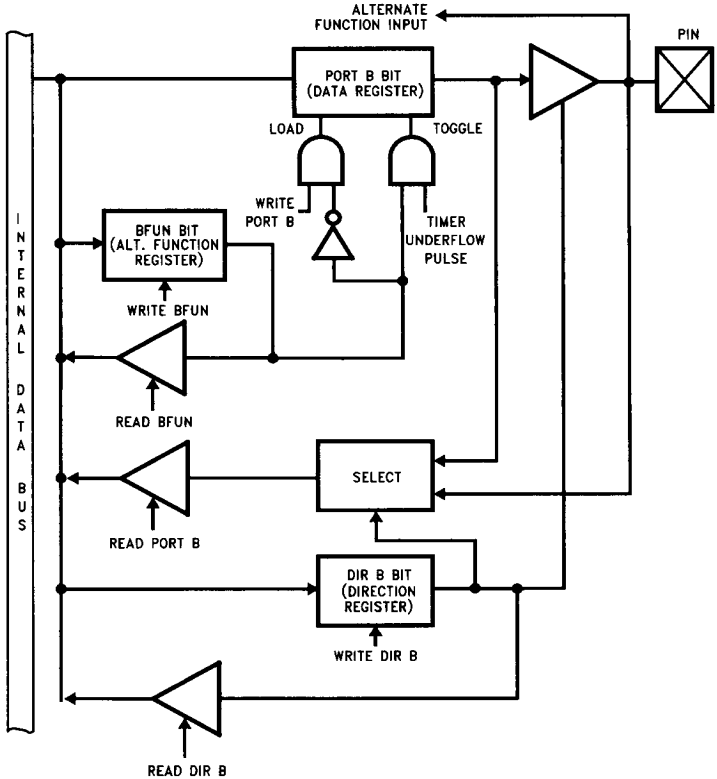


FIGURE 13. Structure of Port B Pins B3, B4, B8, B9, B13 and B14 (Timer Synchronous Pins)

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Ports A & B (Continued)

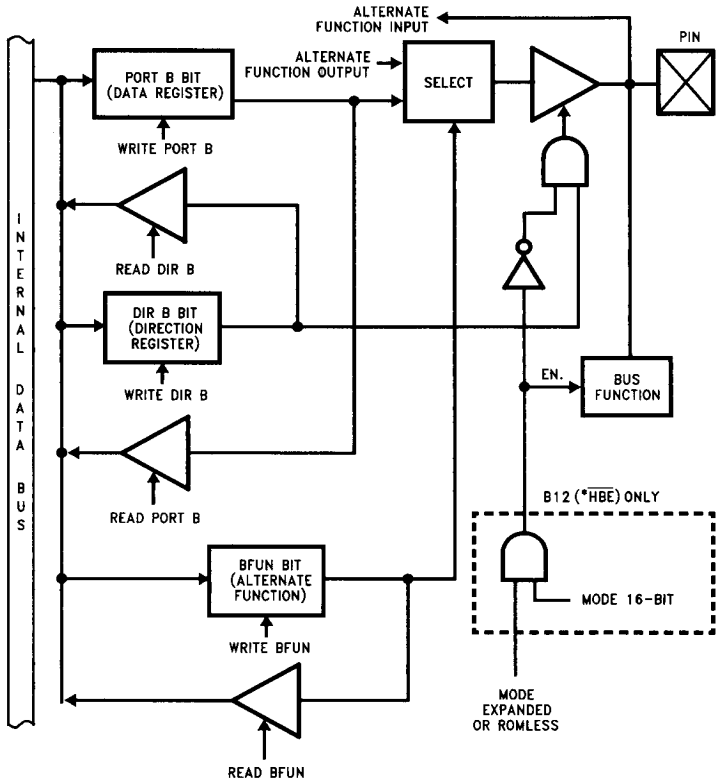


FIGURE 14. Structure of Port B Pins B10, B11, B12 and B15 (Pins with Bus Control Roles)

Operating Modes

To offer the user a variety of I/O and expanded memory options, the HPC46064 and HPC46004 have four operating modes. The ROMless HPC46004 has one mode of operation. The various modes of operation are determined by the state of both the EXM pin and the EA bit in the PSW register. The state of the EXM pin determines whether on-chip ROM will be accessed or external memory will be accessed within the address range of the on-chip ROM. The on-chip ROM range of the HPC46064 is C000 to FFFF (16k bytes). The HPC46004 has no on-chip ROM and is intended for use with external memory for program storage. A logic "0" state on the EXM pin will cause the HPC device to address on-chip ROM when the Program Counter (PC) contains addresses within the on-chip ROM address range. A logic "1" state on the EXM pin will cause the HPC device to address memory that is external to the HPC when the PC contains on-chip ROM addresses. The EXM pin should always be pulled high (logic "1") on the HPC46004 because no on-chip ROM is available. The function of the EA bit is to determine the legal addressing range of the HPC device. A logic "0" state in the EA bit of the PSW register does two things—addresses are limited to the on-chip ROM range and on-chip RAM and Register range, and the "illegal address detection" feature of the WATCHDOG logic is engaged. A logic "1" in the EA bit enables accesses to be made anywhere within the 64k byte address range and the "illegal address detection" feature of the WATCHDOG logic is disabled. The EA bit should be set to "1" by software when using the HPC46004 to disable the "illegal address detection" feature of WATCHDOG.

All HPC devices can be used with external memory. External memory may be any combination of RAM and ROM. Both 8-bit and 16-bit external data bus modes are available. Upon entering an operating mode in which external memory is used, port A becomes the Address/Data bus. Four pins of port B become the control lines ALE, $\overline{RD}$, $\overline{WR}$ and $\overline{HBE}$. The High Byte Enable pin ($\overline{HBE}$) is used in 16-bit mode to select high order memory bytes. The $\overline{RD}$ and $\overline{WR}$ signals are only generated if the selected address is off-chip. The 8-bit mode is selected by pulling $\overline{HBE}$ high at reset. If $\overline{HBE}$ is left floating or connected to a memory device chip select at reset, the 16-bit mode is entered. The following sections describe the operating modes of the HPC46064 and HPC46004.

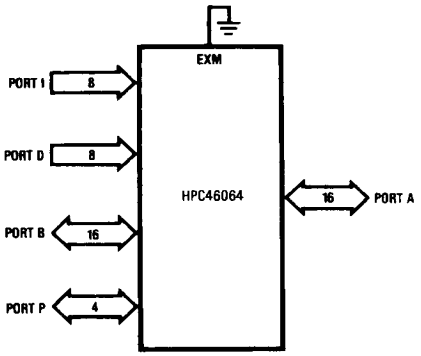
Note: The HPC devices use 16-bit words for stack memory. Therefore, when using the 8-bit mode, User's Stack must be in internal RAM.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

HPC46064 Operating Modes

SINGLE CHIP NORMAL MODE

In this mode, the HPC46064 functions as a self-contained microcomputer (see *Figure 15*) with all memory (RAM and ROM) on-chip. It can address internal memory only, consisting of 16k bytes of ROM (C000 to FFFF) and 512 bytes of on-chip RAM and Registers (0000 to 02FF). The "illegal address detection" feature of the WATCHDOG is enabled in the Single-Chip Normal mode and a WATCHDOG Output ($\overline{WO}$) will occur if an attempt is made to access addresses that are outside of the on-chip ROM and RAM range of the device. Ports A and B are used for I/O functions and not for addressing external memory. The EXM pin and the EA bit of the PSW register must both be logic "0" to enter the Single-Chip Normal mode.



TL/DD/11372-13

FIGURE 15. Single-Chip Mode

EXPANDED NORMAL MODE

The Expanded Normal mode of operation enables the HPC46064 to address external memory in addition to the on-chip ROM and RAM (see Table I). WATCHDOG illegal address detection is disabled and memory accesses may be made anywhere in the 64k byte address range without triggering an illegal address condition. The Expanded Normal mode is entered with the EXM pin pulled low (logic "0") and setting the EA bit in the PSW register to "1".

SINGLE-CHIP ROMLESS MODE

In this mode, the on-chip mask programmed ROM of the HPC46064 is not used. The address space corresponding to the on-chip ROM is mapped into external memory so 16k of external memory may be used with the HPC46064 (see Table I). The WATCHDOG circuitry detects illegal addresses (addresses not within the on-chip ROM and RAM range). The Single-Chip ROMless mode is entered when the EXM pin is pulled high (logic "1") and the EA bit is logic "0".

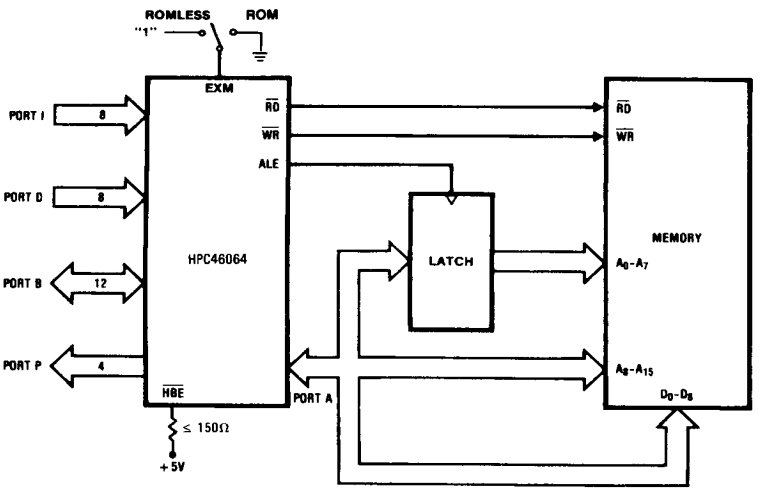
TABLE I. HPC46064 Operating Modes

Operating Mode	EXM Pin	EA Bit	Memory Configuration
Single-Chip Normal	0	0	C000:FFFF on-chip
Expanded Normal	0	1	C000:FFFF on-chip 0300:BFFF off-chip
Single-Chip ROMless	1	0	C000:FFFF off-chip
Expanded ROMless	1	1	0300:FFFF off-chip

Note: In all operating modes, the on-chip RAM and Registers (0000:02FF) may be accessed.

EXPANDED ROMLESS MODE

This mode of operation is similar to Single-Chip ROMless mode in that no on-chip ROM is used, however, a full 64k bytes of external memory may be used. The "illegal address detection" feature of WATCHDOG is disabled. The EXM pin must be pulled high (logic "1") and the EA bit in the PSW register set to "1" to enter this mode.

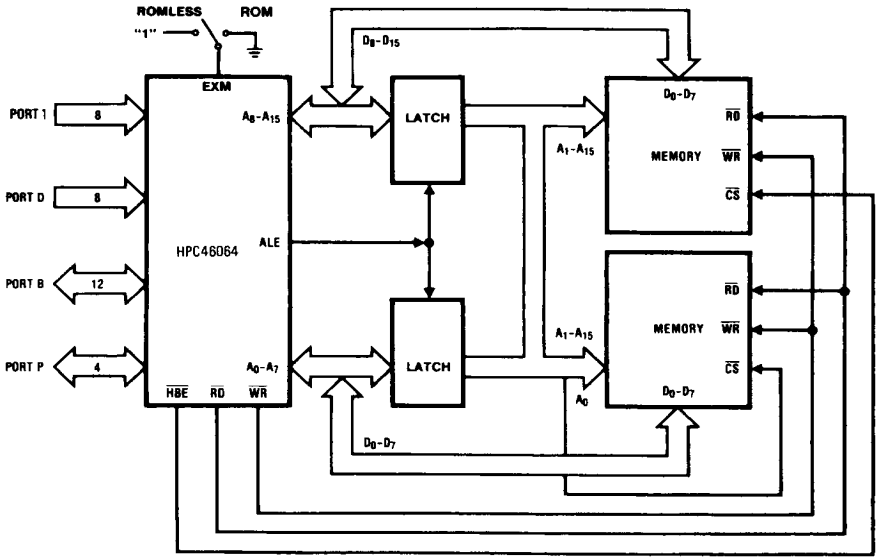


TL/DD/11372-14

FIGURE 16. 8-Bit External Memory

Powered by ICminer.com Electronic-Library Service CopyRight 2003

HPC46064 Operating Modes (Continued)



TL/DD/11372-15

FIGURE 17. 16-Bit External Memory

HPC46004 Operating Modes

EXPANDED ROMLESS MODE

Because the HPC46004 has no on-chip ROM, it has only one mode of operation, the Expanded ROMless Mode. The EXM pin must be pulled high (logic "1") on power up, the EA bit in the PSW register should be set to a "1". The HPC46004 is a ROMless device and is intended for use with external memory. The external memory may be any combination of ROM and RAM. Up to 64k bytes of external memory may be accessed. It is necessary to vector on reset to an address between C000 and FFFF, therefore the user should have external memory at these addresses. The EA bit in the PSW register must immediately be set to "1" at the beginning of the user's program to disable illegal address detection in the WATCHDOG logic.

TABLE II. HPC46004 Operating Modes

Operating Mode	EXM Pin	EA Bit	Memory Configuration
Expanded ROMless	1	1	0300:FFFF off-chip

Note: The on-chip RAM and Registers (0000:02FF) of the HPC46004 may be accessed at all times.

Wait States

The internal ROM can be accessed at the maximum operating frequency with one wait state. With 0 wait states, internal ROM accesses are limited to $\frac{2}{3}$ f_C max. The HPC46064 provides four software selectable Wait States that allow access to slower memories. The Wait States are selected by the state of two bits in the PSW register. Additionally, the RDY input may be used to extend the instruction cycle, allowing the user to interface with slow memories and peripherals.

Power Save Modes

Two power saving modes are available on the HPC46064: HALT and IDLE. In the HALT mode, all processor activities are stopped. In the IDLE mode, the on-board oscillator and timer T0 are active but all other processor activities are stopped. In either mode, all on-board RAM, registers and I/O are unaffected.

HALT MODE

The HPC46064 is placed in the HALT mode under software control by setting bits in the PSW. All processor activities, including the clock and timers, are stopped. In the HALT mode, power requirements for the HPC46064 are minimal and the applied voltage (V_{CC}) may be decreased without altering the state of the machine. There are two ways of exiting the HALT mode: via the $\overline{\text{RESET}}$ or the NMI. The $\overline{\text{RESET}}$ input reinitializes the processor. Use of the NMI input will generate a vectored interrupt and resume operation from that point with no initialization. The HALT mode can be enabled or disabled by means of a control register HALT enable. To prevent accidental use of the HALT mode the HALT enable register can be modified only once.

IDLE MODE

The HPC46064 is placed in the IDLE mode through the PSW. In this mode, all processor activity, except the on-board oscillator and Timer T0, is stopped. As with the HALT mode, the processor is returned to full operation by the $\overline{\text{RESET}}$ or NMI inputs, but without waiting for oscillator stabilization. A timer T0 overflow will also cause the HPC46064 to resume normal operation.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

HPC46064 Interrupts

Complex interrupt handling is easily accomplished by the HPC46064's vectored interrupt scheme. There are eight possible interrupt sources as shown in Table III.

TABLE III. Interrupts

Vector Address	Interrupt Source	Arbitration Ranking
FFFF:FFFE	$\overline{\text{RESET}}$	0
FFFD:FFFC	Nonmaskable external on rising edge of I1 pin	1
FFFB:FFFA	External interrupt on I2 pin	2
FFF9:FFF8	External interrupt on I3 pin	3
FFF7:FFF6	External interrupt on I4 pin	4
FFF5:FFF4	Overflow on internal timers	5
FFF3:FFF2	Internal on the UART transmit/receive complete	6
FFF1:FFF0	External interrupt on EI pin	7

Interrupt Arbitration

The HPC46064 contains arbitration logic to determine which interrupt will be serviced first if two or more interrupts occur simultaneously. The arbitration ranking is given in Table III. The interrupt on Reset has the highest rank and is serviced first.

Interrupt Processing

Interrupts are serviced after the current instruction is completed except for the $\overline{\text{RESET}}$, which is serviced immediately. $\overline{\text{RESET}}$ and $\overline{\text{EXUI}}$ are level-LOW-sensitive interrupts and EI is programmable for edge-(RISING or FALLING) or level-(HIGH or LOW) sensitivity. All other interrupts are edge-sensitive. NMI is positive-edge sensitive. The external interrupts on I2, I3 and I4 can be software selected to be rising or falling edge. External interrupt ($\overline{\text{EXUI}}$) is shared with the on-board UART. The $\overline{\text{EXUI}}$ interrupt is level-LOW-sensitive. To select this interrupt, disable the ERI and ETI UART interrupts by resetting these enable bits in the ENUI register. To select the on-board UART interrupt, leave this pin floating.

Interrupt Control Registers

The HPC46064 allows the various interrupt sources and conditions to be programmed. This is done through the various control registers. A brief description of the different control registers is given below.

INTERRUPT ENABLE REGISTER (ENIR)

$\overline{\text{RESET}}$ and the External Interrupt on I1 are non-maskable interrupts. The other interrupts can be individually enabled or disabled. Additionally, a Global Interrupt Enable Bit in the ENIR Register allows the Maskable interrupts to be collectively enabled or disabled. Thus, in order for a particular interrupt to request service, both the individual enable bit and the Global Interrupt bit (GIE) have to be set.

INTERRUPT PENDING REGISTER (IRPD)

The IRPD register contains a bit allocated for each interrupt vector. The occurrence of specified interrupt trigger conditions causes the appropriate bit to be set. There is no indication of the order in which the interrupts have been received. The bits are set independently of the fact that the interrupts may be disabled. IRPD is a Read/Write register. The bits corresponding to the maskable, external interrupts are normally cleared by the HPC46064 after servicing the interrupts.

For the interrupts from the on-board peripherals, the user has the responsibility of resetting the interrupt pending flags through software.

The NMI bit is read only and I2, I3, and I4 are designed as to only allow a zero to be written to the pending bit (writing a one has no affect). A LOAD IMMEDIATE instruction is to be the only instruction used to clear a bit or bits in the IRPD register. This allows a mask to be used, thus ensuring that the other pending bits are not affected.

INTERRUPT CONDITION REGISTER (IRCD)

Three bits of the register select the input polarity of the external interrupt on I2, I3, and I4.

Servicing the Interrupts

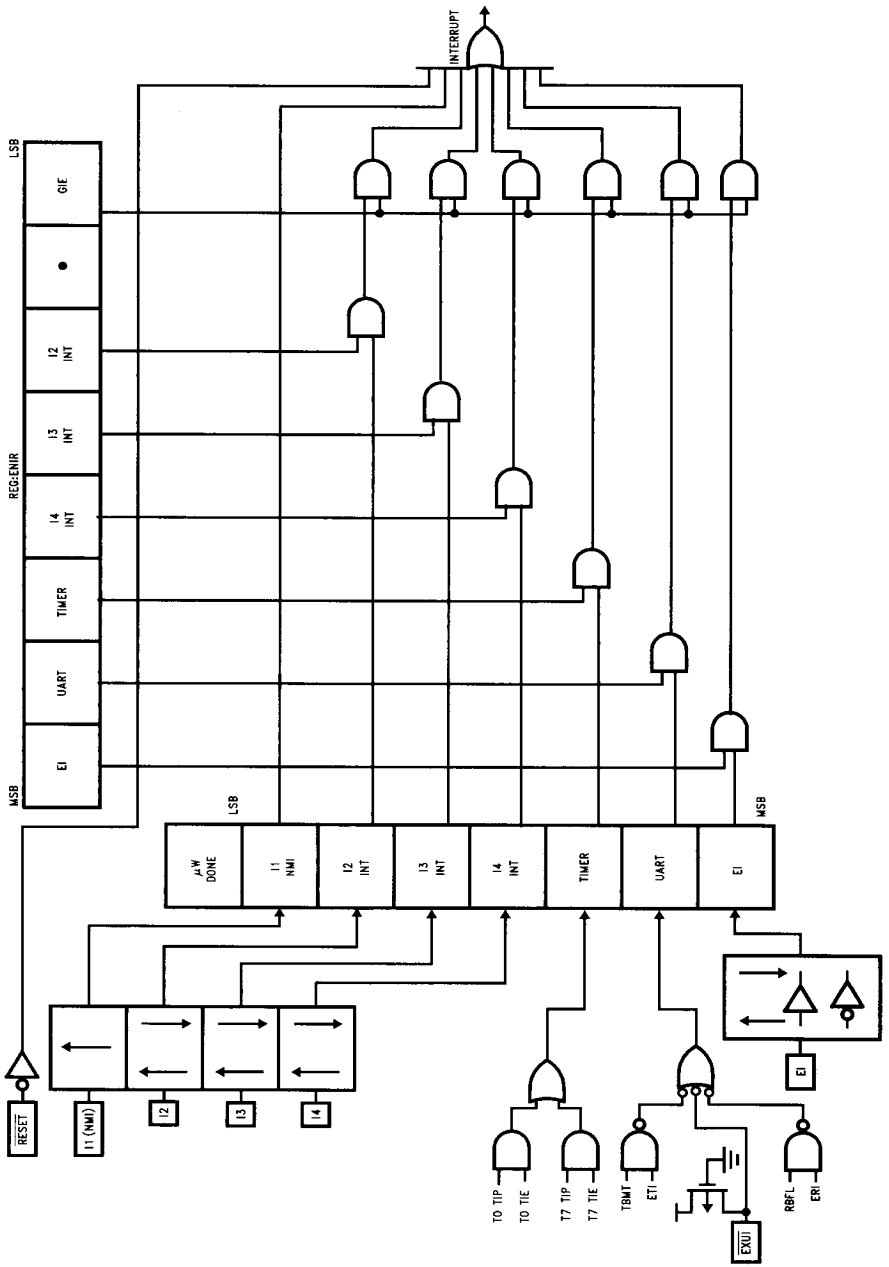
The Interrupt, once acknowledged, pushes the program counter (PC) onto the stack thus incrementing the stack pointer (SP) twice. The Global Interrupt Enable bit (GIE) is copied into the CGIE bit of the PSW register; it is then reset, thus disabling further interrupts. The program counter is loaded with the contents of the memory at the vector address and the processor resumes operation at this point. At the end of the interrupt service routine, the user does a RETI instruction to pop the stack and re-enable interrupts if the CGIE bit is set, or RET to just pop the stack if the CGIE bit is clear, and then returns to the main program. The GIE bit can be set in the interrupt service routine to nest interrupts if desired. *Figure 18* shows the Interrupt Enable Logic.

Reset

The $\overline{\text{RESET}}$ input initializes the processor and sets ports A and B in the TRI-STATE condition and Port P in the LOW state. $\overline{\text{RESET}}$ is an active-low Schmitt trigger input. The processor vectors to FFFF:FFFE and resumes operation at the address contained at that memory location (which must correspond to an on board location). The Reset vector address must be between C000 and FFFF when using the HPC46004.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Servicing the Interrupts (Continued)



TL/DD/11372–16

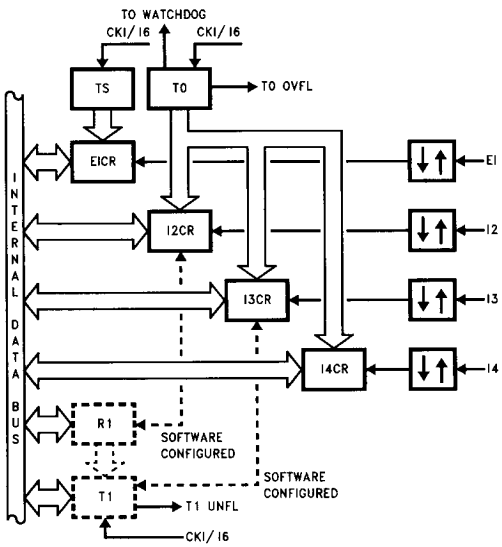
FIGURE 18. Block Diagram of Interrupt Logic

4

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Timer Overview

The HPC46064 contains a powerful set of flexible timers enabling the HPC46064 to perform extensive timer functions not usually associated with microcontrollers. The HPC46064 contains nine 16-bit timers. Timer T0 is a free-running timer, counting up at a fixed CKI/16 (Clock Input/16) rate. It is used for WATCHDOG logic, high speed event capture, and to exit from the IDLE mode. Consequently, it cannot be stopped or written to under software control. Timer T0 permits precise measurements by means of the capture registers I2CR, I3CR, and I4CR. A control bit in the register TMMODE configures timer T1 and its associated register R1 as capture registers I3CR and I2CR. The capture registers I2CR, I3CR, and I4CR respectively, record the value of timer T0 when specific events occur on the interrupt pins I2, I3, and I4. The control register IRCD programs the capture registers to trigger on either a rising edge or a falling edge of its respective input. The specified edge can also be programmed to generate an interrupt (see *Figure 19*).



TL/DD/11372–17

FIGURE 19. Timers T0, T1 and T8 with Four Input Capture Registers

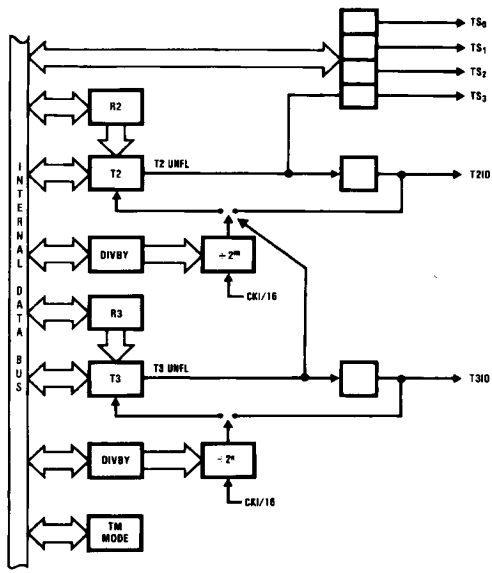
The HPC46064 provides an additional 16-bit free running timer, T8, with associated input capture register EICR (External Interrupt Capture Register) and Configuration Register, EICON. EICON is used to select the mode and edge of the EI pin. EICR is a 16-bit capture register which records the value of T8 (which is identical to T0) when a specific event occurs on the EI pin.

The timers T2 and T3 have selectable clock rates. The clock input to these two timers may be selected from the following two sources: an external pin, or derived internally by dividing the clock input. Timer T2 has additional capability of being clocked by the timer T3 underflow. This allows the user to cascade timers T3 and T2 into a 32-bit timer/counter. The control register DIVBY programs the clock input to timers T2 and T3 (see *Figure 20*).

The timers T1 through T7 in conjunction with their registers form Timer-Register pairs. The registers hold the pulse duration values. All the Timer-Register pairs can be read from or written to. Each timer can be started or stopped under software control. Once enabled, the timers count down, and upon underflow, the contents of its associated register are automatically loaded into the timer.

SYNCHRONOUS OUTPUTS

The flexible timer structure of the HPC46064 simplifies pulse generation and measurement. There are four synchronous timer outputs (TS0 through TS3) that work in conjunction with the timer T2. The synchronous timer outputs can be used either as regular outputs or individually programmed to toggle on timer T2 underflows (see *Figure 20*).



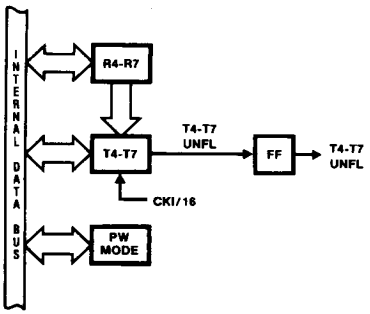
TL/DD/11372–18

FIGURE 20. Timers T2–T3 Block

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Timer Overview (Continued)

Timer/register pairs 4–7 form four identical units which can generate synchronous outputs on port P (see *Figure 21*). Maximum output frequency for any timer output can be obtained by setting timer/register pair to zero. This then will produce an output frequency equal to ½ the frequency of the source used for clocking the timer.



TL/DD/11372–19

FIGURE 21. Timers T4–T7 Block

Timer Registers

There are four control registers that program the timers. The divide by (DIVBY) register programs the clock input to timers T2 and T3. The timer mode register (TMMODE) contains control bits to start and stop timers T1 through T3. It also contains bits to latch, acknowledge and enable interrupts from timers T0 through T3. The control register PWMODE similarly programs the pulse width timers T4 through T7 by allowing them to be started, stopped, and to latch and enable interrupts on underflows. The PORTP register contains bits to preset the outputs and enable the synchronous timer output functions.

Timer Applications

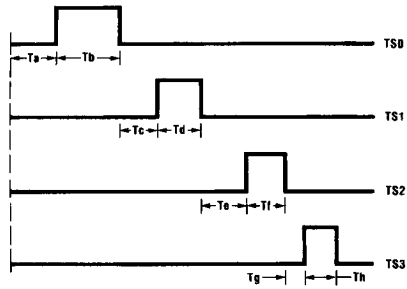
The use of Pulse Width Timers for the generation of various waveforms is easily accomplished by the HPC46064.

Frequencies can be generated by using the timer/register pairs. A square wave is generated when the register value is a constant. The duty cycle can be controlled simply by changing the register value.

TL/DD/11372–20

FIGURE 22. Square Wave Frequency Generation

Synchronous outputs based on Timer T2 can be generated on the 4 outputs TS0–TS3. Each output can be individually programmed to toggle on T2 underflow. Register R2 contains the time delay between events. *Figure 23* is an example of synchronous pulse train generation.



TL/DD/11372–21

FIGURE 23. Synchronous Pulse Generation

WATCHDOG Logic

The WATCHDOG Logic monitors the operations taking place and signals upon the occurrence of any illegal activity. The illegal conditions that trigger the WATCHDOG logic are potentially infinite loops and illegal addresses. Should the WATCHDOG register not be written to before Timer T0 overflows twice, or more often than once every 4096 counts, an infinite loop condition is assumed to have occurred. An illegal condition also occurs when the processor generates an illegal address when in the Single-Chip modes.* Any illegal condition forces the WATCHDOG Output ($\overline{WO}$) pin low. The $\overline{WO}$ pin is an open drain output and can be connected to the $\overline{RESET}$ or NMI inputs or to the users external logic.

***Note:** See Operating Modes for details.

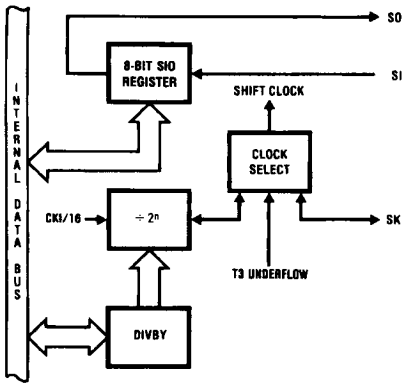
MICROWIRE/PLUS

MICROWIRE/PLUS is used for synchronous serial data communications (see *Figure 24*). MICROWIRE/PLUS has an 8-bit parallel-loaded, serial shift register using SI as the input and SO as the output. SK is the clock for the serial shift register (SIO). The SK clock signal can be provided by an internal or external source. The internal clock rate is programmable by the DIVBY register. A DONE flag indicates when the data shift is completed.

The MICROWIRE/PLUS capability enables it to interface with any of National Semiconductor's MICROWIRE peripherals (i.e., A/D converters, display drivers, EEPROMs).

Powered by ICminer.com Electronic-Library Service CopyRight 2003

MICROWIRE/PLUS (Continued)



TL/DD/11372–22

FIGURE 24. MICROWIRE/PLUS

MICROWIRE/PLUS Operation

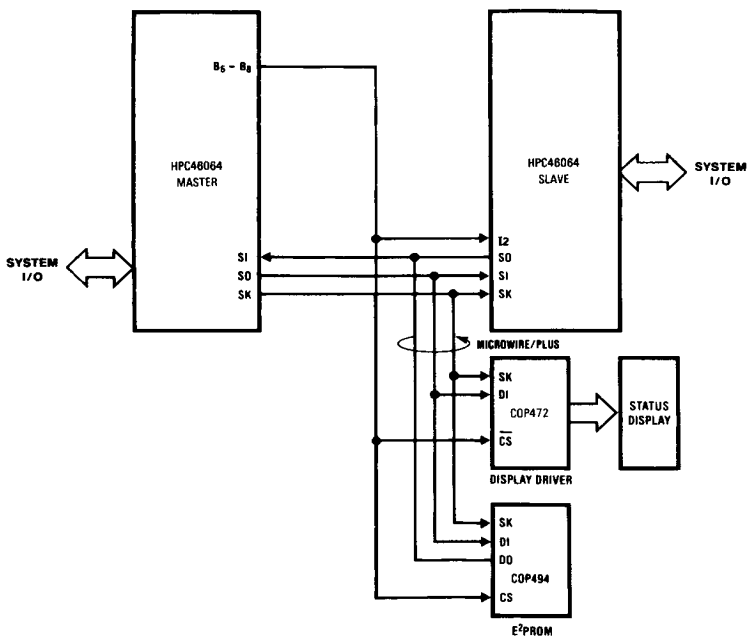
The HPC46064 can enter the MICROWIRE/PLUS mode as the master or a slave. A control bit in the IRCD register determines whether the HPC46064 is the master or slave. The shift clock is generated when the HPC46064 is configured as a master. An externally generated shift clock on the SK pin is used when the HPC46064 is configured as a slave. When the HPC46064 is a master, the DIVBY register programs the frequency of the SK clock. The DIVBY register allows the SK clock frequency to be programmed in 14 selectable binary steps or T3 underflow from 153 Hz to 1.25 MHz with CKI at 20.0 MHz.

The contents of the SIO register may be accessed through any of the memory access instructions. Data waiting to be transmitted in the SIO register is clocked out on the falling edge of the SK clock. Serial data on the SI pin is clocked in on the rising edge of the SK clock.

MICROWIRE/PLUS Application

Figure 25 illustrates a MICROWIRE/PLUS arrangement for an automotive application. The microcontroller-based system could be used to interface to an instrument cluster and various parts of the automobile. The diagram shows two HPC46064 microcontrollers interconnected to other MICROWIRE peripherals. HPC46064 #1 is set up as the master and initiates all data transfers. HPC46064 #2 is set up as a slave answering to the master.

The master microcontroller interfaces the operator with the system and could also manage the instrument cluster in an automotive application. Information is visually presented to the operator by means of an LCD display controlled by the COP472 display driver. The data to be displayed is sent serially to the COP472 over the MICROWIRE/PLUS link. Data such as accumulated mileage could be stored and retrieved from the EEPROM COP494. The slave HPC46064 could be used as a fuel injection processor and generate timing signals required to operate the fuel valves. The master processor could be used to periodically send updated values to the slave via the MICROWIRE/PLUS link. To speed up the response, chip select logic is implemented by connecting an output from the master to the external interrupt input on the slave.



TL/DD/11372–23

FIGURE 25. MICROWIRE/PLUS Application

Powered by ICminer.com Electronic-Library Service CopyRight 2003

HPC46064 UART

The HPC46064 contains a software programmable UART. The UART (see *Figure 26*) consists of a transmit shift register, a receiver shift register and five addressable registers, as follows: a transmit buffer register (TBUF), a receiver buffer register (RBUF), a UART control and status register (ENU), a UART receive control and status register (ENUR) and a UART interrupt and clock source register (ENUI). The ENU register contains flags for transmit and receive functions; this register also determines the length of the data frame (8 or 9 bits) and the value of the ninth bit in transmission. The ENUR register flags framing and data overrun errors while the UART is receiving. Other functions of the ENUR register include saving the ninth bit received in the data frame and enabling or disabling the UART's Wake-up Mode of operation. The determination of an internal or external clock source is done by the ENUI register, as well as selecting the number of stop bits and enabling or disabling transmit and receive interrupts.

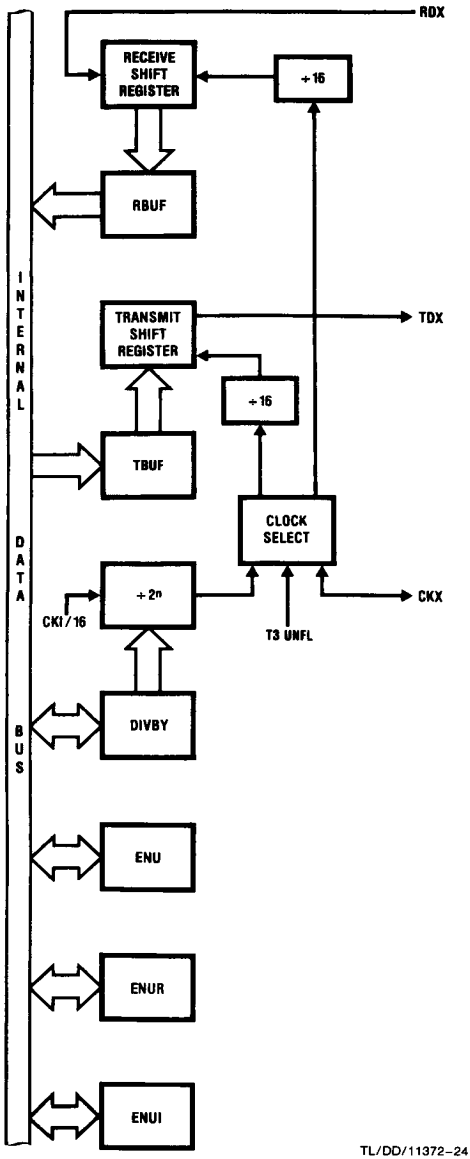
The baud rate clock for the Receiver and Transmitter can be selected for either an internal or external source using two bits in the ENUI register. The internal baud rate is programmed by the DIVBY register. The baud rate may be selected from a range of 8 Hz to 128 kHz in binary steps or T3 underflow. By selecting a 9.83 MHz crystal, all standard baud rates from 75 baud to 38.4 kBaud can be generated. The external baud clock source comes from the CKX pin. The Transmitter and Receiver can be run at different rates by selecting one to operate from the internal clock and the other from an external source.

The HPC46064 UART supports two data formats. The first format for data transmission consists of one start bit, eight data bits and one or two stop bits. The second data format for transmission consists of one start bit, nine data bits, and one or two stop bits. Receiving formats differ from transmission only in that the Receiver always requires only one stop bit in a data frame.

UART Wake-up Mode

The HPC46064 UART features a Wake-up Mode of operation. This mode of operation enables the HPC46064 to be networked with other processors. Typically in such environments, the messages consist of addresses and actual data. Addresses are specified by having the ninth bit in the data frame set to 1. Data in the message is specified by having the ninth bit in the data frame reset to 0.

The UART monitors the communication stream looking for addresses. When the data word with the ninth bit set is received, the UART signals the HPC46064 with an interrupt. The processor then examines the content of the receiver buffer to decide whether it has been addressed and whether to accept subsequent data.



TL/DD/11372–24

FIGURE 26. UART Block Diagram

Powered by ICminer.com Electronic-Library Service CopyRight 2003

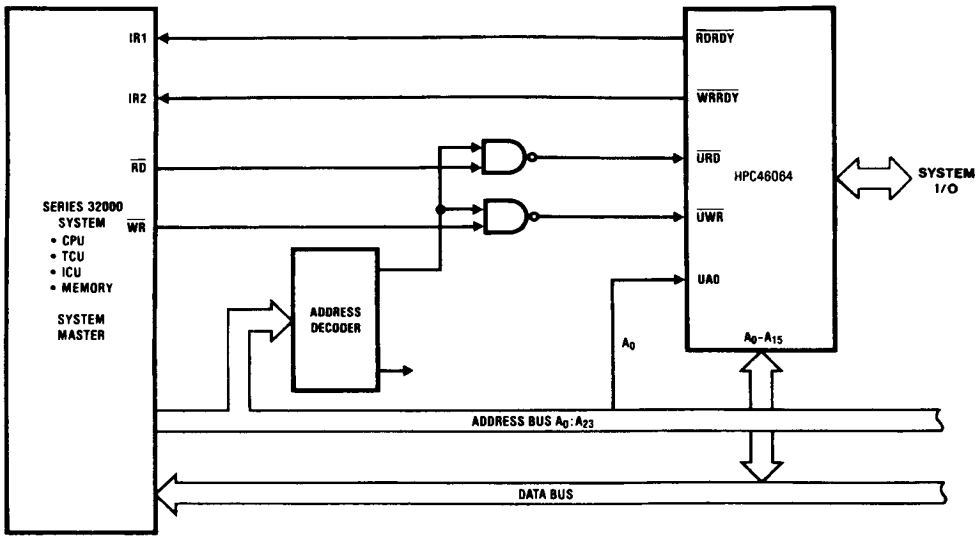
Universal Peripheral Interface

The Universal Peripheral Interface (UPI) allows the HPC46064 to be used as an intelligent peripheral to another processor. The UPI could thus be used to tightly link two HPC46064's and set up systems with very high data exchange rates. Another area of application could be where an HPC46064 is programmed as an intelligent peripheral to a host system such as the Series 32000® microprocessor. *Figure 27* illustrates how an HPC46064 could be used as an intelligent peripherial for a Series 32000-based application.

The interface consists of a Data Bus (port A), a Read Strobe ($\overline{URD}$), a Write Strobe ($\overline{UWR}$), a Read Ready Line ($\overline{RDRDY}$), a Write Ready Line ($\overline{WRRDY}$) and one Address Input (UA0). The data bus can be either eight or sixteen bits wide.

The $\overline{URD}$ and $\overline{UWR}$ inputs may be used to interrupt the HPC46064. The $\overline{RDRDY}$ and $\overline{WRRDY}$ outputs may be used to interrupt the host processor.

The UPI contains an Input Buffer (IBUF), an Output Buffer (OBUF) and a Control Register (UPIC). In the UPI mode, port A on the HPC46064 is the data bus. UPI can only be used if the HPC46064 is in the Single-Chip mode.



TL/DD/11372-25

FIGURE 27. HPC46064 as a Peripheral: (UPI Interface to Series 32000 Application)

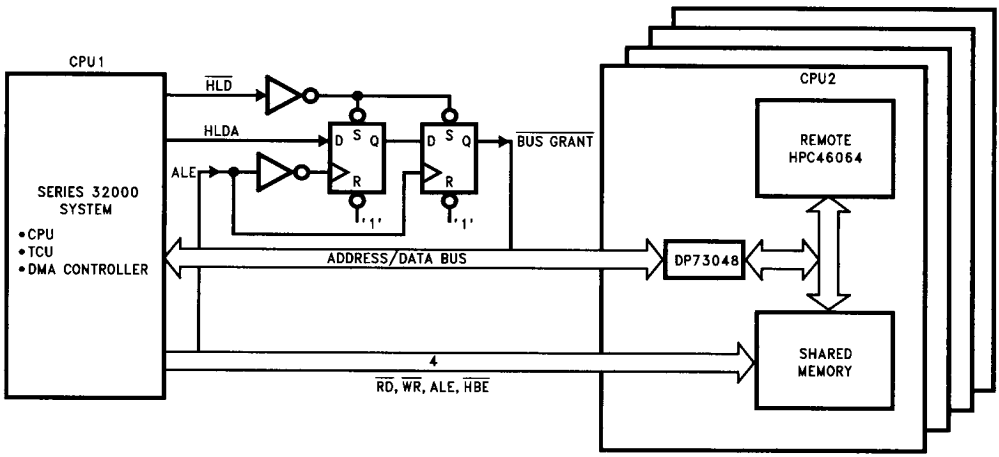
Powered by ICminer.com Electronic-Library Service CopyRight 2003

Shared Memory Support

Shared memory access provides a rapid technique to exchange data. It is effective when data is moved from a peripheral to memory or when data is moved between blocks of memory. A related area where shared memory access proves effective is in multiprocessing applications where two CPUs share a common memory block. The HPC46064 supports shared memory access with two pins. The pins are the RDY/$\overline{HLD}$ input pin and the $\overline{HLDA}$ output pin. The user can software select either the Hold or Ready function by the state of a control bit. The $\overline{HLDA}$ output is multiplexed onto port B.

The host uses DMA to interface with the HPC46064. The host initiates a data transfer by activating the $\overline{HLD}$ input of the HPC46064. In response, the HPC46064 places its system bus in a TRI-STATE Mode, freeing it for use by the host. The host waits for the acknowledge signal ($\overline{HLDA}$) from the HPC46064 indicating that the sytem bus is free. On receiving the acknowledge, the host can rapidly transfer data into, or out of, the shared memory by using a conventional DMA controller. Upon completion of the message transfer, the host removes the HOLD request and the HPC46064 resumes normal operations.

To insure proper operation, the interface logic shown is recommended as the means for enabling and disabling the user's bus. *Figure 28* illustrates an application of the shared memory interface between the HPC46064 and a Series 32000 system.



TL/DD/11372–26

FIGURE 28. Shared Memory Application: HPC46064 Interface to Series 32000 System

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Memory

The HPC46064 has been designed to offer flexibility in memory usage. A total address space of 64 Kbytes can be addressed with 16 Kbytes of ROM and 512 bytes of RAM available on the chip itself. The ROM may contain program instructions, constants or data. The ROM and RAM share the same address space allowing instructions to be executed out of RAM.

Program memory addressing is accomplished by the 16-bit program counter on a byte basis. Memory can be addressed directly by instructions or indirectly through the B, X and SP registers. Memory can be addressed as words or bytes. Words are always addressed on even-byte boundaries. The HPC46064 uses memory-mapped organization to support registers, I/O and on-chip peripheral functions.

The HPC46064 memory address space extends to 64 Kbytes and registers and I/O are mapped as shown in Table IV.

TABLE IV. HPC46064 Memory Map

FFFF:FFF0 FFEF:FFD0 FFCF:FFCE : : E001:C000	Interrupt Vectors JSRP Vectors On-Chip ROM*	USER MEMORY
BFFF:BFFE : : 0301:0300	External Expansion Memory	
02FF:02FE : : 01C1:01C0	On-Chip RAM	USER RAM
0195:0194	WATCHDOG Address	WATCHDOG Logic
0192 0191:0190 018F:018E 018D:018C 018B:018A 0189:0188 0187:0186 0185:0184 0183:0182 0181:0180	T0CON Register TMMODE Register DIVBY Register T3 Timer R3 Register T2 Timer R2 Register I2CR Register/ R1 I3CR Register/ T1 I4CR Register	Timer Block T0:T3
015E:015F 015C 0153:0152 0151:0150 014F:014E 014D:014C 014B:014A 0149:0148 0147:0146 0145:0144 0143:0142 0141:0140	EICR EICON Port P Register PWMODE Register R7 Register T7 Timer R6 Register T6 Timer R5 Register T5 Timer R4 Register T4 Timer	Timer Block T4:T7

0128 0126 0124 0122 0120	ENUR Register TBUF Register RBUF Register ENUI Register ENU Register	UART
0104	Port D Input Register	
00F5:00F4 00F3:00F2 00F1:00F0	BFUN Register DIR B Register DIR A Register / IBUF	PORTS A & B CONTROL
00E6	UPIC Register	UPI CONTROL
00E3:00E2 00E1:00E0	Port B Port A / OBUF	PORTS A & B
00DE 00DD:00DC 00D8 00D6 00D4 00D2 00D0	Reserved HALT Enable Register Port I Input Register SIO Register IRCD Register IRPD Register ENIR Register	PORT CONTROL & INTERRUPT CONTROL REGISTERS
00CF:00CE 00CD:00CC 00CB:00CA 00C9:00C8 00C7:00C6 00C5:00C4 00C3:00C2 00C0	X Register B Register K Register A Register PC Register SP Register Reserved PSW Register	HPC CORE REGISTERS
00BF:00BE : : 0001:0000	On-Chip RAM	USER RAM

***Note:** The HPC46064 On-Chip ROM is on addresses C000:FFFF and the External Expansion Memory is 0300:BFFF. The HPC46004 have no On-Chip ROM, External Memory is 0300:FFFF.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Design Considerations

Designs using the HPC family of 16-bit high speed CMOS microcontrollers need to follow some general guidelines on usage and board layout.

Floating inputs are a frequently overlooked problem. CMOS inputs have extremely high impedance and, if left open, can float to any voltage. You should thus tie unused inputs to V_{CC} or ground, either through a resistor or directly. Unlike the inputs, unused output should be left floating to allow the output to switch without drawing any DC current.

To reduce voltage transients, keep the supply line's parasitic inductances as low as possible by reducing trace lengths, using wide traces, ground planes, and by decoupling the supply with bypass capacitors. In order to prevent additional voltage spiking, this local bypass capacitor must exhibit low inductive reactance. You should therefore use high frequency ceramic capacitors and place them very near the IC to minimize wiring inductance.

- Keep V_{CC} bus routing short. When using double sided or multilayer circuit boards, use ground plane techniques.
- Keep ground lines short, and on PC boards make them as wide as possible, even if trace width varies. Use separate ground traces to supply high current devices such as relay and transmission line drivers.
- In systems mixing linear and logic functions and where supply noise is critical to the analog components' performance, provide separate supply buses or even separate supplies.
- If you use local regulators, bypass their inputs with a tantalum capacitor of at least 1 μF and bypass their outputs with a 10 μF to 50 μF tantalum or aluminum electrolytic capacitor.
- If the system uses a centralized regulated power supply, use a 10 μF to 20 μF tantalum electrolytic capacitor or a 50 μF to 100 μF aluminum electrolytic capacitor to decouple the V_{CC} bus connected to the circuit board.
- Provide localized decoupling. For random logic, a rule of thumb dictates approximately 10 nF (spaced within 12 cm) per every two to five packages, and 100 nF for every 10 packages. You can group these capacitances, but it's more effective to distribute them among the ICs. If the design has a fair amount of synchronous logic with outputs that tend to switch simultaneously, additional decoupling might be advisable. Octal flip-flop and buffers in bus-oriented circuits might also require more decoupling. Note that wire-wrapped circuits can require more decoupling than ground plane or multilayer PC boards.

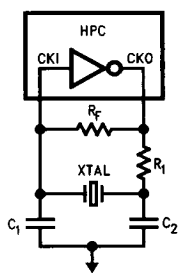
A recommended crystal oscillator circuit to be used with the HPC is shown in *Figure 29*. See Table V for recommended component values. The recommended values given in Table V have yielded consistent results and are made to match a crystal with a 20 pF load capacitance, with some small allowance for layout capacitance.

A recommended layout for the oscillator network should be as close to the processor as physically possible, entirely within "1" distance. This is to reduce lead inductance from long PC traces, as well as interference from other components, and reduce trace capacitance. The layout contains a large ground plane either on the top or bottom surface of the board to provide signal shielding, and a convenient location to ground both the HPC and the case of the crystal.

It is very critical to have an extremely clean power supply for the HPC crystal oscillator. Ideally one would like a V_{CC} and ground plane that provide low inductance power lines to the chip. The power planes in the PC board should be decoupled with three decoupling capacitors as close to the chip as possible. A 1.0 μF, a 0.1 μF, and a 0.001 μF dipped mica or ceramic cap mounted as close to the HPC as is physically possible on the board, using the shortest leads, or surface mount components. This should provide a stable power supply, and noiseless ground plane which will vastly improve the performance of the crystal oscillator network.

TABLE V. HPC Oscillator Table

XTAL Freq (MHz)	R_1 (Ω)
≤2	1500
4	1200
6	910
8	750
10	600
12	470
14	390
16	300
18	220
20	180
22	150
24	120
26	100
28	75
30	62



R_F = 3.3 MΩ
C_1 = 27 pF
C_2 = 33F

XTAL Specifications: The crystal used was an M-TRON Industries MP-1 Series XTAL. "AT" cut, parallel resonant

C_L = 18 pF

Series Resistance is
25Ω @ 25 MHz
40Ω @ 10 MHz
600Ω @ 2 MHz

TL/DD/11372-31

FIGURE 29. Recommended Crystal Circuit

HPC46064 CPU

The HPC46064 CPU has a 16-bit ALU and six 16-bit registers:

Arithmetic Logic Unit (ALU)

The ALU is 16 bits wide and can do 16-bit add, subtract and shift or logic AND, OR and exclusive OR in one timing cycle. The ALU can also output the carry bit to a 1-bit C register.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

HPC46064 CPU (Continued)

Accumulator (A) Register

The 16-bit A register is the source and destination register for most I/O, arithmetic, logic and data memory access operations.

Address (B and X) Registers

The 16-bit B and X registers can be used for indirect addressing. They can automatically count up or down to sequence through data memory.

Boundary (K) Register

The 16-bit K register is used to set limits in repetitive loops of code as register B sequences through data memory.

Stack Pointer (SP) Register

The 16-bit SP register is the pointer that addresses the stack. The SP register is incremented by two for each push or call and decremented by two for each pop or return. The stack can be placed anywhere in user memory and be as deep as the available memory permits.

Program (PC) Register

The 16-bit PC register addresses program memory.

Addressing Modes

ADDRESSING MODES—ACCUMULATOR AS DESTINATION

Register Indirect

This is the "normal" mode of addressing for the HPC46064 (instructions are single-byte). The operand is the memory addressed by the B register (or X register for some instructions).

Direct

The instruction contains an 8-bit or 16-bit address field that directly points to the memory for the operand.

Indirect

The instruction contains an 8-bit address field. The contents of the WORD addressed points to the memory for the operand.

Indexed

The instruction contains an 8-bit address field and an 8- or 16-bit displacement field. The contents of the WORD addressed is added to the displacement to get the address of the operand.

Immediate

The instruction contains an 8-bit or 16-bit immediate field that is used as the operand.

Register Indirect (Auto Increment and Decrement)

The operand is the memory addressed by the X register. This mode automatically increments or decrements the X register (by 1 for bytes and by 2 for words).

Register Indirect (Auto Increment and Decrement) with Conditional Skip

The operand is the memory addressed by the B register. This mode automatically increments or decrements the B register (by 1 for bytes and by 2 for words). The B register is then compared with the K register. A skip condition is generated if B goes past K.

ADDRESSING MODES—DIRECT MEMORY AS DESTINATION

Direct Memory to Direct Memory

The instruction contains two 8- or 16-bit address fields. One field directly points to the source operand and the other field directly points to the destination operand.

Immediate to Direct Memory

The instruction contains an 8- or 16-bit address field and an 8- or 16-bit immediate field. The immediate field is the operand and the direct field is the destination.

Double Register Indirect Using the B and X Registers

Used only with Reset, Set and IF bit instructions; a specific bit within the 64 kbyte address range is addressed using the B and X registers. The address of a byte of memory is formed by adding the contents of the B register to the most significant 13 bits of the X register. The specific bit to be modified or tested within the byte of memory is selected using the least significant 3 bits of register X.

HPC Instruction Set Description

Mnemonic	Description	Action
ARITHMETIC INSTRUCTIONS		
ADD	Add	MA + Meml → MA carry → C
ADC	Add with carry	MA + Meml + C → MA carry → C
ADDS	Add short imm8	A + imm8 → A carry → C
DADC	Decimal add with carry	MA + Meml + C → MA (Decimal) carry → C
SUBC	Subtract with carry	MA − Meml + C → MA carry → C
DSUBC	Decimal subtract w/carry	MA − Meml + C → MA (Decimal) carry → C
MULT	Multiply (unsigned)	MA*Meml → MA & X, 0 → K, 0 → C
DIV	Divide (unsigned)	MA/Meml → MA, rem. → X, 0 → K, 0 → C
DIVD	Divide Double Word (unsigned)	X & MA/Meml → MA, rem → X, 0 → K, Carry → C
IFEQ	If equal	Compare MA & Meml, Do next if equal
IFGT	If greater than	Compare MA & Meml, Do next if MA > Meml
AND	Logical and	MA and Meml → MA
OR	Logical or	MA or Meml → MA
XOR	Logical exclusive-or	MA xor Meml → MA
MEMORY MODIFY INSTRUCTIONS		
INC	Increment	Mem + 1 → Mem
DECSZ	Decrement, skip if 0	Mem − 1 → Mem, Skip next if Mem = 0

Powered by ICminer.com Electronic-Library Service CopyRight 2003

HPC Instruction Set Description (Continued)

Mnemonic	Description	Action
BIT INSTRUCTIONS		
SBIT	Set bit	1 → Mem.bit
RBIT	Reset bit	0 → Mem.bit
IFBIT	If bit	If Mem.bit is true, do next instr.
MEMORY TRANSFER INSTRUCTIONS		
LD	Load	Meml → MA
	Load, incr/decr X	Mem(X) → A, X ±1 (or 2) → X
ST	Store to Memory	A → Mem
X	Exchange	A ↔ Mem
	Exchange, incr/decr X	A ↔ Mem(X), X ±1 (or 2) → X
PUSH	Push Memory to Stack	W → W(SP), SP+2 → SP
POP	Pop Stack to Memory	SP−2 → SP, W(SP) → W
LDS	Load A, incr/decr B, Skip on condition	Mem(B) → A, B ±1 (or 2) → B, Skip next if B greater/less than K
XS	Exchange, incr/decr B, Skip on condition	Mem(B) ↔ A, B ±1 (or 2) → B, Skip next if B greater/less than K
REGISTER LOAD IMMEDIATE INSTRUCTIONS		
LD B	Load B immediate	imm → B
LD K	Load K immediate	imm → K
LD X	Load X immediate	imm → X
LD BK	Load B and K immediate	imm → B,imm → K
ACCUMULATOR AND C INSTRUCTIONS		
CLR A	Clear A	0 → A
INC A	Increment A	A + 1 → A
DEC A	Decrement A	A − 1 → A
COMP A	Complement A	1's complement of A → A
SWAP A	Swap nibbles of A	A15:12 ← A11:8 ← A7:4 ↔ A3:0
RRC A	Rotate A right thru C	C → A15 → ... → A0 → C
RLC A	Rotate A left thru C	C ← A15 ← ... ← A0 ← C
SHR A	Shift A right	0 → A15 → ... → A0 → C
SHL A	Shift A left	C ← A15 ← ... ← A0 ← 0
SC	Set C	1 → C
RC	Reset C	0 → C
IFC	IF C	Do next if C = 1
IFNC	IF not C	Do next if C = 0
TRANSFER OF CONTROL INSTRUCTIONS		
JSRP	Jump subroutine from table	PC → W(SP),SP+2 → SP W(table#) → PC
JSR	Jump subroutine relative	PC → W(SP),SP+2 → SP,PC+# → PC (#is +1025 to −1023)
JSRL	Jump subroutine long	PC → W(SP),SP+2 → SP,PC+# → PC
JP	Jump relative short	PC+# → PC(# is +32 to −31)
JMP	Jump relative	PC+# → PC(#is +257 to −255)
JMPL	Jump relative long	PC+# → PC
JID	Jump indirect at PC + A	PC+A+1 → PC
JIDW		then Mem(PC)+PC → PC
NOP	No Operation	PC + 1 → PC
RET	Return	SP−2 → SP,W(SP) → PC
RETSK	Return then skip next	SP−2 → SP,W(SP) → PC, & skip
RETI	Return from interrupt	SP−2 → SP,W(SP) → PC, interrupt re-enabled

Note: W is 16-bit word of memory
MA is Accumulator A or direct memory (8- or 16-bit)
Mem is 8-bit byte or 16-bit word of memory
Meml is 8- or 16-bit memory or 8- or 16-bit immediate data
imm is 8-bit or 16-bit immediate data
imm8 is 8-bit immediate data only

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Memory Usage

Number of Bytes for Each Instruction (number in parenthesis is 16-Bit field)

	Using Accumulator A						To Direct Memory			
	Reg Indir. (B)	Reg Indir. (X)	Direct	Indir	Index	Immed.	Direct *	Direct **	Immed. *	Immed. **
LD	1	1	2(4)	3	4(5)	2(3)	3(5)	5(6)	3(4)	5(6)
X	1	1	2(4)	3	4(5)	—	—	—	—	—
ST	1	1	2(4)	3	4(5)	—	—	—	—	—
ADC	1	2	3(4)	3	4(5)	4(5)	4(5)	5(6)	4(5)	5(6)
ADDS	—	—	—	—	—	2	—	—	—	—
SBC	1	2	3(4)	3	4(5)	4(5)	4(5)	5(6)	4(5)	5(6)
DADC	1	2	3(4)	3	4(5)	4(5)	4(5)	5(6)	4(5)	5(6)
DSBC	1	2	3(4)	3	4(5)	4(5)	4(5)	5(6)	4(5)	5(6)
ADD	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
MULT	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
DIV	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
DIVD	1	2	3(4)	3	4(5)	—	4(5)	5(6)	4(5)	5(6)
IFEQ	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
IFGT	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
AND	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
OR	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)
XOR	1	2	3(4)	3	4(5)	2(3)	4(5)	5(6)	4(5)	5(6)

*8-bit direct address
**16-bit direct address

Instructions that Modify Memory Directly

	(B)	(X)	Direct	Indir	Index	B&X
SBIT	1	2	3(4)	3	4(5)	1
RBIT	1	2	3(4)	3	4(5)	1
IFBIT	1	2	3(4)	3	4(5)	1
DECSZ	3	2	2(4)	3	4(5)	
INC	3	2	2(4)	3	4(5)	

Immediate Load Instructions

	Immed.
LD B,*	2(3)
LD X,*	2(3)
LD K,*	2(3)
LD BK,*,*	3(5)

Register Indirect Instructions with Auto Increment and Decrement

Register B With Skip		
	(B+)	(B−)
LDS A,*	1	1
XS A,*	1	1

Register X		
	(X+)	(X−)
LD A,*	1	1
X A,*	1	1

Instructions Using A and C

CLR	A	1
INC	A	1
DEC	A	1
COMP	A	1
SWAP	A	1
RRC	A	1
RLC	A	1
SHR	A	1
SHL	A	1
SC		1
RC		1
IFC		1
IFNC		1

Transfer of Control Instructions

JSRP	1
JSR	2
JSRL	3
JP	1
JMP	2
JMPL	3
JID	1
JIDW	1
NOP	1
RET	1
RETSK	1
RETI	1

Stack Reference Instructions

	Direct
PUSH	2
POP	2

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Code Efficiency

One of the most important criteria of a single chip microcontroller is code efficiency. The more efficient the code, the more features that can be put on a chip. The memory size on a chip is fixed so if code is not efficient, features may have to be sacrificed or the programmer may have to buy a larger, more expensive version of the chip.

The HPC46064 has been designed to be extremely code-efficient. The HPC46064 looks very good in all the standard coding benchmarks; however, it is not realistic to rely only on benchmarks. Many large jobs have been programmed onto the HPC46064, and the code savings over other popular microcontrollers has been considerable.

Reasons for this saving of code include the following:

SINGLE BYTE INSTRUCTIONS

The majority of instructions on the HPC46064 are single-byte. There are two especially code-saving instructions: JP is a 1-byte jump. True, it can only jump within a range of plus or minus 32, but many loops and decisions are often within a small range of program memory. Most other micros need 2-byte instructions for any short jumps.

JSRP is a 1-byte call subroutine. The user makes a table of the 16 most frequently called subroutines and these calls will only take one byte. Most other micros require two and even three bytes to call a subroutine. The user does not have to decide which subroutine addresses to put into this table; the assembler can give this information.

EFFICIENT SUBROUTINE CALLS

The 2-byte JSR instructions can call any subroutine within plus or minus 1k of program memory.

MULTIFUNCTION INSTRUCTIONS FOR DATA MOVEMENT AND PROGRAM LOOPING

The HPC46064 has single-byte instructions that perform multiple tasks. For example, the XS instruction will do the following:

1. Exchange A and memory pointed to by the B register
2. Increment or decrement the B register
3. Compare the B register to the K register
4. Generate a conditional skip if B has passed K

The value of this multipurpose instruction becomes evident when looping through sequential areas of memory and exiting when the loop is finished.

BIT MANIPULATION INSTRUCTIONS

Any bit of memory, I/O or registers can be set, reset or tested by the single byte bit instructions. The bits can be addressed directly or indirectly. Since all registers and I/O are mapped into the memory, it is very easy to manipulate specific bits to do efficient control.

DECIMAL ADD AND SUBTRACT

This instruction is needed to interface with the decimal user world.

It can handle both 16-bit words and 8-bit bytes.

The 16-bit capability saves code since many variables can be stored as one piece of data and the programmer does not have to break his data into two bytes. Many applications store most data in 4-digit variables. The HPC46064 supplies 8-bit byte capability for 2-digit variables and literal variables.

MULTIPLY AND DIVIDE INSTRUCTIONS

The HPC46064 has 16-bit multiply, 16-bit by 16-bit divide, and 32-bit by 16-bit divide instructions. This saves both code and time. Multiply and divide can use immediate data or data from memory. The ability to multiply and divide by immediate data saves code since this function is often needed for scaling, base conversion, computing indexes of arrays, etc.

Development Support

HPC Microcontroller Development System

The HPC microcontroller development system is an in-system emulator (ISE) designed to support the entire family of HPC Microcontrollers. The complete package of hardware and software tools combined with a host system provides a powerful system for design, development and debug of HPC based designs. Software tools are available for IBM® PC-AT® (MS-DOS, PC-DOS) and for Unix based multi-user Sun® Sparcstation (SunOS™).

The stand alone units comes complete with a power supply and external emulation POD. This unit can be connected to various host systems through an RS-232 link. The software package includes an ANSI compatible C-Compiler, Linker, Assembler and librarian package. Source symbolic debug capability is provided through a user friendly MS-windows 3.0 interface for IBM PC-AT environment and through a line debugger under Sunview for Sun Sparcstations.

The ISE provides fully transparent in-system emulation at speeds up to 20 MHz 1 waitstate. A 2k word (48-bit wide) trace buffer gives trace trigger and non-intrusive monitoring of the system. External triggering is also available through an external logic interface socket on the POD. Direct EPROM programming can be done through the use of externally mounted EPROM socket. Form-Fit-Function emulator programming is supported by a programming board included with the system. Comprehensive on-line help and diagnostics features reduce user's design and debug time. 8 hardware breakpoints (Address/range), 64 Kbytes of user memory, and break on external events are some of the other features offered.

Hewlett Packard model HP64775 Emulator/Analyzer providing in-system emulation for up to 30 MHz 1 waitstate is also available. Contact your local sales office for technical details and support.

Powered by ICminer.com Electronic-Library Service CopyRight 2003

Development Support (Continued)

Development Tools Selection Table

Product	Order Number	Description	Included	Manual Number
HPC16004/ 16064	HPC-DEV-ISE4 HPC-DEV-ISE4-E	HPC In-System Emulator HPC in-System Emulator for Europe and South East Asia	HPC MDS User's Manual MDS Comm User's Manual HPC Emulator Programmer User's Manual HPC16004/16064 Manual	420420184-001 424420188-001 420421313-001
	NPC-DEV-IBMA	Assembler/Linker/ Library Package for IBM PC-AT	HPC Assembler/Linker Librarian User's Manual	424410836-001
	HPC-DEV-IBMC	C Compiler/Assembler/ Linker/Library Package for IBM PC-AT	HPC C Compiler User's Manual HPC Assembler/Linker/Library User's Manual	424410883-001 424410836-001
	HPC-DEV-WDBC	Source Symbolic Debugger for IBM PC-AT C Compiler/Assembler/ Linker Library Package for IBM PC-AT	Source/Symbolic Debugger User's Manual HPC C Compiler User's Manual HPC Assembler/Linker/Library User's Manual	424420189-001 424410883-001 424410836-001
	HPC-DEV-SUNC	C-Compiler/Assembler/ Linker Library Package for Sun Sparcstation	HPC C Compiler User's Manual HPC Assembler/Linker/Library User's Manual	
	HPC-DEV-SUNDB	Source/Symbolic Debugger for Sun Sparcstation C Compiler/Assembler/Linker Library Package	Source/Symbolic Debugger User's Manual HPC C Compiler User's Manual HPC Assembler/Linker/Library User's Manual	
Complete System: HPC16004/ 16064	HPC-DEV-SYS4	HPC In-System Emulator with C Compiler/Assembler/ Linker/Library and Source Symbolic Debugger		
	HPC-DEV-SYS4-E	Same for Europe and South East Asia		

How to Order

To order a complete development package, select the section for the microcontroller to be developed and order the parts listed.

DIAL-A-HELPER

Dial-A-Helper is a service provided by the Microcontroller Applications group. Dial-A-Helper is an Electronic Bulletin Board Information system and additionally, provides the capability of remotely accessing the development system at a customer site.

INFORMATION SYSTEM

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

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

Order P/N: MDS-DIAL-A-HLP

Information System Package Contains:
- Dial-A-Helper Users Manual
- Public Domain Communications Software

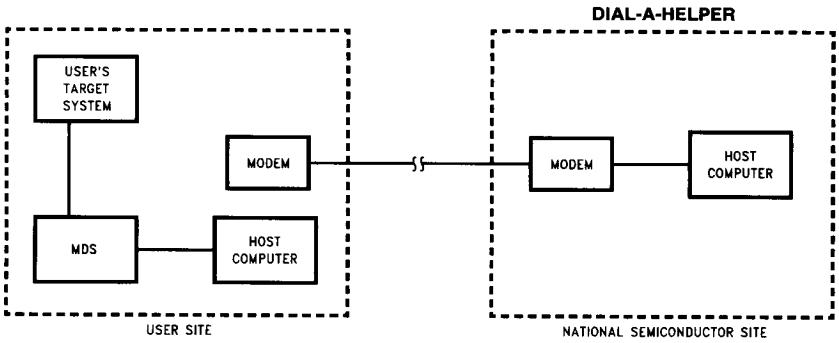
Powered by ICminer.com Electronic-Library Service CopyRight 2003

Development Support (Continued)

FACTORY APPLICATIONS SUPPORT

Dial-A-Helper also provides immediate factory applications support. If a user is having difficulty in operating a development system, he can leave messages on our electronic bulletin board, which we will respond to.

Voice: (408) 721-5582
Modem: (408) 739-1162
Baud: 300 or 1200 baud
Set-Up: Length: 8-Bit
Parity: None
Stop Bit: 1
Operation: 24 Hrs. 7 Days

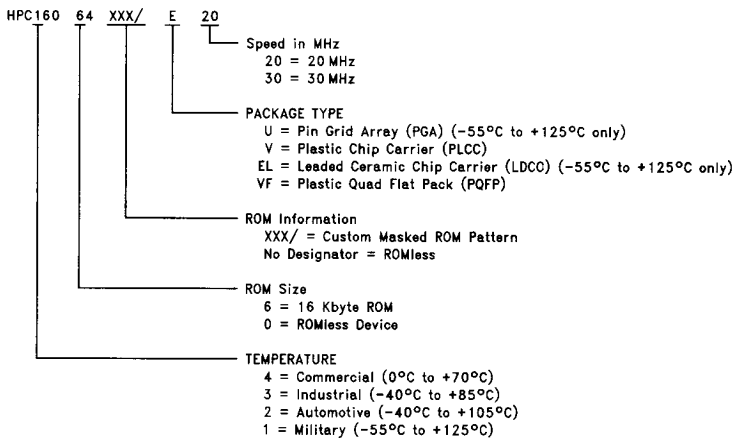


TL/DD/11372–35

Part Selection

The HPC family includes devices with many different options and configurations to meet various application needs. The number HPC46064 has been generically used throughout this datasheet to represent the whole family of parts. The following chart explains how to order various options available when ordering HPC family members.

Note: All options may not currently be available.



TL/DD/11372–36

FIGURE 8. HPC Family Part Numbering Scheme

Examples

HPC46004V20 — ROMless, Commercial temperature (0°C to 70°C), PLCC
HPC16064XXX/U20— 16k masked ROM, Military temperature (−55°C to 125°C), PGA
HPC26004XXX/V20— ROMless, Automotive temperature (−40°C to +105°C), PLCC

Powered by ICminer.com Electronic-Library Service CopyRight 2003