National Semiconductor

PRELIMINARY

HPC16164/26164/36164/46164 HPC16104/26104/36104/46104 HPC16064/26064/36064/46064 HPC16004/26004/36004/46004 High-Performance microControllers with A/D

General Description

The HPC16164, HPC16104, HPC16064 and HPC16004 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 HPC16164 and HPC16104 have 16k bytes of on-chip ROM. The HPC16104 and HPC16104 have 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 "HPC16164" is used throughout this datasheet to refer to the HPC16164, HPC16104, HPC16064 and HPC16004 devices unless otherwise specified.

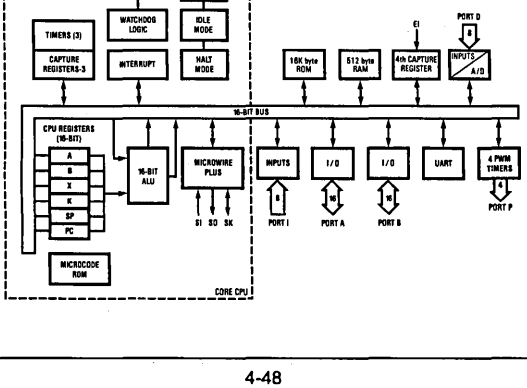
The HPC16164 and HPC16104 have, as an on-board peripheral, an 8-channel 8-bit Analog-to-Digital Converter. This A/D converter can operate in single-ended mode where the analog input voltage is applied across one of the eight input channels (D0-D7) and AGND. The A/D converter can also operate in differential mode where the analog input voltage is applied across two adjacent input channels. The A/D converter will convert up to eight channels in single-ended mode and up to four channel pairs in differential mode. The HPC16064 and HPC16004 do not have onboard A/D.

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, LCC, LDCC, PGA and 84-pin TapePak® packages.

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
  - 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
- A/D—8-channel 8-bit analog-to-digital converter with conversion time minimum 6.6 μs for single conversion
- A/D—supports conversions in "quiet mode"

Block Diagram (HPC16164 with 16k ROM shown)



TL/DD/9682-1

Features (Continued)

- 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 (HPC16104)
- 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

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
ESD Rating	2000V

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.*

20 MHz

DC Electrical Characteristics

V_{CC} = 5.0V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46164/HPC46104, HPC46064/HPC46004, −40°C to +85°C for HPC36164/HPC36104, HPC36064/HPC36004, −40°C to +105°C for HPC26164/HPC26104, HPC26064/HPC26004, −55°C to +125°C for HPC16164/HPC16104, HPC16064/HPC16004

Symbol	Parameter	Test Conditions	Min	Max	Units
I_{CC_1}	Supply Current	V_{CC} = 5.5V, f_{in} = 20.0 MHz (Note 1)		40	mA
		V_{CC} = 5.5V, f_{in} = 2.0 MHz (Note 1)		15	mA
I_{CC_2}	IDLE Mode Current	V_{CC} = 5.5V, f_{in} = 20.0 MHz, (Note 1)		3.5	mA
		V_{CC} = 5.5V, f_{in} = 2.0 MHz, (Note 1)		1	mA
I_{CC_3}	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 RESET, NMI, CKI AND WO (SCHMITT TRIGGERED)					
V_{IH_1}	Logic High		0.9 V_{CC}		V
V_{IL_1}	Logic Low			0.1 V_{CC}	V
ALL OTHER INPUTS					
V_{IH_2}	Logic High		0.7 V_{CC}		V
V_{IL_2}	Logic Low			0.2 V_{CC}	V
I_{LI}	Input Leakage Current			±1	μA
I_{L2}	Input Leakage Current RDY/HLD, EXU1		−3	−50	μA
I_{L3}	Input Leakage Current B12		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_{OH_1}	Logic High (CMOS)	I_{OH} = −10 μA (Note 2)	V_{CC} − 0.1		V
V_{OL_1}	Logic Low (CMOS)	I_{OH} = 10 μA (Note 2)		0.1	V
V_{OH_2}	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_{OL_2}		I_{OL} = 3 mA		0.4	V
V_{OH_3}	Other Port Pin Drive, WO (open drain) (B_0–B_9, B_{13}, B_{14}, P_0–P_3)	I_{OH} = −1.6 mA	2.4		V
V_{OL_3}		I_{OL} = 0.5 mA		0.4	V
V_{OH4}	ST1 and ST2 Drive	I_{OH} = −6 mA	2.4		V
V_{OL_4}		I_{OL} = 1.6 mA		0.4	V
V_{RAM}	RAM Keep-Alive Voltage	(Note 3)	2.5	V_{CC}	V
I_{OZ}	TRI-STATE® Leakage Current			±5	μA

Note 1: I_{CC_1}, I_{CC_2}, I_{CC_3} measured with no external drive (I_{OH} and I_{OL} = 0, I_{IH} and I_{IL} = 0). I_{CC_1} is measured with $\overline{RESET}$ = GND. I_{CC_3} is measured with NMI = V_{CC} and A/D inactive. CKI driven to V_{IH1} and V_{IL1} with rise and fall times less than 10 ns. V_{REF} = AGND = GND.

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

Note 3: Test duration is 100 ms.

20 MHz

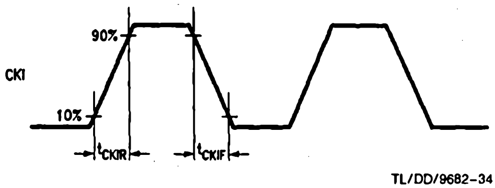
AC Electrical Characteristics

V_{CC} = 5.0V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46164/HPC46104, HPC46064/HPC46004, −40°C to +85°C for HPC36164/HPC36104, HPC36064/HPC36004, −40°C to +105°C for HPC26164/HPC26104, HPC26064/HPC26004, −55°C to +125°C for HPC16164/HPC16104, HPC16064/HPC16004

Symbol	Parameter	Min	Max	Units
f_C = CKI freq.	Operating Frequency	2	20	MHz
$t_{C1} = 1/f_C$	Clock Period	50	500	ns
t_{CKIR} (Note 3)	CKI Rise Time		7	ns
t_{CKIF} (Note 3)	CKI Fall Time		7	ns
$[t_{CKIH}/(t_{CKIH} + t_{CKIL})]100$	Duty Cycle	45	55	%
$t_C = 2/f_C$	Timing Cycle	100		ns
$t_{LL} = \frac{1}{2} t_C - 9$	ALE Pulse Width	41		ns
t_{DC1C2R} (Notes 1, 2)	Delay from CKI Falling Edge to CK2 Rising Edge	0	55	ns
t_{DC1C2F} (Notes 1, 2)	Delay from CKI Falling Edge to CK2 Falling Edge	0	55	ns
$t_{DC1ALER}$ (Notes 1, 2)	Delay from CKI Rising Edge to ALE Rising Edge	0	35	ns
$t_{DC1ALEF}$ (Notes 1, 2)	Delay from CKI Rising Edge to ALE Falling Edge	0	35	ns
$t_{DC2ALER} = \frac{1}{4} t_C + 20$ (Note 2)	Delay from CK2 Rising Edge to ALE Rising Edge		45	ns
$t_{DC2ALEF} = \frac{1}{4} t_C + 20$ (Note 2)	Delay from CK2 Falling Edge to ALE Falling Edge		45	ns
$t_{ST} = \frac{1}{4} t_C - 7$	Address Valid to ALE Falling Edge	18		ns
$t_{VP} = \frac{1}{4} t_C - 5$	Address Hold from ALE Falling Edge	20		ns
$t_{WAIT} = t_C$	Wait State Period	100		ns
$f_{XIN} = f_C/19$	External Timer Input Frequency		1.052	MHz
$t_{XIN} = t_C$	Pulse Width for Timer Inputs	100		ns
$f_{XOUT} = f_C/16$	Timer Output Frequency		1.25	MHz
$f_{MW} = f_C/19$	External MICROWIRE/PLUS Clock Input Frequency		1.052	MHz
$f_U = f_C/8$	External UART Clock Input Frequency		2.5	MHz

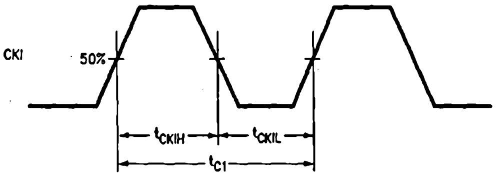
CKI Input Signal Characteristics

Rise/Fall Time



TL/DD/9682-34

Duty Cycle



TL/DD/9682-35

20 MHz

Read Cycle Timing

Symbol	Parameter	Min	Max	Units
$t_{ARR} = \frac{1}{4} t_C - 5$	ALE Falling Edge to $\overline{RD}$ Falling Edge	20		ns
$t_{RW} = \frac{1}{2} t_C + WS - 10$	$\overline{RD}$ Pulse Width	140		ns
$t_{DR} = \frac{3}{4} t_C - 15$	Data Hold after Rising Edge of $\overline{RD}$	0	60	ns
$t_{ACC} = t_C + WS - 55$ (Note 2)	Address Valid to Input Data Valid		145	ns
$t_{RD} = \frac{1}{2} t_C + WS - 65$	$\overline{RD}$ Falling Edge to Input Data Valid		85	ns
$t_{RDA} = t_C - 5$	$\overline{RD}$ Rising Edge to Address Valid	95		ns

Write Cycle Timing

Symbol	Parameter	Min	Max	Units
$t_{ARW} = \frac{1}{2} t_C - 5$	ALE Falling Edge to $\overline{WR}$ Falling Edge	45		ns
$t_{WW} = \frac{3}{4} t_C + WS - 15$	$\overline{WR}$ Pulse Width	160		ns
$t_{HW} = \frac{1}{4} t_C - 5$	Data Hold after Rising Edge of $\overline{WR}$	20		ns
$t_V = \frac{1}{2} t_C + WS - 5$	Data Valid before Rising Edge of $\overline{WR}$	145		ns

Note: Bus Output (Port A) C_L = 100 pF, CK2 Output C_L = 50 pF, other Outputs C_L = 80 pF. AC parameters are tested using DC Characteristics Inputs and non CMOS Outputs. Measurement of AC Specifications is done with external clock driving CKI with 50% duty cycle. The capacitive load on CKO must be kept below 15 pF or AC measurement will be skewed.

Note: WS = t_{WAIT} • number of pre-programmed wait states. Minimum and maximum values are calculated from maximum operating frequency with one (1) wait state pre-programmed.

Note 1: Do not design with this parameter unless CKI is driven with an active signal. When using a passive crystal circuit, CKI or CKO ***should not*** be connected to any external logic since any load (besides the passive components in the crystal circuit) will affect the stability of the crystal unpredictably.

Note 2: These are not directly tested parameters. Therefore the given min/max value cannot be guaranteed. It is, however, derived from measured parameters, and may be used for system design with a very high confidence level.

Note 3: This is guaranteed by design and not tested.

Ready/Hold Timing

Symbol	Parameter	Min	Max	Units
$t_{DAR} = \frac{1}{4} t_C + WS - 50$	Falling Edge of ALE to Falling Edge of RDY		75	ns
$t_{RWP} = t_C$	RDY Pulse Width	100		ns
$t_{SALE} = \frac{3}{4} t_C + 40$	Falling Edge of $\overline{HLD}$ to Rising Edge of ALE	115		ns
$t_{HWP} = t_C + 10$	$\overline{HLD}$ Pulse Width	110		ns
$t_{HAD} = \frac{3}{4} t_C + 85$	Rising Edge on $\overline{HLD}$ to Rising Edge on $\overline{HLDA}$		160	ns
$t_{HAE} = t_C + 100$	Falling Edge on $\overline{HLD}$ to Falling Edge on $\overline{HLDA}$		200*	ns
$t_{BF} = \frac{1}{2} t_C + 66$	Data Valid after Falling Edge on $\overline{HLDA}$		116†	ns
$t_{BE} = \frac{1}{2} t_C + 66$	Bus Enable from Rising Edge of $\overline{HLDA}$	116†		ns

***Note:** t_{HAE} may be as long as ($3t_C + 4ws + 72t_C + 90$) depending on which instruction is being executed, the addressing mode and number of wait states. t_{HAE} maximum value is for the optimal case.

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

20 MHz

MICROWIRE/PLUS Timing

Symbol	Parameter	Min	Max	Units
t_{UWS}	MICROWIRE Setup Time			ns
Master		100		
Slave		20		
t_{UWH}	MICROWIRE Hold Time			ns
Master		20		
Slave		50		
t_{UWV}	MICROWIRE Output Valid Time			ns
Master			50	
Slave			150	

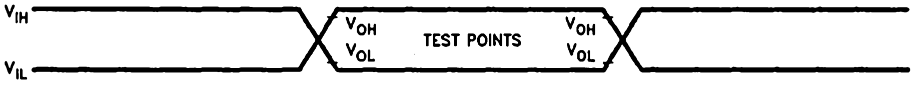
UPI Read/Write Timing

Symbol	Parameter	Min	Max	Units
t_{UAS}	Address Setup Time to Falling Edge of $\overline{\text{URD}}$	10		ns
t_{UAH}	Address Hold Time from Rising Edge of $\overline{\text{URD}}$	10		ns
t_{RPW}	$\overline{\text{URD}}$ Pulse Width	100		ns
t_{OE}	$\overline{\text{URD}}$ Falling Edge to Output Data Valid	0	60	ns
t_{OD}	Rising Edge of $\overline{\text{URD}}$ to Output Data Invalid (Note 4)	5	35	ns
t_{DRDY}	$\overline{\text{RDRDY}}$ Delay from Rising Edge of $\overline{\text{URD}}$		70	ns
t_{WDW}	$\overline{\text{UWR}}$ Pulse Width	40		ns
t_{UDS}	Input Data Valid before Rising Edge of $\overline{\text{UWR}}$	10		ns
t_{UDH}	Input Data Hold after Rising Edge of $\overline{\text{UWR}}$	15		ns
t_A	$\overline{\text{WRRDY}}$ Delay from Rising Edge of $\overline{\text{UWR}}$		70	ns

Note: Bus Output (Port A) C_L = 100 pF, CK2 Output C_L = 50 F, other Outputs C_L = 80 pF.

Note 4: Guaranteed by design.

Input and Output for AC Tests



TL/DD/9682-40

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".

20 MHz

A/D Converter Specifications

V_{CC} = 5V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46164/HPC46104, −40°C to +85°C for HPC36164/HPC36104, −40°C to +105°C for HPC26164/HPC26104, −55°C to +125°C for HPC16164/HPC16104

Symbol	Parameter	Min	Max	Units
	Resolution		8	bits
f_{CCLK}	Clock Frequency (Note 4)	0.1	1.6	MHz
$t_{CON} = 8.5/f_{CCLK}$	Conversion Time (Note 3)	5.3		μs
V_{REF}	Reference Voltage Input (AGND = 0V)	3.0	V_{CC}	V
	Total Unadjusted Error (Note 1) (V_{REF} = 5.000V)		±½	LSB
R_{VREF}	Reference Input Resistance (Note 5)	1.6	4.8	kΩ
	DC Common Mode Error		±¼	LSB
	Power Supply Sensitivity (V_{REF} = 5.000V, V_{CC} = 5V ±10%)		±¼	LSB
	Voltage Reference Tolerance (V_{REF})		TBD	LSB
	Port D Input Capacitance (Note 5)		35	pF
	Analog Input Voltage Range (Note 2)	GND − 0.05	V_{CC} + 0.05	V
	On Channel Leakage		1	μA
	Off Channel Leakage		1	μA

Note 1: Total unadjusted error includes offset, full-scale, and multiplexer errors.

Note 2: 8 single-ended or 4 differential channels. Inherent sample and hold for single-ended inputs (GND = Pin 62).

Note 3: Conversion time does not include sample/hold time plus overhead. Sample and hold time is $2/f_{CCLK}$. Overhead is $1/f_{CCLK}$

Note 4: Clock supplied to A/D converter is derived from CKI. See A/D description for details.

Note 5: This is guaranteed by design and not tested.

30 MHz

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
ESD Rating	2000V

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.*

20 MHz

DC Electrical Characteristics

V_{CC} = 5.0V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46164/HPC46104, HPC46064/HPC46004, −40°C to +85°C for HPC36164/HPC36104, HPC36064/HPC36004, −40°C to +105°C for HPC26164/HPC26104, HPC26064/HPC26004, −55°C to +125°C for HPC16164/HPC16104, HPC16064/HPC16004

Symbol	Parameter	Test Conditions	Min	Max	Units
I_{CC_1}	Supply Current	V_{CC} = 5.5V, f_{in} = 30.0 MHz (Note 1)		TBD	mA
		V_{CC} = 5.5V, f_{in} = 2.0 MHz (Note 1)		15	mA
I_{CC_2}	IDLE Mode Current	V_{CC} = 5.5V, f_{in} = 30.0 MHz, (Note 1)		TBD	mA
		V_{CC} = 5.5V, f_{in} = 2.0 MHz, (Note 1)		2	mA
I_{CC_3}	HALT Mode Current	V_{CC} = 5.5V, f_{in} = 0 kHz, (Note 1)		400	μA
		V_{CC} = 2.5V, f_{in} = 0 kHz, (Note 1)		250	μA
INPUT VOLTAGE LEVELS RESET, NMI, CKI AND WO (SCHMITT TRIGGERED)					
V_{IH_1}	Logic High		0.9 V_{CC}		V
V_{IL_1}	Logic Low			0.1 V_{CC}	V
ALL OTHER INPUTS					
V_{IH_2}	Logic High		0.7 V_{CC}		V
V_{IL_2}	Logic Low			0.2 V_{CC}	V
I_{LI}	Input Leakage Current			±1	μA
I_{L2}	Input Leakage Current RDY/HLD, EXU1		−3	−50	μA
I_{L3}	Input Leakage Current B12		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_{OH_1}	Logic High (CMOS)	I_{OH} = −10 μA (Note 2)	V_{CC} − 0.1		V
V_{OL_1}	Logic Low (CMOS)	I_{OH} = 10 μA (Note 2)		0.1	V
V_{OH_2}	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_{OL_2}		I_{OL} = 3 mA		0.4	V
V_{OH_3}	Other Port Pin Drive, WO (open drain) (B_0–B_9, B_{13}, B_{14}, P_0–P_3)	I_{OH} = −1.6 mA	2.4		V
V_{OL_3}		I_{OL} = 0.5 mA		0.4	V
V_{OH4}	ST1 and ST2 Drive	I_{OH} = −6 mA	2.4		V
V_{OL_4}		I_{OL} = 1.6 mA		0.4	V
V_{RAM}	RAM Keep-Alive Voltage	(Note 3)	2.5	V_{CC}	V
I_{OZ}	TRI-STATE® Leakage Current			±5	μA

Note 1: I_{CC_1}, I_{CC_2}, I_{CC_3} measured with no external drive (I_{OH} and I_{OL} = 0, I_{IH} and I_{IL} = 0). I_{CC_1} is measured with $\overline{\text{RESET}}$ = GND. I_{CC_3} is measured with NMI = V_{CC} and A/D inactive. CKI driven to V_{IH1} and V_{IL1} with rise and fall times less than 10 ns. V_{REF} = AGND = GND.

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

Note 3: Test duration is 100 ms.

30 MHz

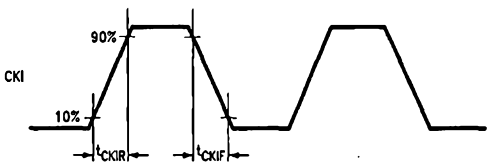
AC Electrical Characteristics

V_{CC} = 5.0V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46164/HPC46104, HPC46064/HPC46004, −40°C to +85°C for HPC36164/HPC36104, HPC36064/HPC36004, −40°C to +105°C for HPC26164/HPC26104, HPC26064/HPC26004, −55°C to +125°C for HPC16164/HPC16104, HPC16064/HPC16004

Symbol	Parameter	Min	Max	Units
f_C = CKI freq.	Operating Frequency	2	30	MHz
$t_{C1} = 1/f_C$	Clock Period	33	500	ns
t_{CKIR} (Note 3)	CKI Rise Time		7	ns
t_{CKIF} (Note 3)	CKI Fall Time		7	ns
$[t_{CKIH}/(t_{CKIH} + t_{CKIL})]100$	Duty Cycle	45	55	%
$t_C = 2/f_C$	Timing Cycle	66		ns
$t_{LL} = \frac{1}{2} t_C - 9$	ALE Pulse Width	24		ns
t_{DC1C2R} (Notes 1, 2)	Delay from CKI Falling Edge to CK2 Rising Edge	0	55	ns
t_{DC1C2F} (Notes 1, 2)	Delay from CKI Falling Edge to CK2 Falling Edge	0	55	ns
$t_{DC1ALER}$ (Notes 1, 2)	Delay from CKI Rising Edge to ALE Rising Edge	0	35	ns
$t_{DC1ALEF}$ (Notes 1, 2)	Delay from CKI Rising Edge to ALE Falling Edge	0	35	ns
$t_{DC2ALER} = \frac{1}{4} t_C + 20$ (Note 2)	Delay from CK2 Rising Edge to ALE Rising Edge		37	ns
$t_{DC2ALEF} = \frac{1}{4} t_C + 20$ (Note 2)	Delay from CK2 Falling Edge to ALE Falling Edge		37	ns
$t_{ST} = \frac{1}{4} t_C - 7$	Address Valid to ALE Falling Edge	9		ns
$t_{VP} = \frac{1}{4} t_C - 5$	Address Hold from ALE Falling Edge	11		ns
$t_{WAIT} = t_C$	Wait State Period	66		ns
$f_{XIN} = f_C/19$	External Timer Input Frequency		1.579	MHz
$t_{XIN} = t_C$	Pulse Width for Timer Inputs	66		ns
$f_{XOUT} = f_C/16$	Timer Output Frequency		1.875	MHz
$f_{MW} = f_C/19$	External MICROWIRE/PLUS Clock Input Frequency		1.579	MHz
$f_U = f_C/8$	External UART Clock Input Frequency		3.75	MHz

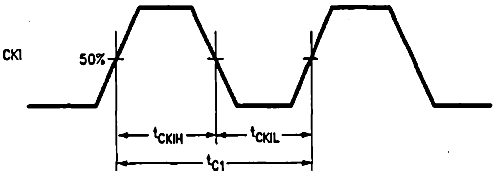
CKI Input Signal Characteristics

Rise/Fall Time



TL/DD/9682-34

Duty Cycle



TL/DD/9682-35

30 MHz

Read Cycle Timing

Symbol	Parameter	Min	Max	Units
$t_{ARR} = \frac{1}{4} t_C - 5$	ALE Falling Edge to $\overline{\text{RD}}$ Falling Edge	12		ns
$t_{RW} = \frac{1}{2} t_C + WS - 14$	$\overline{\text{RD}}$ Pulse Width	85		ns
$t_{DR} = \frac{3}{4} t_C - 15$	Data Hold after Rising Edge of $\overline{\text{RD}}$	0	35	ns
$t_{ACC} = t_C + WS - 32$ (Note 2)	Address Valid to Input Data Valid		100	ns
$t_{RD} = \frac{1}{2} t_C + WS - 39$	$\overline{\text{RD}}$ Falling Edge to Input Data Valid		60	ns
$t_{RDA} = t_C - 5$	$\overline{\text{RD}}$ Rising Edge to Address Valid	61		ns

Write Cycle Timing

Symbol	Parameter	Min	Max	Units
$t_{ARW} = \frac{1}{2} t_C - 5$	ALE Falling Edge to $\overline{\text{WR}}$ Falling Edge	28		ns
$t_{WW} = \frac{3}{4} t_C + WS - 15$	$\overline{\text{WR}}$ Pulse Width	101		ns
$t_{HW} = \frac{1}{4} t_C - 10$	Data Hold after Rising Edge of $\overline{\text{WR}}$	7		ns
$t_V = \frac{1}{2} t_C + WS - 5$	Data Valid before Rising Edge of $\overline{\text{WR}}$	94		ns

Note: Bus Output (Port A) C_L = 100 pF, CK2 Output C_L = 50 pF, other Outputs C_L = 80 pF. AC parameters are tested using DC Characteristics Inputs and non CMOS Outputs. Measurement of AC Specifications is done with external clock driving CKI with 50% duty cycle. The capacitive load on CKO must be kept below 15 pF or AC measurement will be skewed.

Note: WS = t_{WAIT} • number of pre-programmed wait states. Minimum and maximum values are calculated from maximum operating frequency with one (1) wait state pre-programmed.

Note 1: Do not design with this parameter unless CKI is driven with an active signal. When using a passive crystal circuit, CKI or CKO ***should not*** be connected to any external logic since any load (besides the passive components in the crystal circuit) will affect the stability of the crystal unpredictably.

Note 2: These are not directly tested parameters. Therefore the given min/max value cannot be guaranteed. It is, however, derived from measured parameters, and may be used for system design with a very high confidence level.

Note 3: This is guaranteed by design and not tested.

Ready/Hold Timing

Symbol	Parameter	Min	Max	Units
$t_{DAR} = \frac{1}{4} t_C + WS - 50$	Falling Edge of ALE to Falling Edge of RDY		75	ns
$t_{RWP} = t_C$	RDY Pulse Width	100		ns
$t_{SALE} = \frac{3}{4} t_C + 40$	Falling Edge of $\overline{\text{HLD}}$ to Rising Edge of ALE	115		ns
$t_{HWP} = t_C + 10$	$\overline{\text{HLD}}$ Pulse Width	110		ns
$t_{HAD} = \frac{3}{4} t_C + 85$	Rising Edge on $\overline{\text{HLD}}$ to Rising Edge on $\overline{\text{HLDA}}$		160	ns
$t_{HAE} = t_C + 85$	Falling Edge on $\overline{\text{HLD}}$ to Falling Edge on $\overline{\text{HLDA}}$		151*	ns
t_{BF}	Data Valid after Falling Edge on $\overline{\text{HLDA}}$	0		ns
$t_{BE} = \frac{1}{2} t_C$	Bus Enable from Rising Edge of $\overline{\text{HLDA}}$	33		ns

***Note:** t_{HAE} may be as long as ($3t_C$ + 4ws + $72t_C$ + 90) depending on which instruction is being executed, the addressing mode and number of wait states. t_{HAE} maximum value is for the optimal case.

30 MHz

MICROWIRE/PLUS Timing

Symbol	Parameter	Min	Max	Units
t_{UWS} Master Slave	MICROWIRE Setup Time	 100 20		ns
t_{UWH} Master Slave	MICROWIRE Hold Time	 20 50		ns
t_{UWV} Master Slave	MICROWIRE Output Valid Time		 50 150	ns

UPI Read/Write Timing

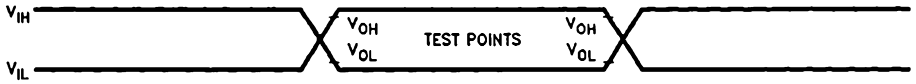
Symbol	Parameter	Min	Max	Units
t_{UAS}	Address Setup Time to Falling Edge of $\overline{\text{URD}}$	10		ns
t_{UAH}	Address Hold Time from Rising Edge of $\overline{\text{URD}}$	10		ns
t_{RPW}	$\overline{\text{URD}}$ Pulse Width	100		ns
t_{OE}	$\overline{\text{URD}}$ Falling Edge to Output Data Valid	0	60	ns
t_{OD}	Rising Edge of $\overline{\text{URD}}$ to Output Data Invalid (Note 4)	5	35	ns
t_{DRDY}	$\overline{\text{RDRDY}}$ Delay from Rising Edge of $\overline{\text{URD}}$		70	ns
t_{WDW}	$\overline{\text{UWR}}$ Pulse Width	40		ns
t_{UDS}	Input Data Valid before Rising Edge of $\overline{\text{UWR}}$	10		ns
t_{UDH}	Input Data Hold after Rising Edge of $\overline{\text{UWR}}$	15		ns
t_A	$\overline{\text{WRRDY}}$ Delay from Rising Edge of $\overline{\text{UWR}}$		70	ns

Note: Bus Output (Port A) C_L = 100 pF, CK2 Output C_L = 50 F, other Outputs C_L = 80 pF.

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

Note 4: Guaranteed by design.

Input and Output for AC Tests



TL/DD/9682-40

30 MHz

A/D Converter Specifications

V_{CC} = 5V ±10% unless otherwise specified, T_A = 0°C to +70°C for HPC46164/HPC46104, −40°C to +85°C for HPC36164/HPC36104, −40°C to +105°C for HPC26164/HPC26104, −55°C to +125°C for HPC16164/HPC16104

Symbol	Parameter	Min	Max	Units
	Resolution		8	bits
f_{CCLK}	Clock Frequency (Note 4)	0.1	1.6	MHz
$t_{CON} = 8.5/f_{CCLK}$	Conversion Time (Note 3)	5.3		μs
V_{REF}	Reference Voltage Input (AGND = 0V)	3.0	V_{CC}	V
	Total Unadjusted Error (Note 1) (V_{REF} = 5.000V)		±½	LSB
R_{VREF}	Reference Input Resistance (Note 5)	1.6	4.8	kΩ
	DC Common Mode Error		±¼	LSB
	Power Supply Sensitivity (V_{REF} = 5.000V, V_{CC} = 5V ±10%)		±¼	LSB
	Voltage Reference Tolerance (V_{REF})		TBD	LSB
	Port D Input Capacitance (Note 5)		35	pF
	Analog Input Voltage Range (Note 2)	GND − 0.05	V_{CC} + 0.05	V
	On Channel Leakage		1	μA
	Off Channel Leakage		1	μA

Note 1: Total unadjusted error includes offset, full-scale, and multiplexer errors.

Note 2: 8 single-ended or 4 differential channels. Inherent sample and hold for single-ended inputs (GND = Pin 62).

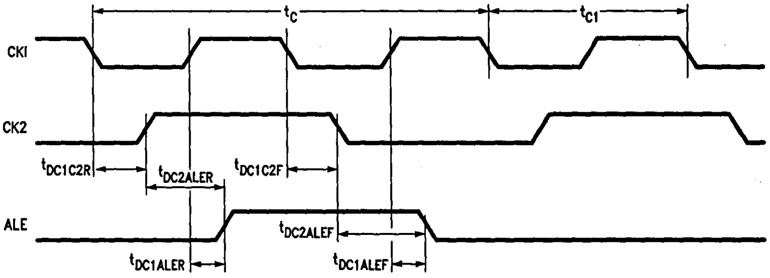
Note 3: Conversion time does not include sample/hold time. Sample and hold time is $2/f_{CCLK}$.

Note 4: Clock supplied to A/D converter is derived from CKI. See A/D description for details.

Note 5: This is guaranteed by design and not tested.

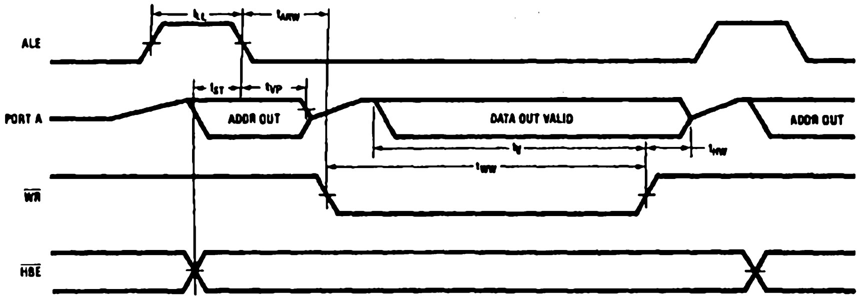
Timing Waveforms

CKI, CK2, ALE Timing Diagram



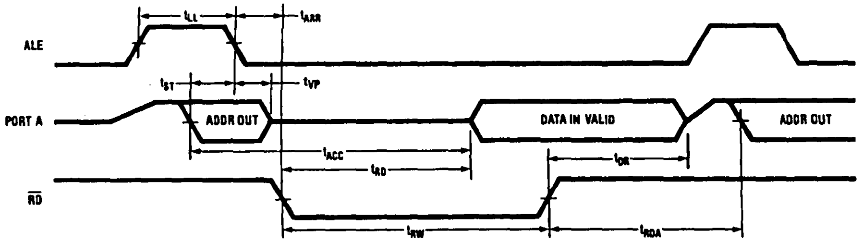
TL/DD/9682-2

Timing Waveforms (Continued)



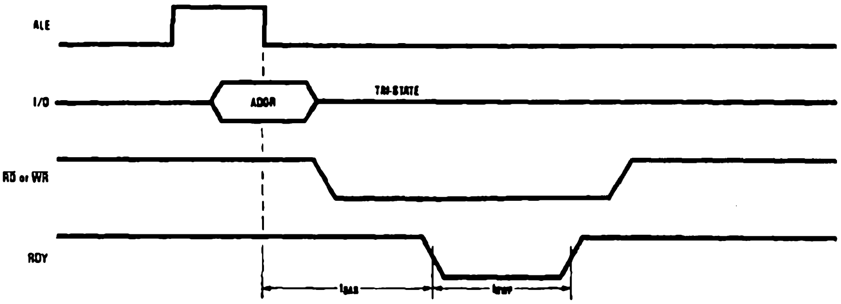
TL/DD/9682-3

FIGURE 1. Write Cycle



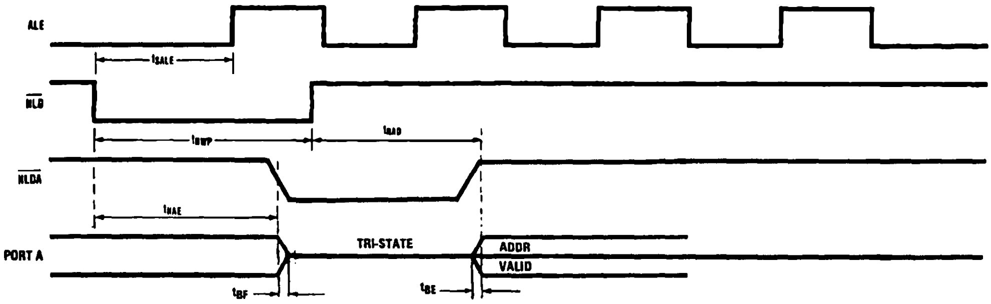
TL/DD/9682-4

FIGURE 2. Read Cycle



TL/DD/9682-5

FIGURE 3. Ready Mode Timing

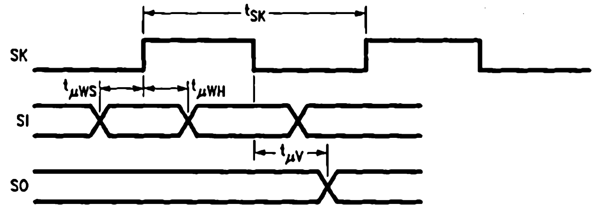


TL/DD/9682-6

FIGURE 4. Hold Mode Timing

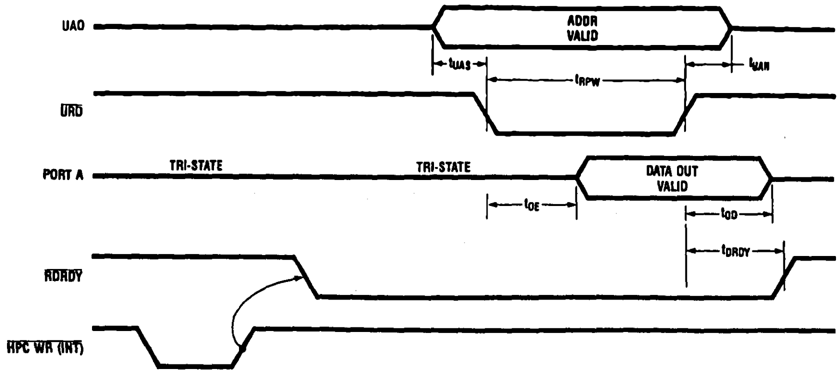
HPC16164/26164/36164/46164/16104/26104/36104/46104/16064/26064/36064/46064/16004/26004/36004/46004

Timing Waveforms (Continued)



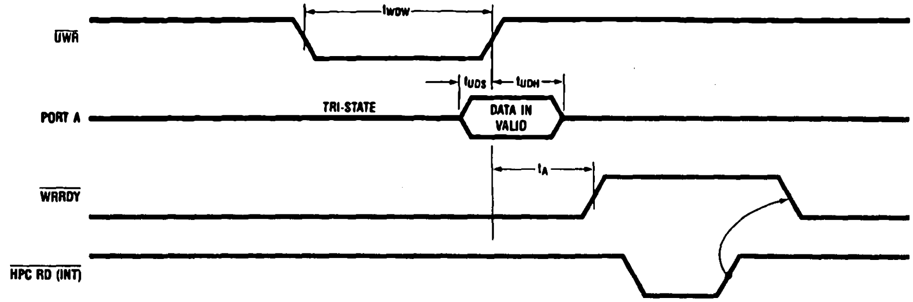
TL/DD/9682-39

FIGURE 5. MICROWIRE Setup/Hold Timing



TL/DD/9682-9

FIGURE 6. UPI Read Timing



TL/DD/9682-10

FIGURE 7. UPI Write Timing

Pin Descriptions

The HPC16164 is available in 68-pin PLCC, LCC, LDCC, PGA, and TapePak 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:	T310	Timer3 I/O Pin
B5:	SO	MICROWIRE/PLUS Output
B6:	SK	MICROWIRE/PLUS Clock (Input or Output)
B7:	$\overline{HLDA}$	Hold Acknowledge Output
B8:	TS0	Timer Synchronous Output
B9:	TS1	Timer Synchronous Output
B10:	UA0	Address 0 Input for UPI Mode
B11:	$\overline{WRRDY}$	Write Ready Output for UPI Mode
B12:		
B13:	TS2	Timer Synchronous Output
B14:	TS3	Timer Synchronous Output
B15:	$\overline{RDRDY}$	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:	$\overline{WR}$	Write Output
B12:	$\overline{HBE}$	High Byte Enable Output/Input (sampled at reset)
B15:	$\overline{RD}$	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/$\overline{URD}$
I3:	INT3	Maskable Interrupt/Input Capture/$\overline{UWR}$
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 or as analog channel inputs for the A/D converter. These functions of Port D are mutually exclusive and under the control of software.

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

$\overline{WO}$ This is an active low open drain output that signals an illegal situation has been detected by the Watch Dog logic.

ST1 Bus Cycle Status Output: indicates first opcode fetch.

ST2 Bus Cycle Status Output: indicates machine states (skip, interrupt and first instruction cycle).

$\overline{RESET}$ is an active low input that forces the chip to restart and sets the ports in a TRI-STATE® mode.

RDY/$\overline{HLD}$ has two uses, 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.

VREF A/D converter reference voltage input.

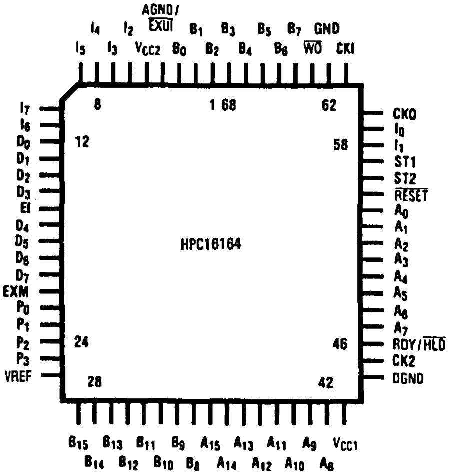
EXM External memory enable (active high) disables internal ROM and maps it to external memory.

EI External interrupt with vector address FFF1:FFF0. (Rising/falling edge or high/low level sensitive). Alternately can be configured as 4th input capture.

AGND/$\overline{EXUI}$ has two uses, selected by a software bit. It can be an external active low interrupt which is internally OR'ed with the UART interrupt with vector address FFF3:FFF2 or it can be the analog ground for the A/D converter.

Connection Diagrams

Plastic, Leadless and Leaded Chip Carriers

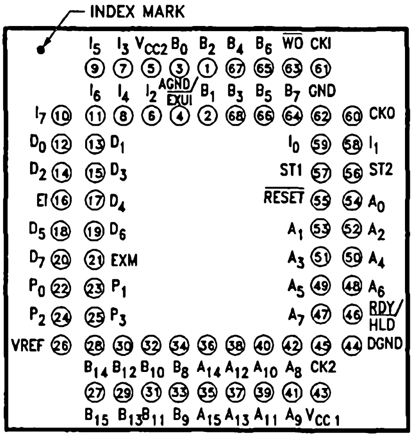

I4 I2 AGND/EXUI B1 B3 B5 B7 GND
I5 I3 VCC2 B0 B2 B4 B6 WO CKI
8
1 68
62
12
58
HPC16164
24
46
28
42
I7 I6 D0 D1 D2 D3 EI D4 D5 D6 D7 EXM P0 P1 P2 P3 VREF
CKO I0 I1 ST1 ST2 RESET A0 A1 A2 A3 A4 A5 A6 A7 RDY/HLD CK2 DGND
B15 B13 B11 B9 A15 A13 A11 A9 VCC1
B14 B12 B10 B8 A14 A12 A10 A8

TL/DD/9682-11

Top View

Order Number HPC16164E, EL or V
See NS Package Number E68B, EL68A or V68A

Pin Grid Array Pinout

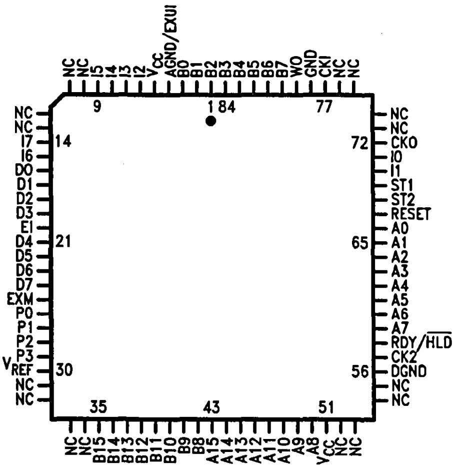


TL/DD/9682-12

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

Order Number HPC16164U
See NS Package Number U68A

TapePak Package


NC NC I5 I4 I3 I2 VCC AGND/EXUI B0 B1 B2 B3 B4 B5 B6 B7 WO GND CKI NC NC
9
1 84
77
NC NC I7 I6 D0 D1 D2 D3 EI D4 D5 D6 D7 EXM P0 P1 P2 P3 VREF NC NC
14
21
30
NC NC CKO I0 I1 ST1 ST2 RESET A0 A1 A2 A3 A4 A5 A6 A7 RDY/HLD CK2 DGND NC NC
72
65
56
35
43
51
NC NC B15 B14 B13 B12 B11 B10 B9 B8 A15 A14 A13 A12 A11 A10 A9 A8 VCC NC NC

TL/DD/9682-36

Top View

Order Number HPC16164TP
Available in TapePak

Operating Modes

To offer the user a variety of I/O and expanded memory options, the HPC16164 and HPC16104 have four operating modes. The ROMless HPC16104 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 HPC16164 is C000 to FFFF (16k bytes). The HPC16104 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 HPC16104 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 HPC16104 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, $\overline{\text{RD}}$, $\overline{\text{WR}}$ and $\overline{\text{HBE}}$. The High Byte Enable pin ($\overline{\text{HBE}}$) is used in 16-bit mode to select high order memory bytes. The $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals are only generated if the selected address is off-chip. The 8-bit mode is selected by pulling $\overline{\text{HBE}}$ high at reset. If $\overline{\text{HBE}}$ 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 HPC16164 and HPC16104.

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.

HPC16164/HPC16064 Operating Modes

SINGLE CHIP NORMAL MODE

In this mode, the HPC16164/HPC16064 functions as a self-contained microcomputer (see *Figure 11*) 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 ($\overline{\text{WO}}$) 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.

EXPANDED NORMAL MODE

The Expanded Normal mode of operation enables the HPC16164 to address external memory in addition to the on-chip ROM and RAM (see Table II). 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 HPC16164 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 HPC16164 (see Table II). 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".

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.

TABLE II. HPC16164 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.

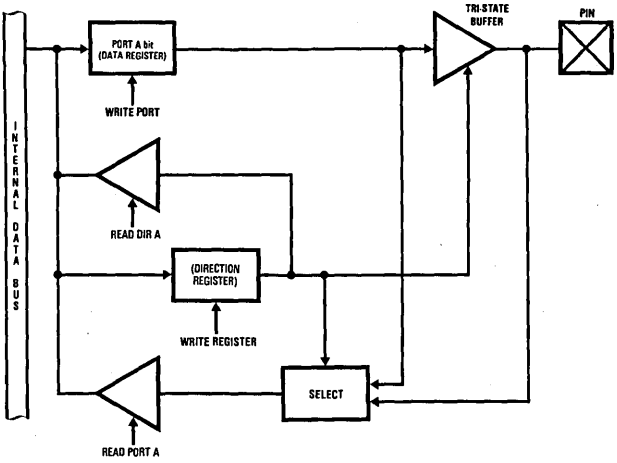
Ports A & B

The highly flexible A and B ports are similarly structured. The Port A (see *Figure 7*), consists of a data register and a direction register. Port B (see *Figures 8, 9* and *10*) 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.

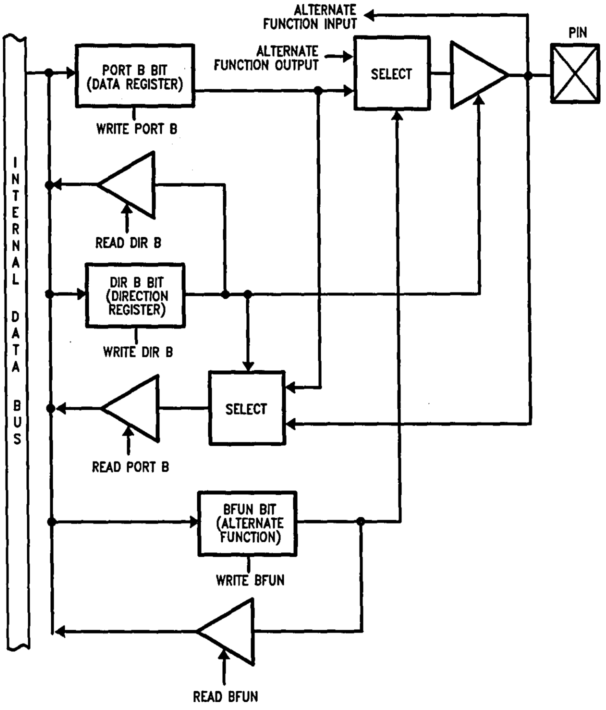
A write operation to a port pin configured as an input causes the value to be written into the data register, a read operation returns the value of the pin. Writing to port pins configured as outputs causes the pins to have the same value, reading the pins returns the value of the data register.

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.



TL/DD/9682-13

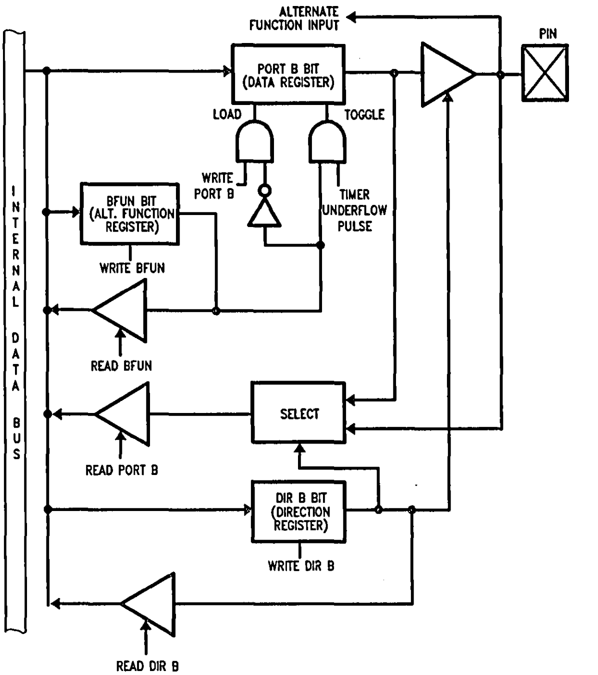
FIGURE 7. Port A: I/O Structure



TL/DD/9682-14

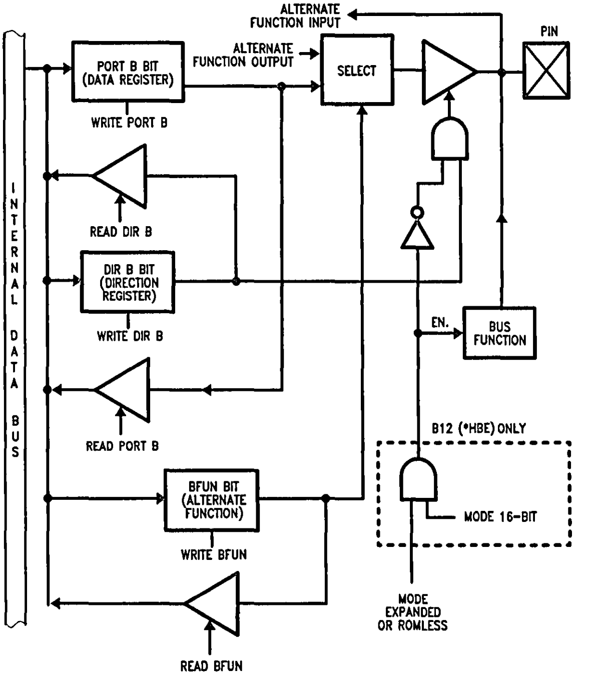
FIGURE 8. Structure of Port B Pins B0, B1, B2, B5, B6 and B7 (Typical Pins)

Ports A & B (Continued)



TL/DD/9682–15

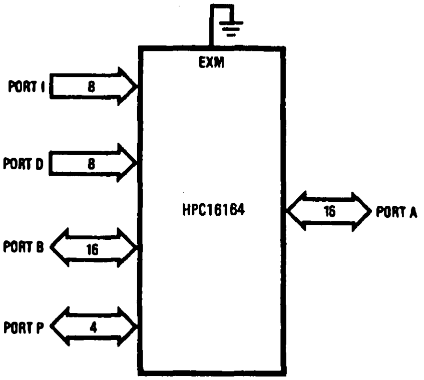
FIGURE 9. Structure of Port B Pins B3, B4, B8, B9, B13 and B14 (Timer Synchronous Pins)



TL/DD/9682–16

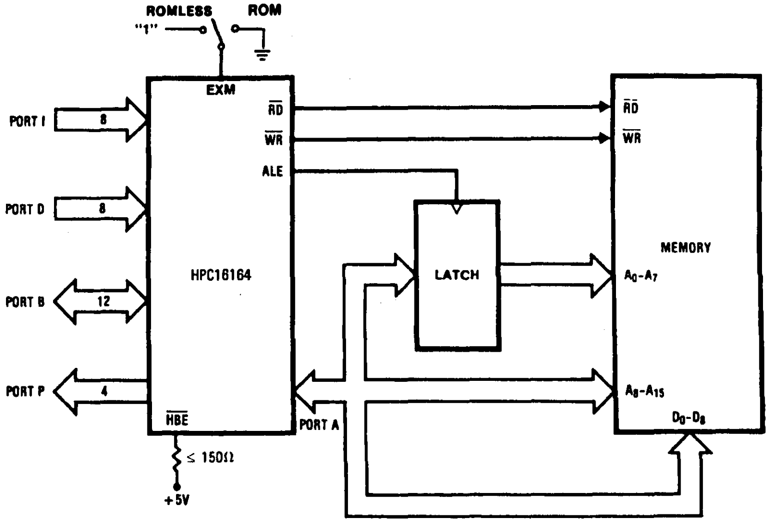
FIGURE 10. Structure of Port B Pins B10, B11, B12 and B15 (Pins with Bus Control Roles)

HPC16164 Operating Modes (Continued)



TL/DD/9682-17

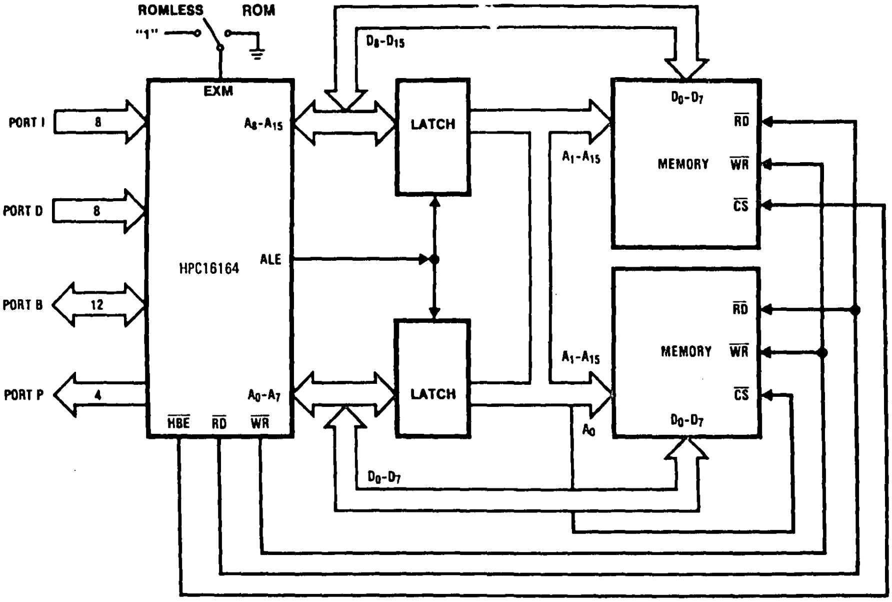
FIGURE 11. Single-Chip Mode



TL/DD/9682-18

FIGURE 12. 8-Bit External Memory

HPC16164 Operating Modes (Continued)



TL/DD/9682-19

FIGURE 13. 16-Bit External Memory

HPC16104/HPC16004 Operating Modes

EXPANDED ROMLESS MODE (HPC16104/HPC16004)

Because the HPC16104 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 HPC16104/HPC16004 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 III. HPC16104 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 HPC16104 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 ⅔ f_C max.

The HPC16164 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 HPC16164: 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, ail on-board RAM, registers and I/O are unaffected.

HALT MODE

The HPC16164 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 HPC16164 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 HPC16164 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

Power Save Modes (Continued)

mode, the processor is returned to full operation by the RESET or NMI inputs, but without waiting for oscillator stabilization. A timer T0 overflow will also cause the HPC16164 to resume normal operation.

HPC16164 Interrupts

Complex interrupt handling is easily accomplished by the HPC16164's vectored interrupt scheme. There are eight possible interrupt sources as shown in Table IV.

TABLE IV. Interrupts

Vector Address	Interrupt Source	Arbitration Ranking
$FFFF:FFFE	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 by on-board peripherals or external on EXUI	6
$FFF1:FFF0	External interrupt on EI pin	7

Interrupt Arbitration

The HPC16164 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 IV. 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 RESET, which is serviced immediately.

RESET and 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 (EXUI) is shared with the on-board peripherals, UART and A/D. The 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 and disable the A/D function by resetting the ADEN bit in the A/D control register #3 (CR3). To select the on-board peripherals interrupt, leave this pin floating or tie it high if the A/D function is disabled. If the A/D function is enabled, this pin becomes the analog ground (AGND).

Interrupt Control Registers

The HPC16164 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)

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 HPC16164 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.

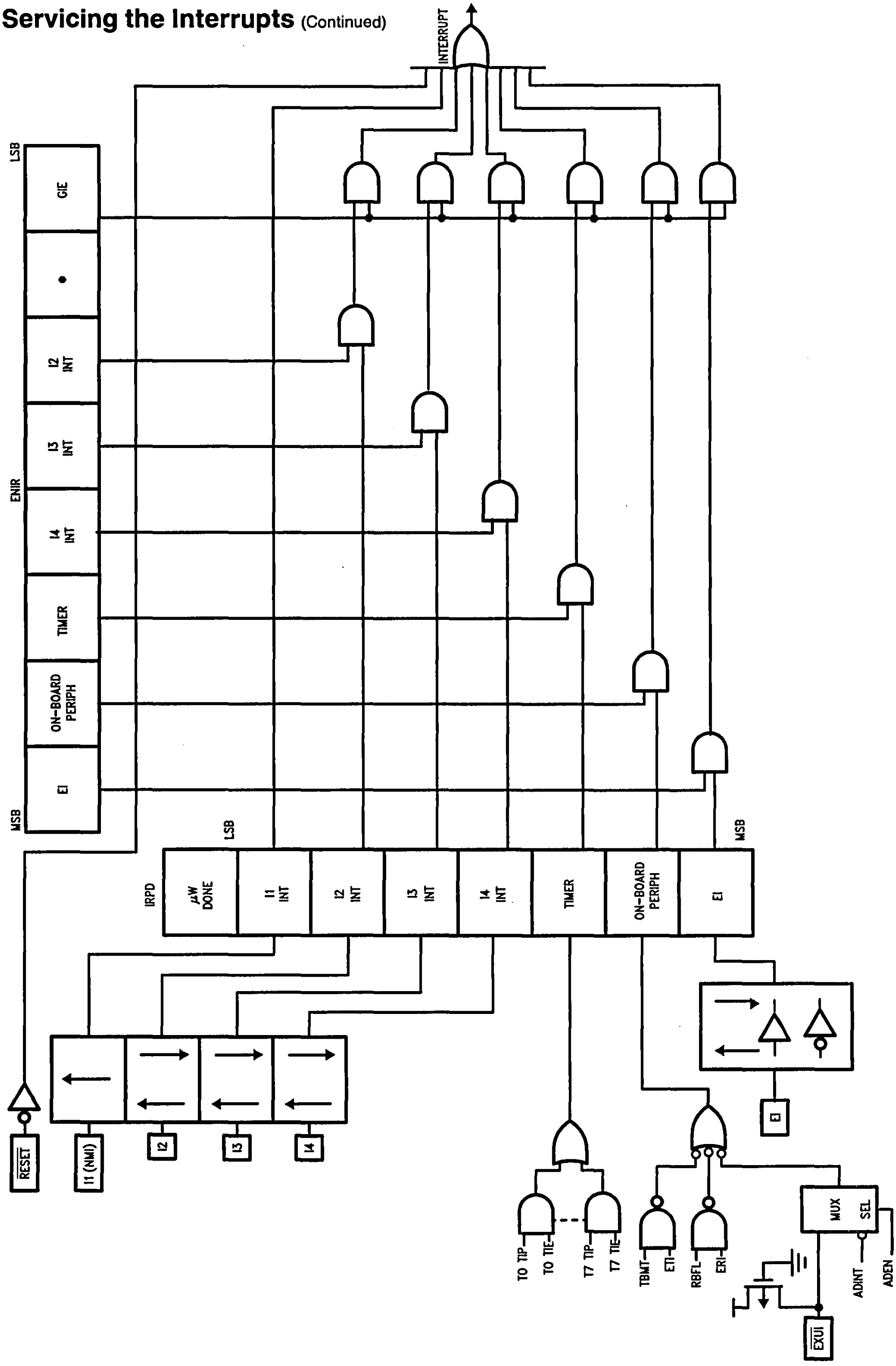
Servicing the Interrupts

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 14* shows the Interrupt Enable Logic.

Reset

The RESET input initializes the processor and sets ports A and B in the TRI-STATE condition and Port P in the LOW state. 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 HPC16104.

Servicing the Interrupts (Continued)



TL/DD/9682-20

FIGURE 14. Block Diagram of Interrupt Logic

Timer Overview

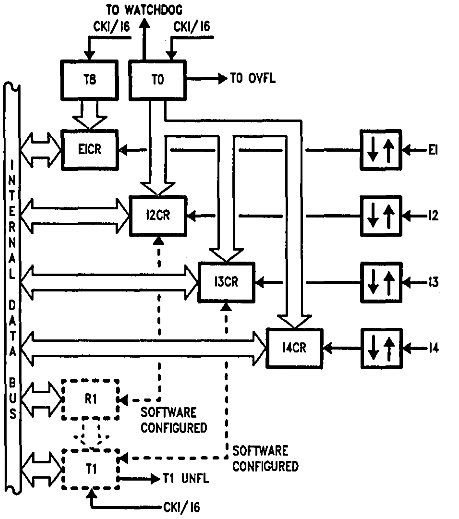
The HPC16164 contains a powerful set of flexible timers enabling the HPC16164 to perform extensive timer functions; not usually associated with microcontrollers.

The HPC16164 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 15*).

The HPC16164 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 16*).

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.

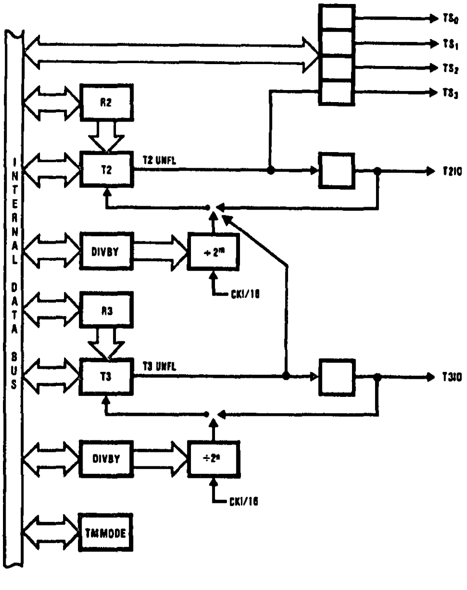


TL/DD/9682-21

FIGURE 15. Timers T0, T1 and T8 with Four Input Capture Registers

SYNCHRONOUS OUTPUTS

The flexible timer structure of the HPC16164 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 16*).

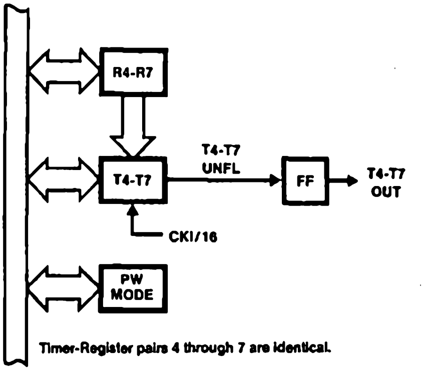


TL/DD/9682-22

FIGURE 16. Timers T2–T3 Block

Timer Overview (Continued)

Timer/register pairs 4–7 form four identical units which can generate synchronous outputs on port P (see *Figure 17*). 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/9682-23

FIGURE 17. Timers T4–T7 Block

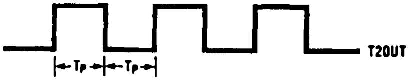
Timer Registers

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 HPC16164.

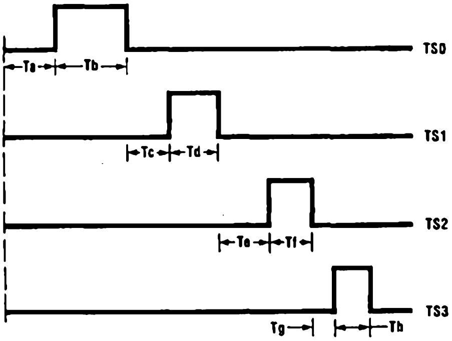
Frequencies can be generated by using the timer/register pairs. A square wave is generated when the register value is a constant. The duty cycle can be controlled simply by changing the register value.



TL/DD/9682-24

FIGURE 18. 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 19* is an example of synchronous pulse train generation.



TL/DD/9682-25

FIGURE 19. Synchronous Pulse Generation

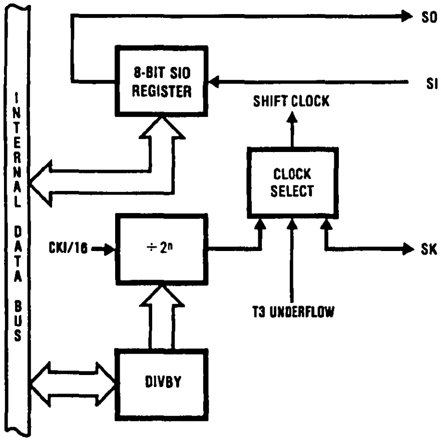
Watchdog Logic

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 ($\overline{\text{WO}}$) pin low. The $\overline{\text{WO}}$ pin is an open drain output and can be connected to the $\overline{\text{RESET}}$ 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 20*). 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.



TL/DD/9682-26

FIGURE 20. MICROWIRE/PLUS

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

MICROWIRE/PLUS Operation

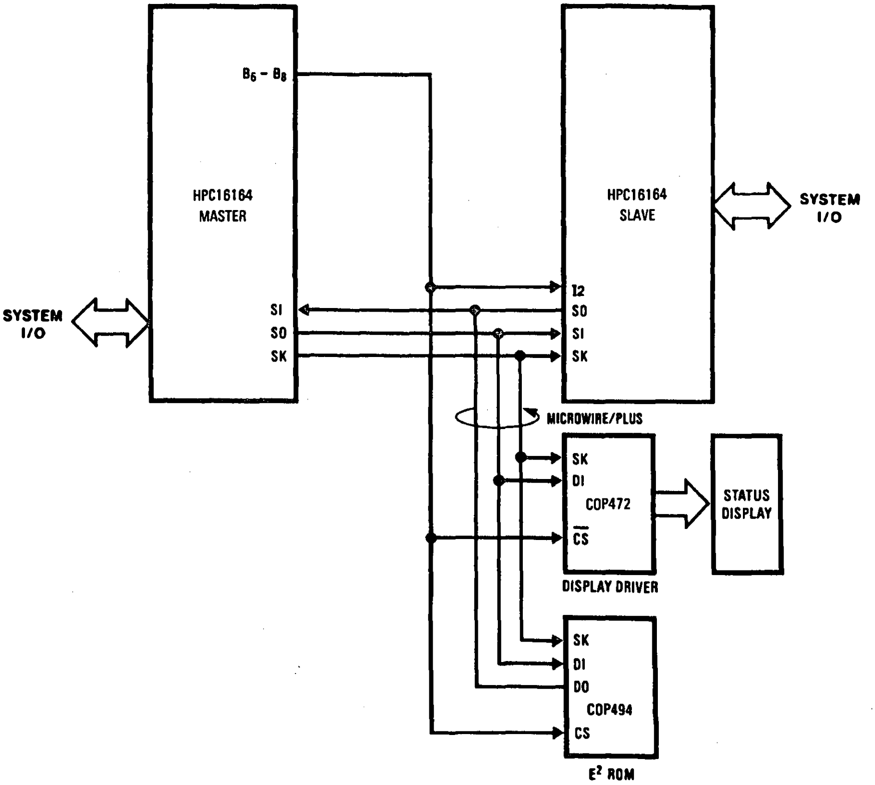
The HPC16164 can enter the MICROWIRE/PLUS mode as the master or a slave. A control bit in the IRCD register determines whether the HPC16164 is the master or slave. The shift clock is generated when the HPC16164 is configured as a master. An externally generated shift clock on the SK pin is used when the HPC16164 is configured as a slave. When the HPC16164 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 15 selectable steps from 64 Hz to 1 MHz with CKI at 16.0 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 21 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 HPC16164 microcontrollers interconnected to other MICROWIRE peripherals. HPC16164 #1 is set up as the master and initiates all data transfers. HPC16164 #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 HPC16164 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/9682-27

FIGURE 21. MICROWIRE/PLUS Application

HPC16164 UART

The HPC16164 contains a software programmable UART. The UART (see *Figure 22*) 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.

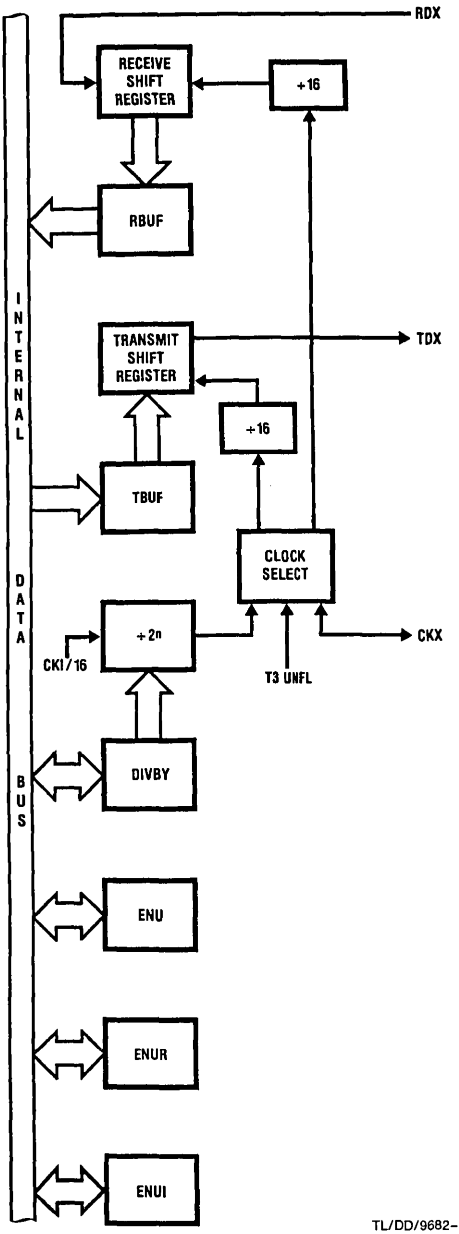
The baud rate clock for the Receiver and Transmitter can be selected for either an internal or external source using two bits in the ENUI register. The internal baud rate is programmed by the DIVBY register. The baud rate may be selected from a range of 8 Hz to 128 kHz in binary steps or T3 underflow. 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 HPC16164 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 HPC16164 UART features a Wake-up Mode of operation. This mode of operation enables the HPC16164 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.

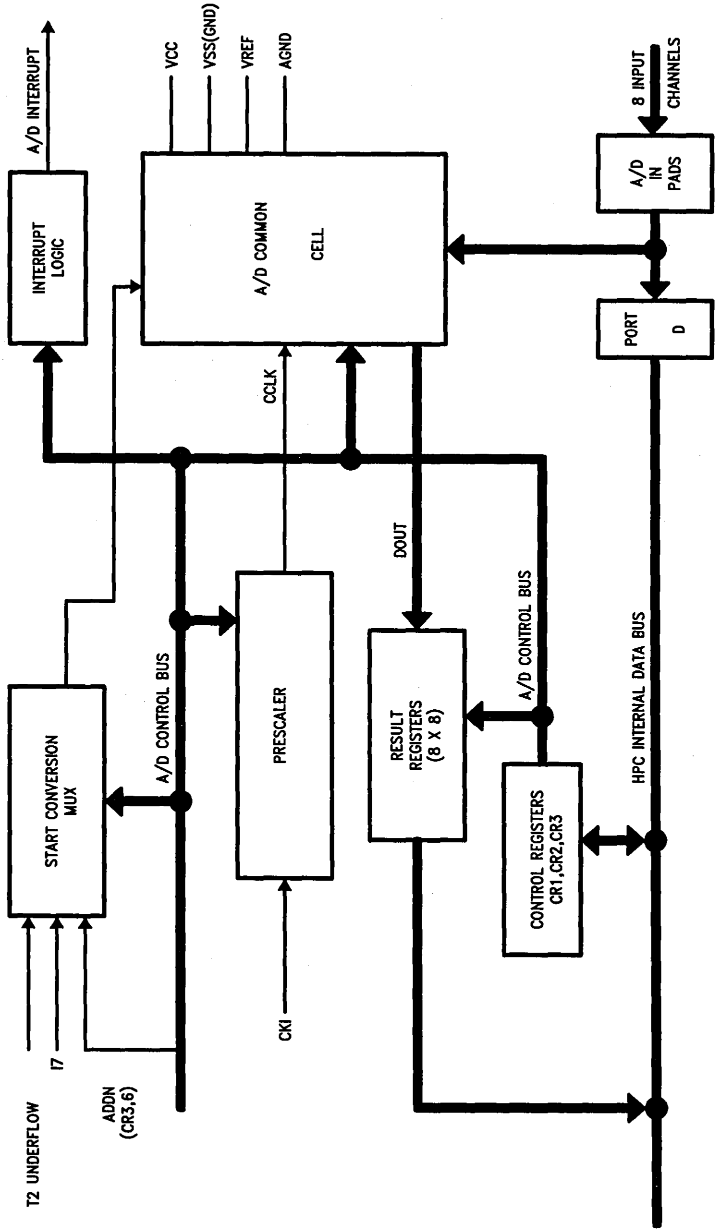
The UART monitors the communication stream looking for addresses. When the data word with the ninth bit set is received, the UART signals the HPC16164 with an interrupt. The processor then examines the content of the receiver buffer to decide whether it has been addressed and whether to accept subsequent data.



TL/DD/9682-28

FIGURE 22. UART Block Diagram

A/D Converter Operation



TL/DD/9682-29

FIGURE 23. A/D Block Diagram

A/D Converter Operation (Continued)

The HPC16164 has an on-board eight-channel 8-bit Analog to Digital converter. Conversion is performed using a successive approximation technique. The A/D converter cell can operate in single-ended mode where the input voltage is applied across one of the eight input channels (D0–D7) and AGND or in differential mode where the input voltage is applied across two adjacent input channels. The A/D converter will convert up to eight channels in single-ended mode and up to four channel-pairs in differential mode.

OPERATING MODES

The operating modes of the converter are selected by 4 bits called ADMODE (CR2.4–7). Associated with the eight input channels in single-ended mode are eight result registers, one for each channel. The A/D converter can be programmed by software to convert on any specific channel storing the result in the result register associated with that channel. It can also be programmed to stop after one conversion or to convert continuously. If a brief history of the signal on any specific input channel is required, the converter can be programmed to convert on that channel and store the consecutive results in each of the result registers before stopping. As a final configuration in single-ended mode, the converter can be programmed to convert the signal on each input channel and store the result in its associated result register continuously.

Associated with each even-odd pair of input channels in differential mode of operation are four result register-pairs. The A/D converter performs two conversions on the selected pair of input channels. One conversion is performed assuming the positive connection is made to the even channel and the negative connection is made to the following odd channel. This result is stored in the result register associated with the even channel. Another conversion is performed assuming the positive connection is made to the odd channel and the negative connection is made to the preceding even channel. This result is stored in the result register associated with the odd channel. This technique does not require that the programmer know the polarity of the input signal. If the even channel result register is non-zero (meaning the odd channel result register is zero), then the input signal is positive with respect to the odd channel. If the odd channel result register is non-zero (meaning the even channel result register is zero), then the input signal is positive with respect to the even channel.

The same operating modes for single-ended operation also apply when the inputs are taken from channel-pairs in differential mode. The programmer can configure the A/D to convert on any selected channel-pair and store the result in its associated result register-pair then stop. The A/D can also be programmed to do this continuously. Conversion can also be done any channel-pair storing the result into four result register-pairs for a history of the differential input. Finally, all input channel-pairs can be converted continuously.

The final mode of operation suppresses the external address/data bus activity during the single conversion modes. These quiet modes of operation utilize the RDY function of the HPC Core to insert wait states in the instruction being executed in order to limit digital noise in the environment due to external bus activity when addressing external memory. The overall effect is to increase the accuracy of the A/D.

CONTROL

The conversion clock supplied to the A/D converter can be selected by three bits in CR1 used as a prescaler on CKI. These bits can be used to ensure that the A/D is clocked as fast as possible when different external crystal frequencies are used. Controlling the starting of conversion cycles in each of the operating modes can be done by four different methods. The method is selected by two bits called SC (CR3.0–1). Conversion cycles can be initiated through software by resetting a bit in a control register, through hardware by an underflow of Timer T2, or externally by a rising or falling edge of a signal input on I7.

INTERRUPTS

The A/D converter can interrupt the HPC when it completes a conversion cycle if one of the non-continuous modes has been selected. If one of the cycle modes was selected, then the converter will request an interrupt after eight conversions. If one of the one-shot modes was selected, then the converter will request an interrupt after every conversion. When this interrupt is generated, the HPC vectors to the on-board peripheral interrupt vector location at address FFF2. The service routine must then determine if the A/D converter requested the interrupt by checking the A/D done flag which doubles as the A/D interrupt pending flag.

REGISTER MAP

The A/D converter status and control registers and the result registers are detailed as follows:

Control Register #1 (CR1)

	Prescaler(3)	Result Reg pntr(4)
msb		lsb

byte at location 0100

Result Register pointer—These four bits are read/only by the software. In all the operating modes that are single channel or single channel-pair, this pointer gets the value of the Channel Select bits (CR2.0–3) and remains constant. In the operating modes that work on multiple channels or multiple channel-pairs, this pointer gets initialized to zero and will change to reflect the current channel that is being converted (default value on power-up is 0000).

Prescaler—These three bits are used to select the clock (CCLK) supplied to the SAR in the A/D converter cell. The maximum clock that can be supplied is 1.67 MHz and the minimum is 100 kHz. Therefore, these bits can be used to ensure that the A/D is clocked as fast as posible at different external crystal frequencies.

000	=	stop the clock (CCLK) to the A/D cell (default value on power-up)
011	=	use CKI/4 to allow max CKI of 6.66 MHz
010	=	use CKI/8 to allow max CKI of 13.33 MHz
111	=	use CKI/12 to allow max CKI of 20 MHz
101	=	use CKI/16 to allow max CKI of 26.66 MHz
001	=	use CKI/20 to allow max CKI of 33.33 MHz
110	=	use CKI/24 to allow max CKI of 40 MHz
100	=	use CKI/32 to allow max CKI of 53.4 MHz

Note: All remaining unused bits in this control register are UNDEFINED and not available for use by the program.

A/D Converter Operation (Continued)

Control Register #2 (CR2)

ADMODE(4)	Channel Select(4)
msb	lsb

byte at location 0102

ADMODE—These four bits are used to select the mode of operation for the A/D converter as described in OPERATING MODES.

0000 = single-ended, single channel, single result register, one-shot (default value on power-up)

0001 = single-ended, single channel, single result register, continuous

0010 = single-ended, single channel, multiple result registers, stop after 8

0011 = single-ended, multiple channel, multiple result registers, continuous

0100 = differential, single channel-pair, single result register-pair, one-shot

0101 = differential, single channel-pair, single result register-pair, continuous

0110 = differential, single channel-pair, multiple result register-pairs, stop after 4 pairs

0111 = differential, multiple channel-pair, multiple result register-pairs, continuous

Channel Select—These four bits are used to select the channel on which to initiate conversions.

Single-ended

x000 = Convert on Channel 0 (Input Port D.0)

x001 = Convert on Channel 1 (Input Port D.1)

x010 = Convert on Channel 2 (Input Port D.2)

x011 = Convert on Channel 3 (Input Port D.3)

x100 = Convert on Channel 4 (Input Port D.4)

x101 = Convert on Channel 5 (Input Port D.5)

x110 = Convert on Channel 6 (Input Port D.6)

x111 = Convert on Channel 7 (Input Port D.7)

Differential

x000 = Convert on Channel-Pair 0,1

x010 = Convert on Channel-Pair 2,3

x100 = Convert on Channel-Pair 4,5

x110 = Convert on Channel-Pair 6,7

Control Register #3 (CR3)

	ADDN			ADIE	ADEN	SC Mode (2)
msb						lsb

byte at location 0106

SC mode—These two bits are used to select the mode for starting a conversion cycle.

00 = A conversion cycle is initiated by resetting the A/D done flag (ADDN) (default value on power-up).

01 = A conversion cycle is initiated by an underflow of Timer T2.

10 = A conversion cycle is initiated by the falling edge of the signal on input I7.

11 = A conversion cycle is initiated by the rising edge of the signal on input I7.

ADEN—Setting this bit enables pin 4 to be the analog ground, AGND. Resetting this bit returns pin 4 as $\overline{\text{EXUI}}$ (reset on power-up).

ADIE—This is the A/D interrupt enable bit. (reset on power-up).

ADDN—This bit is the A/D done flag and doubles as the A/D interrupt pending flag. If one of the one-shot modes was selected using ADMODE(=xx00) and control was selected as SC = 00, then this bit must be reset by software to initiate the conversion and is set by the hardware at the end of one conversion. If one of the cycle modes was selected using ADMODE(=xx10) and control was selected as SC = 00, then this bit must be reset by software to initiate the conversion cycle and is not set by the hardware until the end of one conversion cycle. If any of the continuous modes were selected and control was selected as SC = 00, then this bit must be reset by software to initiate the conversions and is not set by the hardware until the clock to the A/D cell is stopped by selecting the value 000 for the prescaler. In all other control selections, this bit has no effect on the initiation of conversions but is still necessary for proper interrupt operation. The ADDN flag must also be reset for the quiet modes to work properly (set on power-up).

Note: All remaining unused bits in this control register are UNDEFINED and not available for use by the program. Also, all result register contents are UNDEFINED on power-up.

Universal Peripheral Interface

The Universal Peripheral Interface (UPI) allows the HPC16164 to be used as an intelligent peripheral to another processor. The UPI could thus be used to tightly link two HPC16164's and set up systems with very high data exchange rates. Another area of application could be where a HPC16164 is programmed as an intelligent peripheral to a host system such as the Series 32000® microprocessor. *Figure 24* illustrates how a HPC16164 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 ($\overline{\text{URD}}$), a Write Strobe ($\overline{\text{UWR}}$), a Read Ready Line ($\overline{\text{RDRDY}}$), a Write Ready Line ($\overline{\text{WRRDY}}$) and one Address Input (UA0). The data bus can be either eight or sixteen bits wide.

The $\overline{\text{URD}}$ and $\overline{\text{UWR}}$ inputs may be used to interrupt the HPC16164. The $\overline{\text{RDRDY}}$ and $\overline{\text{WRRDY}}$ outputs may be used to interrupt the host processor.

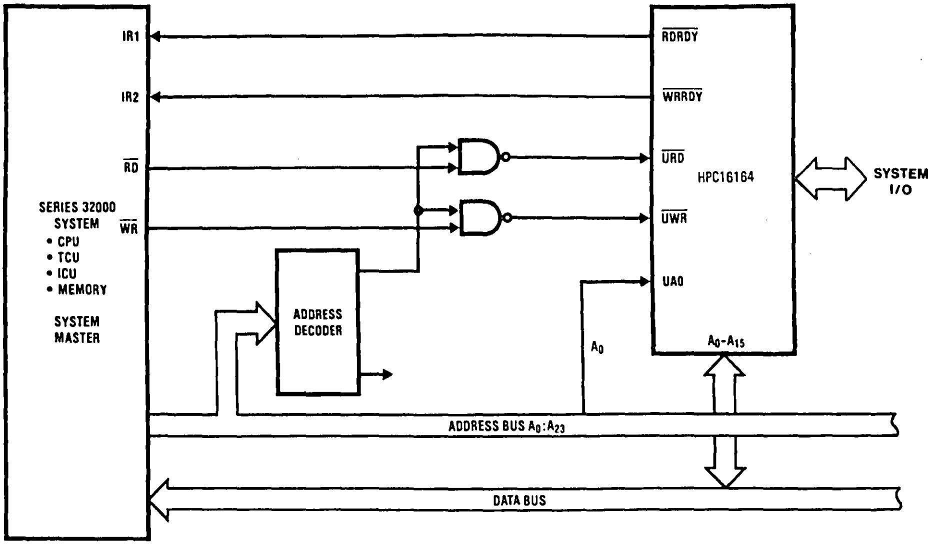
The UPI contains an Input Buffer (IBUF), an Output Buffer (OBUF) and a Control Register (UPIC). In the UPI mode, port A on the HPC16164 is the data bus. UPI can only be used if the HPC16164 is in the Single-Chip mode.

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 HPC16164 supports shared memory access with two pins. The pins are the RDY/$\overline{\text{HLD}}$ input pin and the $\overline{\text{HLDA}}$ output pin. The user can software select either the Hold or Ready function by the state of a control bit. The $\overline{\text{HLDA}}$ output is multiplexed onto port B.

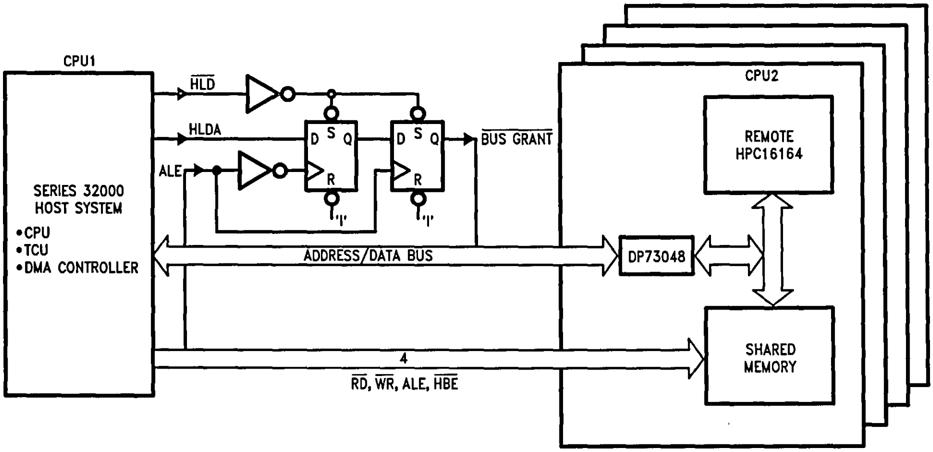
The host uses DMA to interface with the HPC16164. The host initiates a data transfer by activating the $\overline{\text{HLD}}$ input of the HPC16164. In response, the HPC16164 places its system bus in a TRI-STATE Mode, freeing it for use by the host. The host waits for the acknowledge signal ($\overline{\text{HLDA}}$) from the HPC16164 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 HPC16164 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 25* illustrates an application of the shared memory interface between the HPC16164 and a Series 32000 system.



TL/DD/9682-30

FIGURE 24. HPC16164 as a Peripheral: (UPI Interface to Series 32000 Application)



TL/DD/9682-31

FIGURE 25. Shared Memory Application: HPC16164 Interface to Series 32000 System

Memory

The HPC16164 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 HPC16164 uses memory-mapped organization to support registers, I/O and on-chip peripheral functions.

The HPC16164 memory address space extends to 64 kbytes and registers and I/O are mapped as shown in Table V.

TABLE V. HPC16164 Memory Map

FFFF:FFF0	Interrupt Vectors	USER MEMORY
FFEF:FFD0	JSRP Vectors	
FFCF:FFCE		
: :	On-Chip ROM*	
E001:C000		
BFFF:BFFE		
: :	External Expansion Memory	
0301:0300		
02FF:02FE	On-Chip RAM	USER RAM
: :		
01C1:01C0		
0195:0194	Watchdog Address	Watchdog Logic
0192	T0CON Register	Timer Block T0:T3
0191:0190	TMMODE Register	
018F:018E	DIVBY Register	
018D:018C	T3 Timer	
018B:018A	R3 Register	
0189:0188	T2 Timer	
0187:0186	R2 Register	
0185:0184	I2CR Register/ R1	
0183:0182	I3CR Register/ T1	
0181:0180	I4CR Register	
015E:015F	EICR	Timer Block T4:T7
015C	EICON	
0153:0152	Port P Register	
0151:0150	PWMODE Register	
014F:014E	R7 Register	
014D:014C	T7 Timer	
014B:014A	R6 Register	
0149:0148	T6 Timer	
0147:0146	R5 Register	
0145:0144	T5 Timer	
0143:0142	R4 Register	
0141:0140	T4 Timer	
0128	ENUR Register	UART
0126	TBUF Register	
0124	RBUF Register	
0122	ENUI Register	
0120	ENU Register	

011F:011E	A/D Result Register 7	A to D Registers†
011D:011C	A/D Result Register 6	
011B:011A	A/D Result Register 5	
0119:0118	A/D Result Register 4	
0117:0116	A/D Result Register 3	
0115:0114	A/D Result Register 2	
0113:0112	A/D Result Register 1	
0111:0110	A/D Result Register 0	
0106	A/D Control Register #3	
0104	Port D Input Register	
0102	A/D Control Register #2	A to D Registers†
0100	A/D Control Register #1	
00F5:00F4	BFUN Register	PORTS A & B CONTROL
00F3:00F2	DIR B Register	
00F1:00F0	DIR A Register / IBUF	
00E6	UPIC Register	UPI CONTROL
00E3:00E2	Port B	PORTS A & B
00E1:00E0	Port A / OBUF	
00DE	Microcode ROM Dump	PORT CONTROL & INTERRUPT CONTROL REGISTERS
00DD:00DC	HALT Enable Register	
00D8	Port I Input Register	
00D6	SIO Register	
00D4	IRCD Register	
00D2	IRPD Register	
00D0	ENIR Register	
00CF:00CE	X Register	HPC CORE REGISTERS
00CD:00CC	B Register	
00CB:00CA	K Register	
00C9:00C8	A Register	
00C7:00C6	PC Register	
00C5:00C4	SP Register	
00C3:00C2	(reserved)	
00C0	PSW Register	
00BF:00BE	On-Chip RAM	USER RAM
: :		
0001:0000		

*Note: The HPC16164 and HPC16064 On-Chip ROM is on addresses C000:FFFF and the External Expansion Memory is 0300:BFFF. The HPC16104 and HPC16004 have no On-Chip ROM, External Memory is 0300:FFFF.

†Note: Only one HPC16164 and HPC16104 have on-board A/D.

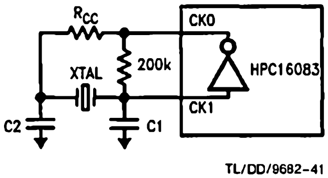
Design Considerations

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.



TL/DD/9682-41

A recommended crystal oscillator circuit to be used with the HPC is shown below. See table for recommended component values. The recommended values given in the table below 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.

HPC Oscillator Table

f_C (MHz)	R_{CC} (Ω)	C1 (pF)	C2 (pF)
2	50	82	100
4	50	62	75
6	50	50	56
8	50	47	50
10	50	39	50
12	0	39	39
14	0	33	39
16	0	33	39
18	0	33	33
20	0	33	33
22	0	27	39
24	0	27	39
26	0	27	33
28	0	27	33
30	0	27	27

Crystal Specifications:

"AT" cut, parallel resonant crystals tuned to the desired frequency with the following specifications are recommended:
Series resistance < 65Ω
Loading capacitance: C_L = 20 pF

HPC16164/26164/36164/46164/16104/26104/36104/46104/16064/26064/36064/46064/16004/26004/36004/46004

HPC16164 CPU

The HPC16164 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.

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 HPC16164 (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+MemI → MA carry → C
ADC	Add with carry	MA+MemI+C → MA carry → C
ADDS	Add short imm8	A+imm8 → A carry → C
DADC	Decimal add with carry	MA+MemI+C → MA (Decimal) carry → C
SUBC	Subtract with carry	MA−MemI+C → MA carry → C
DSUBC	Decimal subtract w/carry	MA−MemI+C → MA (Decimal) carry → C
MULT	Multiply (unsigned)	MA*MemI → MA & X, 0 → K, 0 → C
DIV	Divide (unsigned)	MA/MemI → MA, rem. → X, 0 → K, 0 → C
DIVD	Divide Double Word (unsigned)	X & MA/MemI → MA, rem → X, 0 → K, Carry → C
IFEQ	If equal	Compare MA & MemI, Do next if equal
IFGT	If greater than	Compare MA & MemI, Do next if MA > MemI
AND	Logical and	MA and MemI → MA
OR	Logical or	MA or MemI → MA
XOR	Logical exclusive-or	MA xor MemI → MA
MEMORY MODIFY INSTRUCTIONS		
INC	Increment	Mem + 1 → Mem
DECSZ	Decrement, skip if 0	Mem −1 → Mem, Skip next if Mem = 0

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	MemI → 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

MemI 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

Memory Usage

Number Of Bytes For Each Instruction (number in parenthesis is 16-Bit field)

	Using Accumulator A						To Direct Memory			
	Reg Indir.		Direct	Indir	Index	Immed.	Direct		Immed.	
	(B)	(X)					*	**	*	**
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

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 HPC16164 has been designed to be extremely code-efficient. The HPC16164 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 HPC16164, 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 HPC16164 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 his 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 his table; the assembler can give him 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 HPC16164 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 HPC16164 supplies 8-bit byte capability for 2-digit variables and literal variables.

MULTIPLY AND DIVIDE INSTRUCTIONS

The HPC16164 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

DEVELOPMENT SYSTEM

The Microcomputer On Line Emulator (MOLE) is a low cost development system and emulator for all microcontroller products. These include COPS™ microcontrollers and the HPC family of products. The development system consists of a BRAIN Board, Personality Board and optional host software.

The purpose of the development system is to provide the user with a tool to write and assemble code, emulate code for the target microcontroller and assist in both software and hardware debugging of the system.

It is a self contained computer with its own firmware which provides for all system operation, emulation control, communication, PROM programming and diagnostic operations.

It contains three serial ports to optionally connect to a terminal, a host system, a printer or a modem, or to connect to other development systems in a multi-development system environment.

The development system can be used in either a stand alone mode or in conjunction with a selected host system using PC-DOS communicating via a RS-232 port.

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 MOLE 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: MOLE-DIAL-A-HLP

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

Development Support (Continued)

Development Tools Selection Table

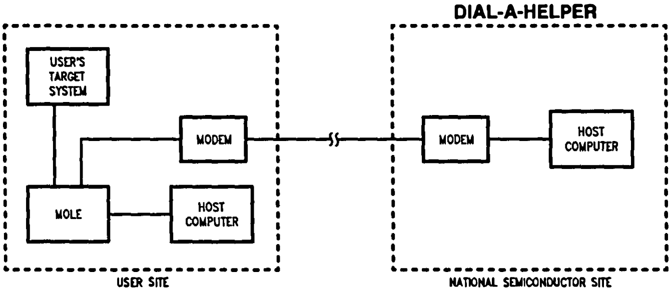
Microcontroller	Order Part Number	Description	Includes	Manual Number
HPC	MOLE-BRAIN	Brain Board	Brain Board Users Manual	420408188-001
	MOLE-HPC-PB1	Personality Board	HPC Personality Board Users Manual	420410477-001
	MOLE-HPC-IBM-R	Relocatable Assembler Software for IBM	HPC Software Users Manual and Software Disk PC-DOS Communications Software Users Manual	424410836-001 420040416-001
	MOLE-HPC-IBM-CR	C Compiler for IBM	HPC C Compiler Users Manual and Software Disk Assembler Software for IBM MOLE-HPC-IBM	424410883-001
	MOLE-HPC-VMS	Assembler, Loader, Librarian for VAX/VMS	HPC Software User's Manual and 9 Track Tape	424410836-001
	MOLE-HPC-VMS-C	C Compiler for VAX/VMS	HPC Software User's Manual and 9 Track Tape (Includes Assembler)	424410883-001
	424410897-001	Users Manual		424410897-001

Development Support (Continued)

FACTORY APPLICATIONS SUPPORT

Dial-A-Helper also provides immediate factory applications support. If a user is having difficulty in operating a MOLE, he can leave messages on our electronic bulletin board, which we will respond to, or under extraordinary circumstances, he can arrange for us to actually take control of his system via modem for debugging purposes.

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	

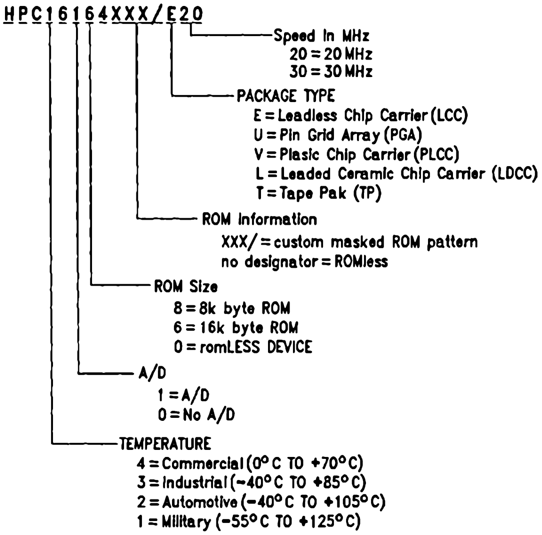


TL/DD/9682-37

Part Selection

The HPC family includes devices with many different options and configurations to meet various application needs. The number HPC16164 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.


HPC16164XXX/E20
Speed In MHz
20 = 20 MHz
30 = 30 MHz
PACKAGE TYPE
E = Leadless Chip Carrier (LCC)
U = Pin Grid Array (PGA)
V = Plasic Chip Carrier (PLCC)
L = Leaded Ceramic Chip Carrier (LDCC)
T = Tape Pak (TP)
ROM Information
XXX/ = custom masked ROM pattern
no designator = ROMless
ROM Size
8 = 8k byte ROM
6 = 16k byte ROM
0 = romLESS DEVICE
A/D
1 = A/D
0 = No A/D
TEMPERATURE
4 = Commercial (0°C TO +70°C)
3 = Industrial (-40°C TO +85°C)
2 = Automotive (-40°C TO +105°C)
1 = Military (-55°C TO +125°C)

TL/DD/9682-33

FIGURE 8. HPC Family Part Numbering Scheme

Examples

HPC46104E20 — ROMless, Commercial temp. (0°C to 70°C), LCC

HPC16164XXX/U20— 16k masked ROM, Military temp. (−55°C to +125°C), PGA

HPC26104XXX/V20— ROMless, Automotive temp. (−40°C to +105°C), PLCC