H8/300H SERIES Microcontrollers

HITACHI

Notice

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.
2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without Hitachi's permission.
3. Hitachi will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's unit according to this document.
4. Circuitry and other examples described herein are meant merely to indicate the characteristics and performance of Hitachi's semiconductor products. Hitachi assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein.
5. No license is granted by implication or otherwise under any patents or other rights of any third party or Hitachi, Ltd.
6. MEDICAL APPLICATIONS: Hitachi's products are not authorized for use in MEDICAL APPLICATIONS without the written consent of the appropriate officer of Hitachi's sales company. Such use includes, but is not limited to, use in life support systems. Buyers of Hitachi's products are requested to notify the relevant Hitachi sales offices when planning to use the products in MEDICAL APPLICATIONS.

Contents

Welcome 1

CPU 5

Addressing 7
Addressing Modes 7

Instruction Set 9
Arithmetic Instructions 10
Bit Processing 11
Block Move Instruction 12
Software Interrupt 12

CPU States / Low power modes 13
Low Power Modes 13
Sleep Mode 13
Software Standby Mode 13
Hardware Standby Mode 14
Extra Power Down Support - H8 / 3048 14
Clock Gearing 15

Exceptions and Interrupts 17
Trap Exceptions 18
Interrupt Controller 18
H8 / 300 Compatible Mode 18
H8 / 300H Advanced Mode 18
External Interrupts 18
Interrupt Vectors 18
Interrupt Response Time 19

On-chip Memory 21

Flash Memory (F-ZTAT) 23
Technology 24

Bus State Controller (BSC) 25
DRAM and PSRAM Interface 26
Chip Select Generation 27

Direct Memory Access Controller (DMAC) 29
Short Address Mode 30
Full Address Mode 30
DMAC Interrupts 30
DMAC Modes 30

Integrated Timer Unit (ITU) 33
Output Compare Functions 35
Input Capture Functions 35
Timer Synchronisation 35
PWM Operating Modes 36
Standard PWM Mode 36
AC Motor Control Outputs 37
Complementary 6-Phase PWM 37
Reset Synchronised PWM 38
Phase Counting Mode 38
ITU Interrupts 39

Timing Pattern Controller (TPC) 41
Stepper Motor Control with the TPC 41

Watchdog Timer (WDT) 45

Serial Communications Interface (SCI) 47

Analogue to Digital Converter (ADC) 49
Digital to Analogue Converter (DAC) 50

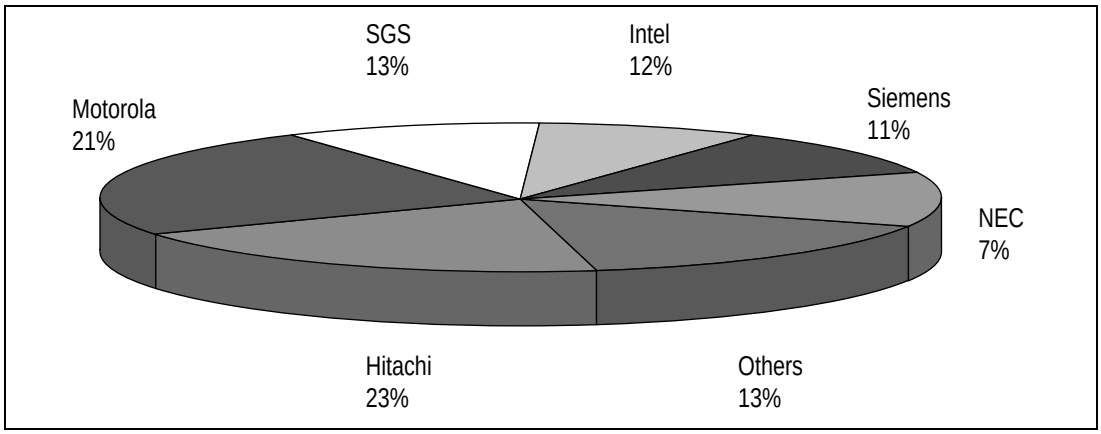
H8/300H Summary 51
H8/3001 series 55
H8/3002 series 56
H8/3003 series 57
H8/3004 and H8/3005 series 58
H8/3032 series 59
H8/3042 series 60
H8/3048 series 61
Packages 62

Ordering Information 63

Welcome

to Hitachi's 16-Bit microcontroller family H8/300H.

H8/300H has enjoyed tremendous success since its introduction in 1993 as a successor to Hitachi's equally successful H8/500 family. Dataquest has found Hitachi's 16-Bit microcontrollers to be the most successful world-wide as well as in Europe.



European 16-Bit µC market shares 1995 **Source:Dataquest**

At Hitachi Europe we think that this success was driven by our strong commitment to be a leading force in the microcontroller marketplace and our belief that we must listen to our customer's requirements and then meet these. As H8/300H is the result of combining many years of experience of Hitachi with the experience of our customers, H8/300H is an excellent example where this policy has worked for our customers and Hitachi.

The European Electronics Industry demands full service and support.

At Hitachi, we responded by setting up a European engineering and tool design subsidiary 12 years ago: Hitachi Microsystems Europe (HMSE) based in Maidenhead (UK).

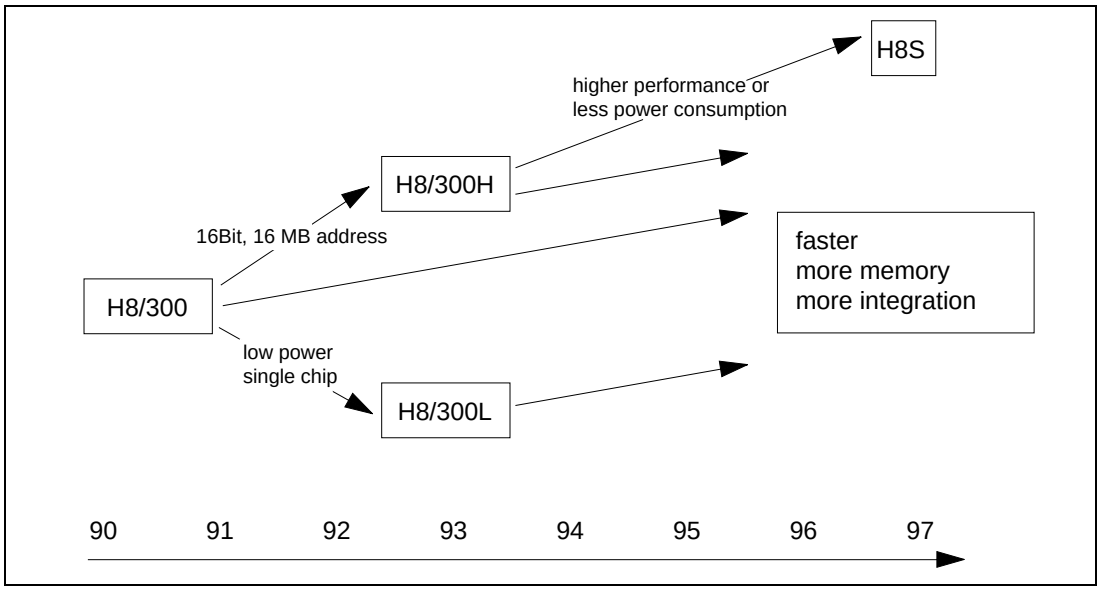
HMSE provides our customers with locally designed and supported tools ranging from **low cost evaluation boards** to fully featured **real time emulators** based on IBM-compatible PC's at a very competitive price. Software ranges from Assembler, an ANSI **C-Compiler** via a C-level debugger to HIOS, Hitachi's real time operating system. To speed development HMSE supplies MakeApp, a tool that sets up peripherals and creates driver routines on the click of a mouse.

HMSE also offers support and engineering resources for customers wishing to use Hitachi's ASIC capabilities. This also applies to our µCBIC program, enabling our customers to select one of Hitachi's CPU cores and combine it with peripherals from our library and adding customer specified logic via VHDL or Verilog.

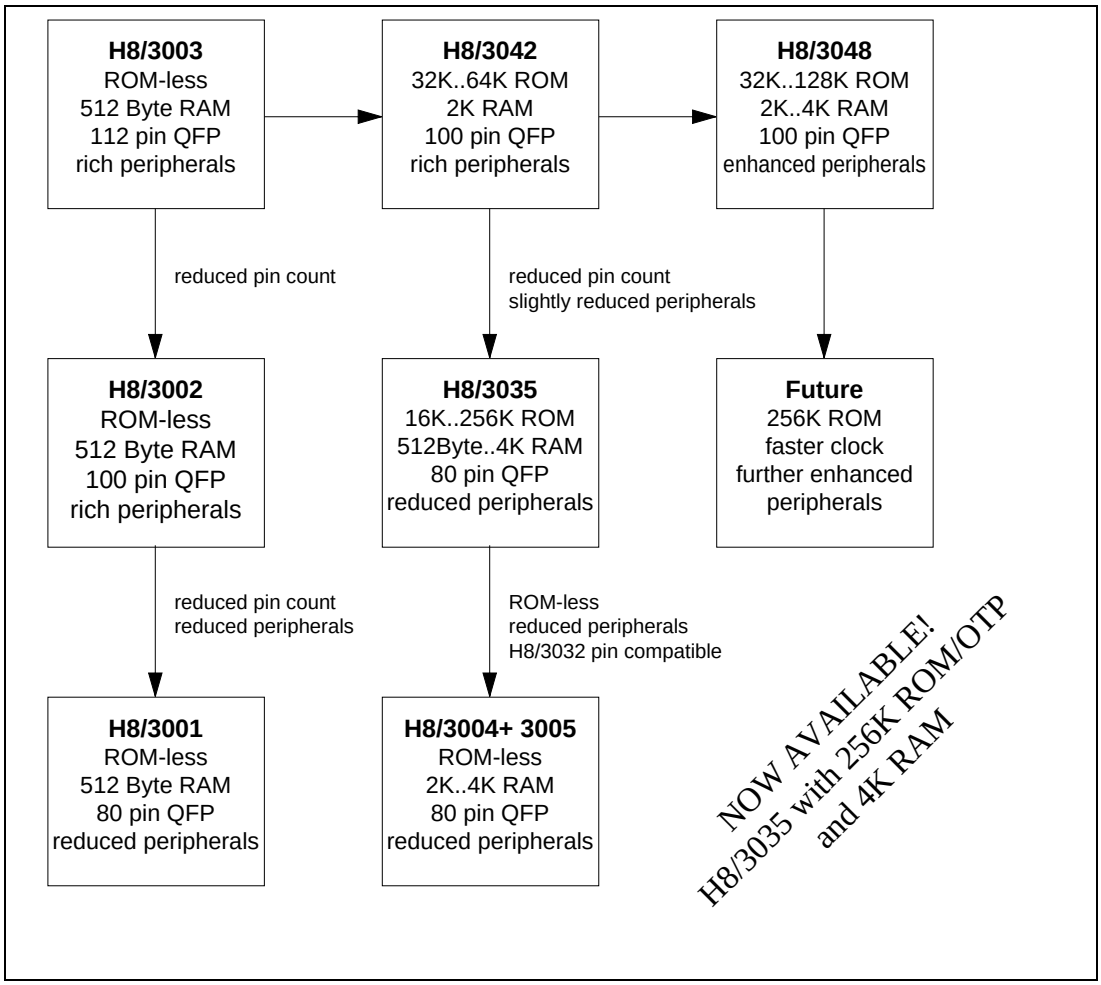
Hitachi also provides two **technical help lines with 24 hour response,** based in Maidenhead and Munich, as well as **local language application support** in Italy, France and Scandinavia.

All of this backed by strong support from third party tool manufacturers, like Hewlett Packard, Pentica and Lauterbach to name just a few.

H8/300H is part of Hitachi's **software compatible H8 product range,** which covers a performance range from 400ns to 50ns cycle time, whilst increasing word length from 8 to 16 bit and memory space from 64KB to 16MB. This product range offers an industry leading mix of memory options (including flash), peripherals and performance/power consumption, all of this being software compatible for protection of our customer's software investment. Hitachi produces and ships over 12 million H8 microcontrollers every month, in a vast range of advanced packaging and temperature options.



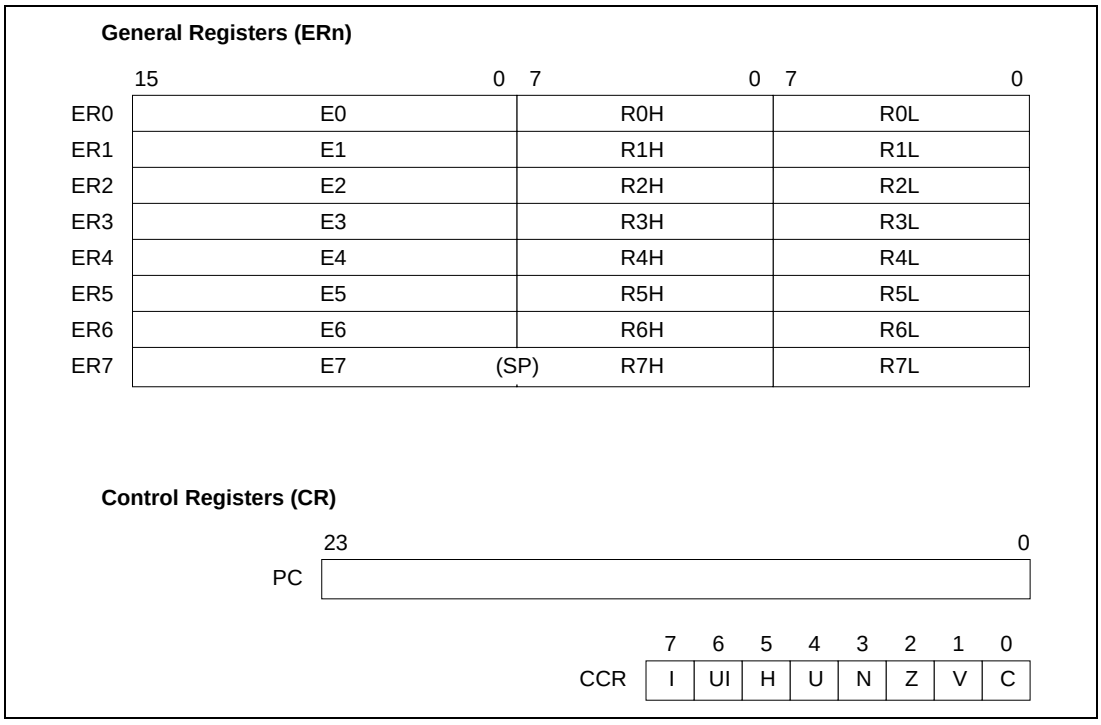
H8/300H has the performance (up to 10 native MIPS), the peripherals (very powerful timers, 10-Bit ADC, serial communications, etc.) and the memory options (including H8/3048F with 128KB flash) to make it an industry standard in **telecommunications** and very successful in **industrial** (e.g. motor control) and emerging **consumer applications** like new electronic video cameras, set top boxes and advanced car radios. Since its introduction H8/300H has been selected for hundreds of designs in Europe alone.


H8/3003
ROM-less
512 Byte RAM
112 pin QFP
rich peripherals
H8/3042
32K..64K ROM
2K RAM
100 pin QFP
rich peripherals
H8/3048
32K..128K ROM
2K..4K RAM
100 pin QFP
enhanced peripherals
reduced pin count
reduced pin count
slightly reduced peripherals
H8/3002
ROM-less
512 Byte RAM
100 pin QFP
rich peripherals
H8/3035
16K..256K ROM
512Byte..4K RAM
80 pin QFP
reduced peripherals
Future
256K ROM
faster clock
further enhanced
peripherals
reduced pin count
reduced peripherals
ROM-less
reduced peripherals
H8/3032 pin compatible
H8/3001
ROM-less
512 Byte RAM
80 pin QFP
reduced peripherals
H8/3004+ 3005
ROM-less
2K..4K RAM
80 pin QFP
reduced peripherals
NOW AVAILABLE!
H8/3035 with 256K ROM/OTP
and 4K RAM

CPU

The H8/300H CPU core has been designed as a 16/32 Bit general purpose register machine. This architecture makes execution of software written in C very efficient and results in dense code in order to reduce the amount of memory needed. Hence, H8/300H lowers total system cost.

The H8/300H CPU is a general purpose register machine as shown below. The CPU comprises eight 32-bit registers, each being further dividable into 16 and 8-bit registers. Being general purpose there is no restriction placed on how each register is used, thus they can be used for pointer or data operations. The architecture also allows any of the data addressing modes to be used in conjunction with any register.



This rich set of general purpose registers provides the compiler writer with ample opportunities to optimise the code generated by the compiler. Local variables can be optimised into registers wherever possible, thus reducing the number of bytes of code needed to manipulate them. Also, because each register can be used as an accumulator, index register or address pointer, the address arithmetic which must be performed by the compiler can be done very effectively. Register ER7 is used as a stack pointer, so all accesses to stack based data can be performed very fast.

These features significantly reduce the amount of code and time that is required to execute lines of C source code when compared to traditional architectures, which have limited numbers of fixed function accumulators and index registers.

The 32-bit register set also proves to be very useful in addressing large areas of memory, as a 24-bit pointer (this size pointer allows access to anywhere in the 16MBytes address space) can be stored and manipulated in one register.

In addition to the general purpose registers there are two control registers, a Condition Code Register (CCR) and a Program Counter (PC). The CCR is an 8-bit wide register which contains all the CPU flags such as overflow, zero and carry as well as the interrupt flags. The carry flag also doubles as a bit accumulator when the bit manipulation operations are used.

The H8/300H CPU offers a H8/300 compatible mode which allows straightforward reuse of existing H8/300 software. This mode is available in H8/3032, H8/3042 and μCBIC products.

Addressing

To support large memory systems the linear address space of the H8/300H CPU core allows direct access to every address in the whole 16MByte address space via 24-bit address pointers. The linear address space means there is no need to set up page registers and there are also no limitations on the size of code modules or data arrays and structures.

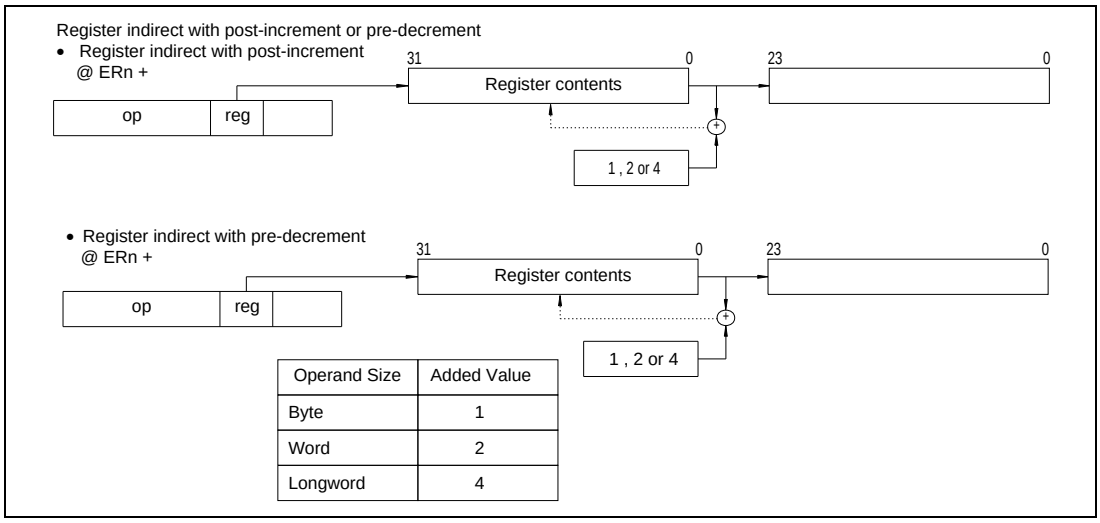
Addressing Modes

Another way a CPU architecture can support the efficiency of the compiler is by providing a full set of powerful and flexible addressing modes. A CPU which only provides rudimentary addressing modes makes a compiler inefficient in the access of variables, thereby increasing both code size and execution time. To ensure that the compiler is as efficient as possible the H8/300H CPU provides eight addressing modes as shown in the figure below.

Register direct	Rn
Register indirect	@ E Rn
Register indirect with displacement	@ (d: 16, E Rn) @ (d: 24, E Rn)
Register indirect with post-increment/ Register indirect with pre-decrement	@ E Rn + @ - E Rn
Absolute address	@ aa : 8 @ aa : 16 @ aa : 24
Immediate	# xx : 8 # xx : 16 # xx : 32
PC-relative	@ (d: 8, PC) @ (d: 16, PC)
Memory indirect	@@ a a : 8

Each instruction can use a subset of the available addressing modes. The data transfer instructions can make use of all addressing modes except PC relative and memory indirect. All arithmetic and logical operations can use the register direct and immediate modes and the bit manipulation instructions use the register direct, register indirect and absolute addressing modes.

Supporting both array and stack data types, the H8/300H has indirect addressing with either postincrement or predecrement. These modes support byte, word and long word data ($\pm$1,2 and 4) as shown.



Operand Size	Added Value
Byte	1
Word	2
Longword	4

Three absolute addressing modes are provided using 8, 16 or 24-bit absolute addresses. Using the 24-bit address the entire 16MBytes address space is accessible. The 8 and 16-bit absolute address modes assume that the upper byte or word of the address is H'FFFF or H'FF respectively. This allows for the efficient address specification for the on-chip I/O area and RAM areas which are both placed at the top of the address map. These shortened addresses save significant amounts of code when these areas are accessed.

Instruction Set

The H8/300H has an instruction set which suits the combined needs of HLL programming and embedded applications. It comprises of 62 instructions, with an emphasis on arithmetic instructions, address manipulation and bit processing. More than half of all instructions have an instruction length of only 2Bytes making very compact code.

Function	Instruction
Data transfer	MOV, PUSH, POP, MOVTPE, MOVFPE
Arithmetic operations	ADD, SUB, ADDX, SUBX, INC, DEC, ADDS, SUBS, DAA, DAS, MULXU, DIVXU, MULXS, DIVXS, CMP, NEG, EXTS, EXTU
Loqic operations	AND,OR,XOR,NOT
Shift operations	SHAL, SHAR, SHLL, SHLR, ROTL, ROTR, ROTXL, ROTXR
Bit manipulation	BSET, BCLR, BNOT, BTST, BAND, BIAND, BOR, BIOR, BXOR, BIXOR, BLD, BILD, BST, BIST
Branch	Bcc, JMP, BSR, JSR, RTS
System control	TRAPA, RTE, SLEEP, LDC, STC, ANDC, ORC, XORC, NOP
Block data transfer	EEPMOV

In comparison with the H8/300 CPU most of the data transfer, logical, shift and arithmetic instructions are improved to handle 16 and 32-bit data. New instructions added to the H8/300H include signed multiplication, sign extension, 16-bit branch instructions and a software trap instruction. The following table illustrates the new and improved instructions provided by the H8/300H.

Instruction \ DataSize	Byte	Word	Long Word
Data Transfer			
MOV	❍	❍	●
EEPMOV (Block Transfer)	❍	●	
Arithmetic Operations			
ADD, SUB, CMP,	❍	❍ *	●
MUL U/S	❍	●	
DIV U/S	❍	●	
EXT. S/U		●	●
Logical Operations			
AND, OR, XOR, etc.	❍	●	●
Shift and Rotate			
SHAL, SHAR, ROT, etc.	❍	●	●
System Control			
PUSH, POP TRAPA	❍	●	●
Branch			
Bcc d : 16		●	

❍ H8/300 & H8/300H

● H8/300H

* improved addressing mode on H8/300H

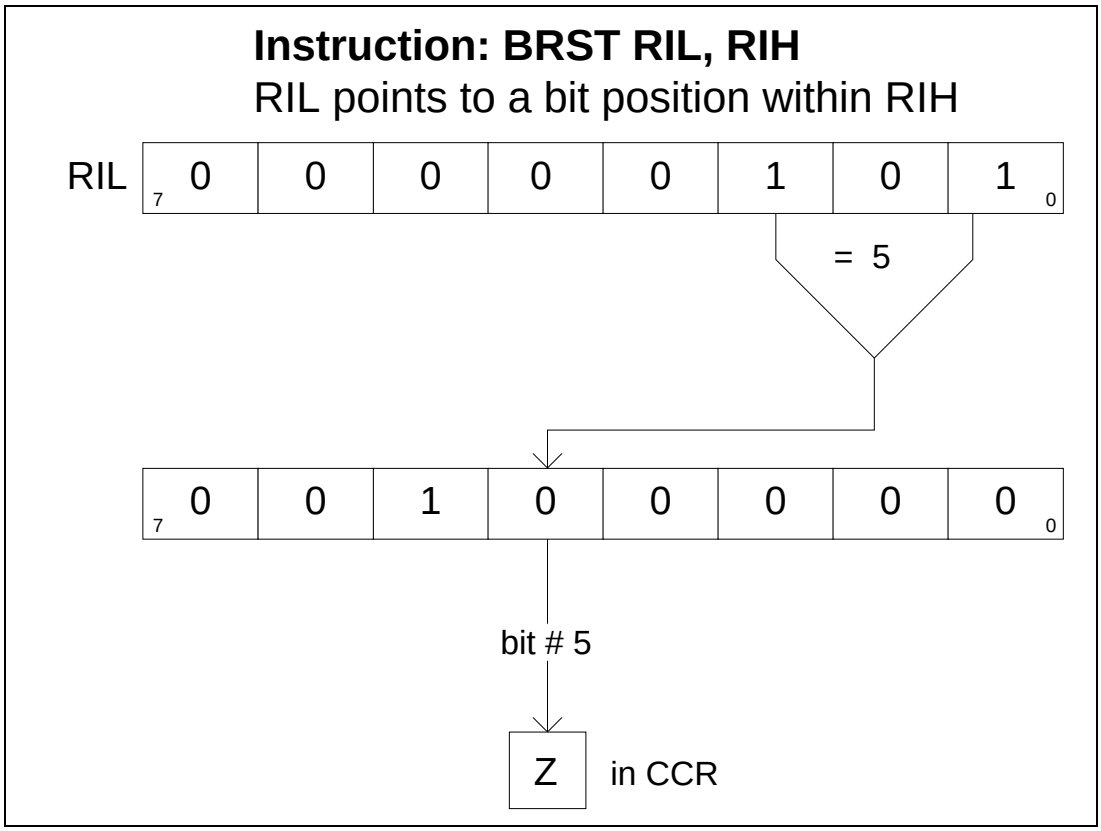
Arithmetic Instructions

In order to perform complex algorithms such as digital filtering the H8/300H is equipped with powerful arithmetic instructions, including addition and subtraction on 32-bit data and multiplication of 16 x 16-bit data and division of 32 / 16-bit data. Multiplication and division are both available as signed and unsigned operations, which eliminate the need for time consuming library calls. The table below gives a guide to the execution speed of various arithmetic instructions with a clock of 16MHz.

Add 32bit operands	125ns
AND 32 bit operands	125ns
multiply/divide 16bit operands (32bit result, signed)	1.5 µs
multiply/divide 16bit operands (32bit result, unsigned)	1.375 µs

Bit Processing

In microcontroller applications it is often necessary to manipulate data on a bit by bit basis. A good example would be where an I/O pin needs to be set to switch on a lamp or a solenoid. To meet this demand the H8/300H has 14 separate bit processing instructions which allow the programmer to manipulate bit data very easily.



It is also possible to perform boolean algebra on bit data using the carry flag of the CCR register as a bit accumulator. In a microcontroller application it is often necessary to perform a branch depending on the values of two bit flags located in RAM or I/O ports. Using the boolean operations provided by the H8/300H the first bit can be loaded into the carry flag. Then a bitwise logical operation can be executed using the second bit. This sequence would then be followed by a branch depending on the value of the carry flag.

Another feature of the H8/300H bit processing capability is its ability to access bits indirectly, using the value from a general purpose register as a bit pointer. This mechanism is shown below and is useful for scanning a byte for set or cleared bits.

Block Move Instruction

Another efficient instruction provided by the H8/300H CPU is the EEPMOV or block data transfer. This is useful when a table stored in ROM has to be transferred to RAM for manipulation. Block sizes up to 64KBytes can be transferred with a single instruction.

Software Interrupt

The TRAPA instruction has been added to the H8/300H CPU. This instructions implements a software interrupt, jumping to a service routine via one of four exception vectors (TRAPA 0 - 3). This operation can be used to implement fast, space efficient calls to often used sub-routines such as schedulers and other O/S routines. The TRAPA instruction can also be used as a call to an error handling routine.

CPU States / Low power modes

The H8/300H CPU has four different processing states: program execution, exception handling, bus-released and power-down.

In the program execution state the CPU executes normal program instructions in sequence, while the exception handling state is a transient state in which the CPU executes an exception handling sequence in response to a reset, interrupt or other exception. In the bus-released state the external bus has been released to an external bus-master other than the CPU. In the power down mode the CPU is halted to conserve power. The power down modes include three modes: sleep, software standby and hardware standby. These modes are enhanced in the H8/3048 series by adding module standby and clock gearing.

Low Power Modes

The H8/300H Series has been designed to be a microcontroller with high performance and low power dissipation. It therefore can be used in 3V or 5V systems and only consumes 20mA (max.) when operating at 3V and 8MHz.

To widen its use in battery operated equipment such as cellular telephones, an impressive set of low power modes are also provided on all devices.

Sleep Mode

In this mode the device switches off the clock to the CPU, but all of the on-board peripherals remain active, and register and memory contents are retained. Sleep mode is entered via the “SLEEP” instruction; the CPU exits this mode whenever an enabled interrupt occurs.

A useful application for this mode is to reduce the average power consumed by a system, using a timer to “wake” the CPU after a period of sleep.

Once woken, the CPU can process for a period of time and then after loading the timer again the SLEEP instruction can be executed, again lowering the power consumption.

When sleep mode is entered the device's current consumption is reduced by approximately one third over its operating value.

Software Standby Mode

Again, this mode can be entered using the SLEEP instruction, but in this case the on-chip oscillator is stopped completely, putting the device into software standby. Standby current is very low with a maximum value of 5µA. This is coupled with a data retention voltage of 2V, allowing the microcontrollers internal RAM contents to be maintained using just two 1.5V

battery cells or possibly a large “reservoir” capacitor. During software standby mode, the microcontroller maintains the value of the I/O ports, so output ports can be set to the values the system requires during power down, with the knowledge that they will remain stable during software standby mode.

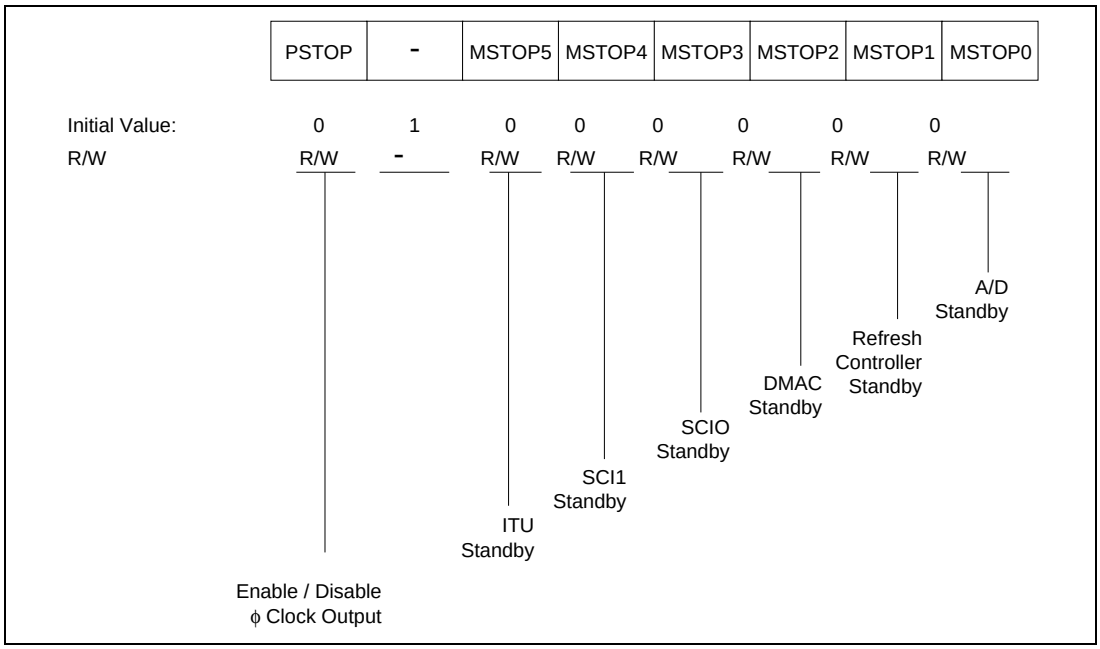
To exit from this mode, an external interrupt can be used, and a specialised timer circuit is provided to ensure that the on-chip oscillator has started and is stable before execution of the interrupt service routine begins.

Hardware Standby Mode

This mode allows the device to be put into the low power mode via an external pin. While in this mode the maximum current consumption is 5μA and to exit hardware standby the chip must be reset.

Extra Power Down Support - H8 / 3048

The H8/3048 range incorporates some extra power down options making it an ideal device in many high performance battery driven systems. The first feature is the ability to put individual peripherals into standby mode via software. The control register used for this operation is shown below.



This feature makes the H8/3048 an ideal fit into cellular handsets as it mirrors the hardware designer’s efforts to enable parts of the circuit to be powered down when they are not required.

Clock Gearing

Another power down feature provided by the H8/3048 is its ability to change the on-chip operating frequency via a software command. This is achieved using a programmable clock divider which allows the clock to be divided by 1, 2, 4 or 8.

Bit 1	Bit 0	Divide Ratio	ϕ = 16 MHz
DIV 1	DIV 0		
0	0	1/1	ϕ = 16MHz
0	1	1/2	ϕ = 8 MHz
1	0	1/4	ϕ = 4 MHz
1	1	1/8	ϕ = 1 MHz

Clock gearing allows the performance and power dissipation to be changed to suit the system's current mode. For example, when scanning a keyboard or other input device the divide by 8 option could be selected, but once a pressed key is detected the divide by 1 option can be selected to instantly give the device full speed operation.

Exceptions and Interrupts

Exceptions on the H8/300H CPU fall into four categories, reset (highest priority), external interrupts, internal interrupts and trap exceptions. The following table shows the exception vector table for the H8/300H CPU core.

Exception Source	Vector	Vector Address
Reset	0	H'0000 to H'0003
Reserved for system use	1	H'0000 to H'0007
	2	H'0008 to H'000B
	3	H'000C to H'000F
	4	H'0010 to H'0013
	5	H'0014 to H'0017
	6	H'0018 to H'001B
External interrupt (NMI)	7	H'001C to H'001F
Trap instruction (4 sources)	8	H'0020 to H'0023
	9	H'0024 to H'0027
	10	H'0028 to H'002B
	11	H'002C to H'002F
External interrupt IRQO	12	H'0030 toH'0033
External interrupt IRQ1	13	H'0034 to H'0037
External intemipt IRQ2	14	H'0038 to H'003B
External interrupt IRQ3	15	H'003C to H'003F
External interrupt IRQ4	16	H'0040 to H'0043
External interrupt IRQ5	17	H'0044 to H'0047
External interrupt IRQ6	18	H'0048 to H'004B
External interrupt IRQ7	19	H'004C to H'004F
Internal interrupts	20 to 60	H'0050 to H'0053 to H'00F0 to H'00F3

Trap Exceptions

When the TRAP instruction is executed, the program will start executing from the location specified in the relevant vector. The TRAP instruction has four vectors as specified by its argument. This exception can be used as an efficient mechanism for calling operating system functions, as it takes only 2 Bytes to execute a TRAP operation, compared to 4 Bytes for a BSR and 8 Bytes for a JSR.

Interrupt Controller

The H8/300H interrupt controller (INTC) can be operated in two modes, either maintaining compatibility with the standard H8/300 INTC, or in a more advanced mode.

H8 / 300 Compatible Mode

In this mode, the acceptance of maskable interrupts is controlled by the I bit in the CCR. If I is set then all interrupts are disabled and if I is cleared then all interrupts are enabled. When an interrupt is accepted the INTC automatically sets the I bit, thus disabling any other maskable interrupt for the duration of the interrupt service routine (ISR), unless it is cleared by the user's code.

H8 / 300H Advanced Mode

To increase the power of the interrupt controller, an extra interrupt status bit is included in the condition code register, known as the Ul or User Interrupt bit. This extended operation allows the user to specify raised priority interrupt sources, which are capable of interrupting a low priority ISR which is already running. The priority of individual interrupt sources is programmed in a number of interrupt priority registers (IPR).

External Interrupts

All the H8/300H devices include several external interrupts, including one non maskable interrupt (NMI). NMI can be programmed to be activated on either the rising or falling edge and the standard external interrupts can be programmed to recognise either a low level or a low going edge.

Interrupt Vectors

To speed up the processing of interrupts, every interrupt source has its own vector. For example, for each serial port there are separate vectors for transmit, receive and error interrupts. Therefore, the ISR does not need to poll a peripheral block to find the source of any interrupt.

Interrupt Response Time

Interrupts are responded to rapidly on the H8/300H. When code and data are both located in the on-chip memory, then the ISR will be reached within 2.6 microseconds (worst case) at 16MHz. If this is coupled with the individual vector structure provided by the H8/300H, the ISR can be performing useful work very quickly indeed. The table below shows how fast H8/300H responds to interrupts in several configurations.

			External Memory			
			8-Bit Bus		16-Bit Bus	
No	**Item**	**On-Chip Memory**	**2States**	**3 States**	**2 States**	**3 States**
1	Interrupt priority decision	2	2 *1	2 *1	2 *1	2 *1
2	Maximum number of states until end of current instruction	1 to 23	1 to 27	1 to 31 *4	1 to 23	1 to 25*4
3	Saving PC and CCR to stack	4	4	12 *4	4	6 *4
4	Vector fetch	4	8	12 *4	4	6 *4
5	Instruction prefetch *2	4	8	12 *4	4	6 *4
6	Internal processing *3	4	4	4	4	4
Total	(Clocks)	19 to 41	27 to 53	43 to 73	19 to 41	25 to 49

Notes:
1. One state for internal interrupts
2. Prefetch after the interrupt is accepted and prefetch of the first instruction in the interrupt service routine
3. Internal processing after the interrupt is accepted and internal processing after prefetch
4. The number of states increases if wait states are inserted in external memory access

On-chip Memory

Often, due to its power and ability, a H8/300H device will be used in applications where the program size is very large. At first glance, these large programs appear to preclude using any microcontroller in single chip mode.

But large programs place no restriction on the H8 / 300H family, as unparalleled sizes of on-chip program and data memory are available, even in its' smallest package.

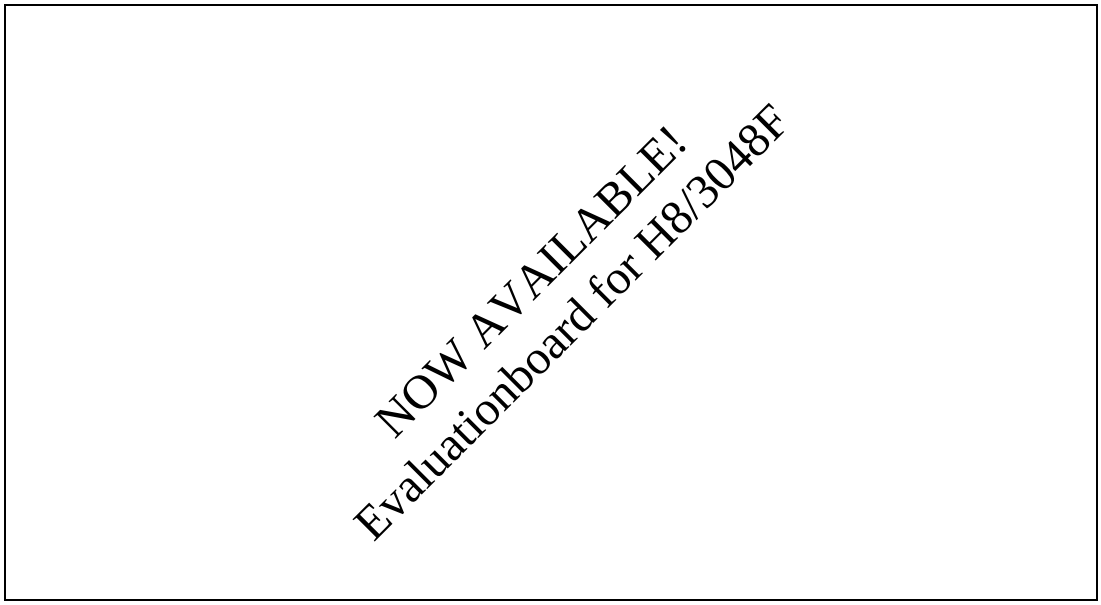
For example the H8 / 3048 with 128K of PROM / ROM / Flash and 4K of RAM in a single chip, 100-pin device measuring just 17.2 mm across its pins. New variants of the H8 / 300H family are also under development.

ROM RAM	**0**	**16k**	**32k**	**48k**	**64k**	**96k**	**128K**	**192K**	**256k**
512 Byte	3001 3002 3003	3030							
1k		3031							
2k	3004		3040 3044	3041	**3032** **3042** 3045				
4k	3005					3047	3033 **3048** **3048 Flash**	3034	**3035**

BOLD products are also available as ZTAT (OTP).

Flash Memory (F-ZTAT)

Together with the Flash-derivatives in other microcontroller families, Hitachi offers the **worlds best line up of microcontrollers with on-chip flash** currently in full production. Hundreds of customers world wide have taken advantage of our flash based microcontrollers in their designs, about one third in industrial and automotive applications, approximately one fourth each in consumer and office automation and the remainder in telecommunications.



The **Advantages** of flash based microcontrollers include:

- end of line programming allows flexibility in the software until shipment
- software flexibility during production ramp up and development
- allows easy and fast update in the field without the need to open the equipment
- allows software updates even remotely, e.g. via modem/phone line
- fast response to changing customer requirements
- non volatile storage of data and parameters

Technology

Flash memory is programmed by applying a high gate-drain voltage that draws hot electrons generated in the vicinity of the drain into a floating gate. The threshold voltage of a programmed cell is therefore higher than that of an erased cell. Cells are erased by grounding the gate and applying a high voltage to the source, causing the electrons stored in the floating gate to tunnel out. A flash memory cell is read like an EPROM cell, by driving the gate to the high level and detecting the drain current, which depends on the threshold voltage. Programming takes 50µs/Byte and erasing 1s per block or for the full flash memory.

Hitachi's **H8/3048F** offers **full speed operation** (16MHz/5V and 8MHz/3V), block division into 8 small (512Byte) and 8 large blocks (12K and 16K) and various ways to program the H8/3048F's flash memory. These modes are selected by pins on reset and allow on-board programming:

- User program mode: In user program mode the flash memory can be erased and programmed under control of an user program. In this mode all the resources of the H8/3048F can be used except the flash memory, as reading from the flash memory is not possible during erasing or programming.
- Boot mode: Boot mode allows the user to erase and program the flash memory via the serial interface SCI1. This is possible even if the microcontroller contains no user software, i.e. it allows in-circuit-programming of new (empty) devices.
- PROM mode: The H8/3048F can also be programmed using general purpose PROM programming equipment. The H8/3048F is inserted in the programmer using a socket adapter available from Hitachi.

Other technical characteristics of Hitachi's on-chip Flash memory include:

- Security mechanism against malicious access makes theft of code virtually impossible.
- Accidental erase or write impossible without Vpp being present.
- 100 erase/write cycles guaranteed with higher specification under preparation.

Bus State Controller (BSC)

As one of the key reasons for using the H8/300H is the 16MBytes linear address space, it is likely that it will be used in a system with a large quantity of memory. Consequently, the H8/300H CPU core is supported by a powerful bus state controller (BSC) that allows the memory to be configured in the system in the most appropriate way.

The H8/300H BSC is able to configure eight memory areas with their own independent attributes. In the advanced modes, these areas are either 256KBytes (1 MBytes mode) or 2MBytes (16MBytes mode) in size. The attributes which can be set for each area are the bus width, the number of cycles for each external access and the wait state mode used.

	Bus width (bits)	access cycles	Size (max.)
SRAM	8/16	2..6	16MB
EPROM	8/16	2..6	16MB
PSRAM	8/16	2..6	16MB
DRAM	16	3..6	2MB

When a memory area is initialised into the 3-state access cycle mode, the wait state controller can be activated for this area to allow slow devices to be connected to the bus of the H8/300H.

There are four wait modes available: the programmable wait mode, pin auto wait mode and the pin wait modes 0 and 1.

Wait state controller mode	**Operation**
Pin wait mode 0	Two wait states are inserted whenever the wait pin is sampled low. The wait state controller is otherwise disabled for memory areas so specified.
Pin wait mode 1	The number of wait states programmed in the WSC are inserted and then the wait pin is sampled. If it is low further wait states are inserted.
Pin auto-wait mode	If the wait pin is low when sampled the number of wait states programmed in the WSC are inserted and then the bus cycle is terminated.
Programmable wait mode	For every access to the specified 3-state access area, the number of wait states programmed in the WSC are inserted automatically.

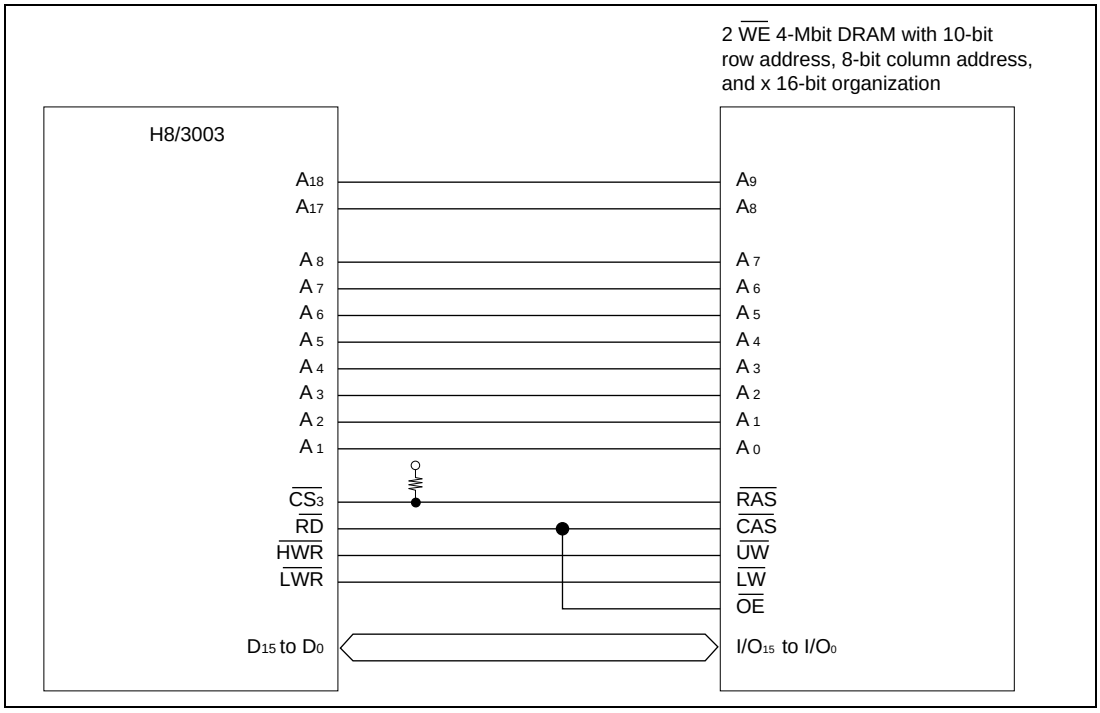
Combining the BSC with the on-chip refresh controller, allows the H8 / 300H to be interfaced to a wide range of memory types, including DRAM, SRAM, PSRAM and EPROM with the minimum of external logic. This function extends to the generation of chip selects corresponding to the memory area being accessed.

DRAM and PSRAM Interface

To provide a large area of RAM, the most cost effective memory type to use is DRAM. However, normally in an embedded system the external devices required to interface with DRAM often causes designers to use SRAM as the lowest cost system option.

To allow designers to utilise the full benefit of using DRAM in their system, the H8/300H has been equipped with a bus interface which can couple directly to DRAM with the minimum of external components.

This has been achieved by incorporating the logic required to produce the DRAM interface signals and a refresh controller into the BSC. For example, the figure below shows the connection of a H8/3003 to a 4-Mbit DRAM.



The H8/300H supports 1Mbit and 4Mbit DRAM (2WE or 2CAS) in 16-bit wide configurations. The refresh controller can be programmed to produce a wide variety of refresh intervals, and the DRAM controller can put the DRAM into self refresh mode whenever the software standby mode is selected.

Chip Select Generation

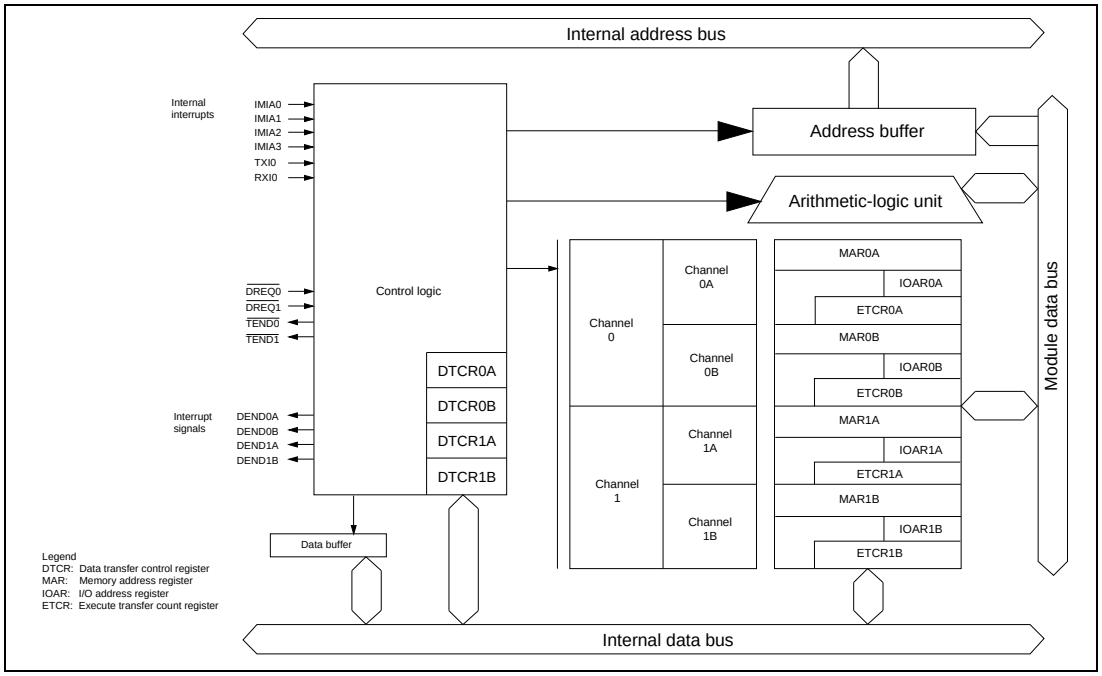
The BSC on the H8/3003, H8/3002 and H8/304X devices can be used to generate chip select signals for different memory areas. These chip selects have the added benefit of becoming active at the same time as the address becomes valid, thus removing any decode delay.

Direct Memory Access Controller (DMAC)

To complement a CPU which provides high performance operation for complex algorithms and an address space which can handle large data structures and program modules, the addition of a direct memory access controller (DMAC) on-chip will significantly increase the performance of the system. The DMAC will allow the utilisation of the whole CPU performance for the system's algorithms, while repetitive but important data transfer can be done via DMA.

Any external or internal interrupt that initiates a data transfer can be completely serviced by the DMA without any interrupts being handled by the CPU. This drastically reduces the CPU overhead needed for interrupt handling. When used with other on-chip peripherals such as the ITU, the DMAC allows the control of real time inputs and outputs, or it can be used to service the serial interfaces.

The basic H8/300H DMAC module incorporates four channels. However, the H8/3003 actually provides eight channels. Each channel in the DMAC module can be utilised to perform transfers between memory and I/O, while 2 channels need to be combined for memory to memory transfers. Byte and word transfer are possible in all available operating modes.



Short Address Mode

In this mode an 8-bit source address and 24-bit destination address (or vice versa) are used. During the transfer the 8-bit address (which points to an I/O register) is fixed while the 24-bit address may change according to the way the channel has been initialised. In this mode DMA transfers are initiated by interrupts from the timer block, the serial port or via an external signal.

One byte or word of data is transferred per request and the 24-bit address incremented by one or two after each transfer. If a fixed memory address is required, then the channel can be put into an idle mode, where the 24-bit address will not increment. Up to 64K transfers can be performed before an interrupt is signaled to the CPU.

The DMA channel can also be set to automatically repeat up to 256 transfers continuously, with the DMA channel being reinitialised and restarted after the specified number of transfers has been performed. This is useful when cyclic data such as a control pattern for a stepper motor has to be transferred.

Full Address Mode

To perform memory to memory transfers the full address mode can be used. Here, the source and destination addresses are 24-bits wide each, and therefore memory to memory transfer can be performed between any areas of the full 16MBytes address space.

DMA transfers can be initiated by software command, external signals or interrupts from the timer block. This mode allows either a single transfer to occur per request, or for a block of data to be moved. In the block mode, the DMAC can be set up in a burst mode, taking over the bus from the processor until all the transfers are complete, or in a cycle steal mode where the processor and the DMAC share the bus.

DMAC Interrupts

The DMAC can be set up to provide an interrupt per request, or to only interrupt the processor when it has performed a programmed number of transfers.

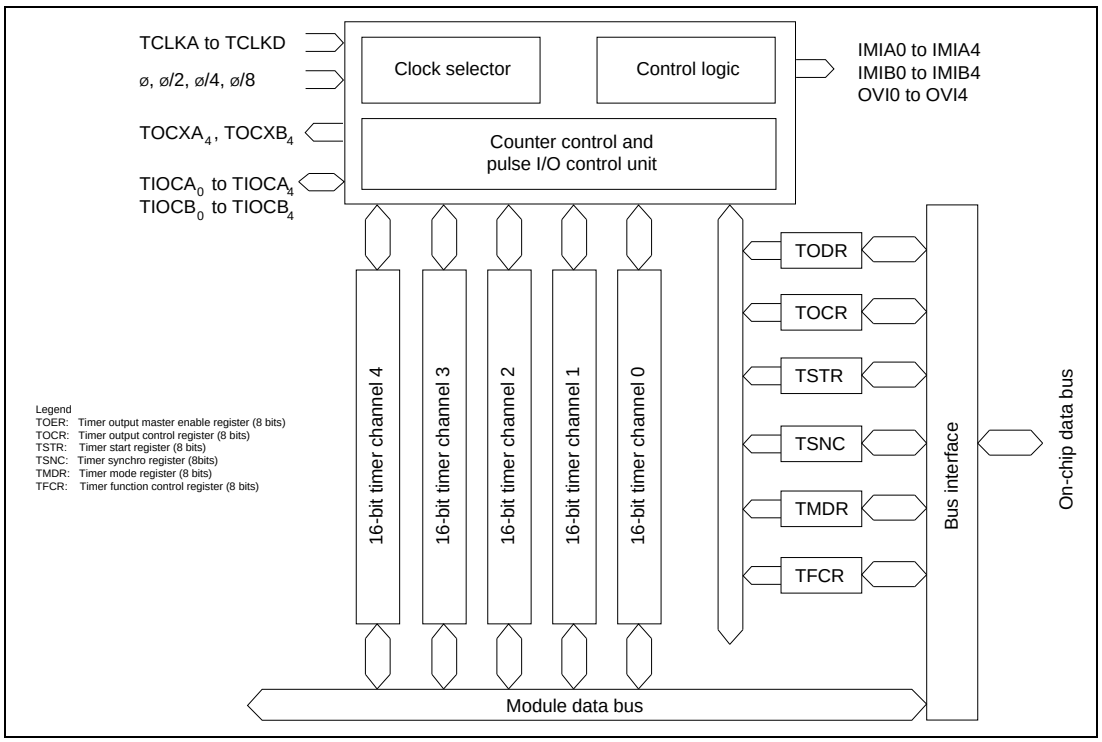
DMAC Modes

The DMAC provides a choice of short and full addressing modes, which allow the DMA to be adapted according to system requirements.

	Transfer mode	Activation	Address Register Length	
			Source	Destination
Short address mode	**IO mode** • transfers 1 byte or 1 word per request • increments or decrements the memory address by 1 or 2 • executes 1..65536 transfers	• Compare match/input capture A interrupts from ITU channels 0..3 • transmit data empty interrupt from SCI	24	8
	Idle mode • transfers 1 byte or 1 word per request • holds the memory address fixed • executes 1..65536 transfers	• Receive data full interrupt from SCI • External request	8	24
	Repeat mode • transfers 1 byte or 1 word per request • increments or decrements the memory address by 1 or 2 • executes a specified number (1..256) of transfers then returns to the initial state and continues		24	8
Full address	**Normal mode** • auto request — retains the transfer request externally — executes a specified number (1..65536) — of transfers continuously — selection of burst mode or cycle steal mode • external request — transfers 1 byte or 1 word per request — executes 1..65536 transfers	• auto request • external request	24	24
	Block transfer • transfers 1 block of a specified size per request • executes 1..65536 transfers • allows either the source or the destination to be a fixed block area • block size can be 1..256 byte or word	• compare match/input capture A interrupt from • ITU channels 0..3 • external request	24	24

Integrated Timer Unit (ITU)

In many microcontroller based systems, very specialised timer functions are required. Often these timer functions are produced in a user developed ASIC device, because the timers provided by the microcontroller do not meet the performance required by the designer. The timer unit on the H8 / 300H has been designed to allow maximum flexibility in its use, therefore allowing the designer to get the timer configuration required without resorting to an external timer device.



The ITU consists of five separate 16-bit timer channels, each of which can be clocked from an internal derivative of the system clock (ϕ, $\phi/2$, $\phi/4$ and $\phi/8$) or from an external pin. If the ϕ clock option is selected, then the minimum resolution of the timer is 62.5ns (ϕ = 16MHz). The standard timer functions provided by the ITU include ten general registers (GR), which can be used as output compares or input captures. Thus the complete timer block provides up to ten pulse inputs or outputs. A further four 16-bit buffer registers (BR) reduce the overhead placed on the CPU when servicing the timer block.

Item		Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Clock Sources		Internal clocks: φ, φ/2, φ/4, φ/8 External clocks: TCLKA, TCLKB, TCLKC, TCLKD, selectable independently				
General registers (output compare/ input capture registers)		GRA0, GRB0	GRA1, GRB1	GRA2, GRB2	GRA3, GRB3	GRA4, GRB4
Buffer registers		-	-	-	BRA3, BRB3	BRA4, BRB4
Input/output pins		TIOCA0 TIOCB0	TIOCA1 TIOCB1	TIOCA2 TIOCB2	TIOCA3 TIOCB3	TIOCA4, TIOCB4
Output pins		-	-	-	-	TOCXA4, TOCXB4
Counter clearing function		GRA0/GRB0 compare match or input capture	GRA1/GRB1 compare match or input capture	GRA2/GRB2 compare match or input capture	GRA3/GRB3 compare match or input capture	GRA4/GRB4 compare match or input capture
Compare match output	0	√	√	√	√	√
	1	√	√	√	√	√
	Toggle	√	√	-	√	√
Input capture		√	√	√	√	√
Synchronization		√	√	√	√	√
PWM mode		√	√	√	√	√
Reset-synchronized PWM mode		-	-	-	-	-
Complementary PWM mode		-	-	-	√	√
Phase counting		-	-	√	-	-
Buffering		-	-	-	√	√

Legend

√:Available

-: Not available

Output Compare Functions

To create output waveforms or timed interrupts, the ITU provides up to 10 output compare registers. The output compares work by producing an output of a pre-programmed level and/or an interrupt when the value in the counter matches the value stored in one of the output compare registers. The events that can be initiated by these compare matches are transitions on an output pin (to high, to low or toggle), a CPU interrupt (used for software timing functions), clearing the counter and the triggering of a DMA channel.

Channels 3 and 4 allow to produce a pulse with duration down to one clock cycle (62.5ns at 16MHz).

A combination of the output compare function with the DMAC and the TPC (a peripheral explained later) can be used to control stepper motors very easily.

Input Capture Functions

The ITU provides up to 10 channels of input capture. In this mode the timer can be set up so that a transition on an input pin causes the value currently in the count register to be transferred into a capture register, thus time stamping that particular event. The ITU can be set to capture rising edges, falling edges and either of these. If required the timer unit can also clear the timer when the programmed external event occurs.

The two buffer registers (BRA and BRB) provided in timer channels 3 and 4 can be used to buffer input time stamps. This allows events which occur very close together to be time stamped using one capture pin. This feature can also be used to measure the width of an incoming pulse, by programming the capture input to be triggered on both the rising and falling edges.

By using input captures to measure the timing of external signals, very accurate measurements of variables such as frequency can be taken. The user can be sure that the accuracy of the measurement is not compromised by interrupt response time, as it is entirely a hardware driven facility.

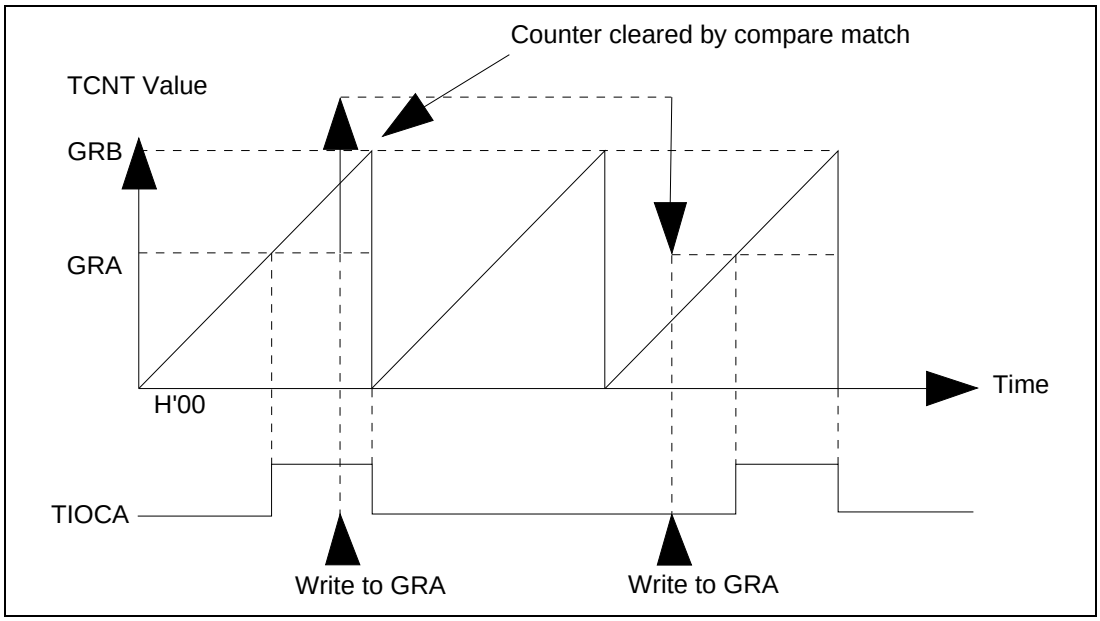
It is also possible to initiate DMA transfers when an input capture event occurs. This allows time stamps to be automatically placed in memory via the DMA controller, without needing to interrupt the CPU.

Timer Synchronisation

To allow timer channels in the ITU to be used in synchronisation, it is possible to set up two or more timers so that they are simultaneously written to via software and cleared by compare matches or input captures. When timer channels are put into this mode then their input and output events are also synchronised.

PWM Operating Modes

The PWM (Pulse Width Modulation) modes of the ITU are described in the following diagrams.



Standard PWM Mode

Each timer channel can be programmed to produce a single phase PWM output.Thus the ITU can output up to five separate channels of PWM. In this mode GRA controls the time when the pin goes high, and GRB when the pin goes low. Either GRA or GRB can be set to clear the counter, thus setting the frequency of the PWM output. The table below shows the PWM frequencies which can be obtained against device clock speed and output resolution.

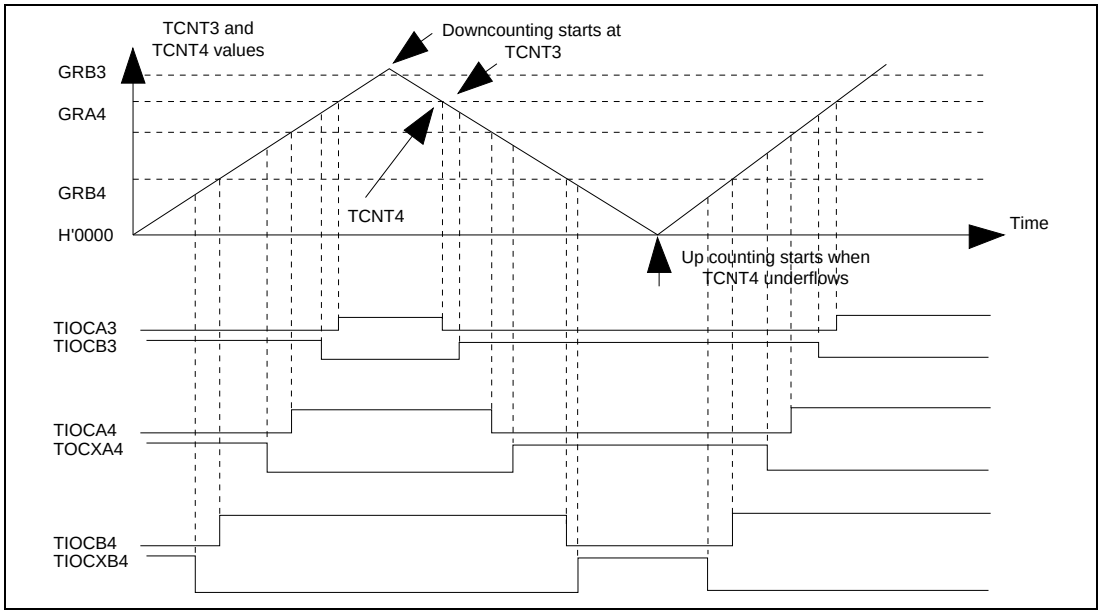
PWM Resolution	16 MHz	12 MHz	10 MHz	8 MHz	6 MHz
14-bit	976.5 Hz	732.7 Hz	610 Hz	488 Hz	365 Hz
12-bit	3.9 KHz	2.9 KHz	2.4 KHz	2 KHz	1.5 KHz
10-bit	15.6 KHz	11.7 KHz	9.7 KHz	7.8 KHz	5.8KHz
9-bit	31.2 KHz	23.4 KHz	19.5 KHz	15.6 KHz	11.7 KHz
8-bit	62.4 KHz	46.9 Khz	39 KHz	31.3 KHz	23.4 KHz
7-bit	124.8 KHz	93.8 KHz	78 Khz	62.5 KHz	46.8 KHz

AC Motor Control Outputs

To provide the PWM signals required to drive AC machines, the ITU provides two further PWM modes: Complementary 6-phase PWM and Reset Synchronised 6-phase PWM. The main differences between these two modes are the transition points for the outputs and the provision of dead time between the phase outputs.

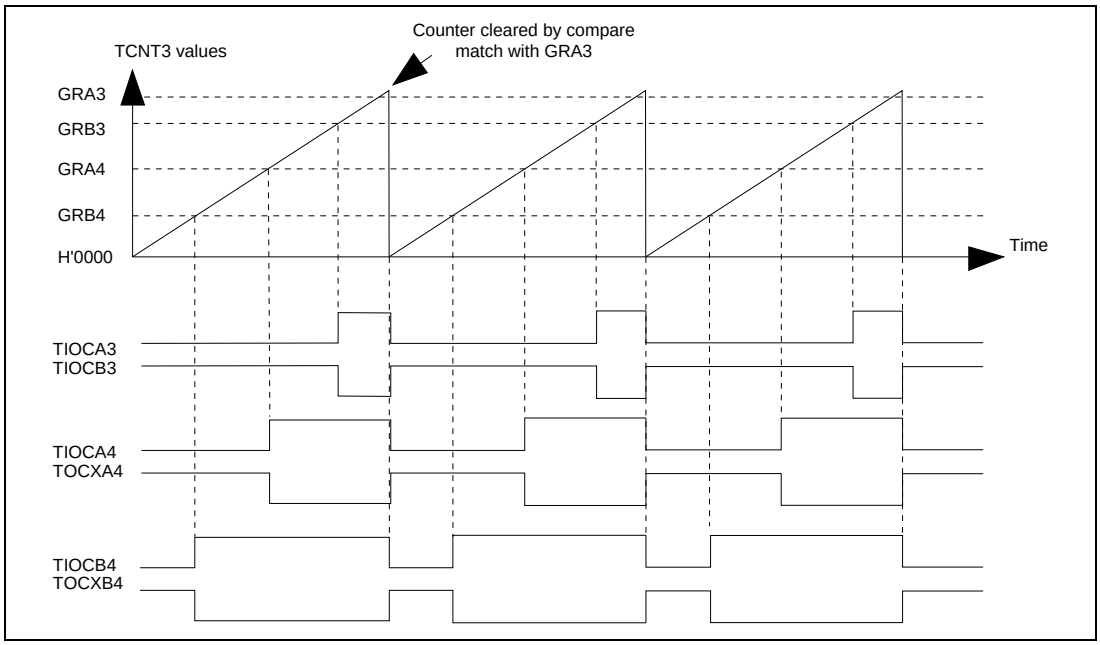
Complementary 6-Phase PWM

In this mode channels 3 and 4 are combined to produce three pairs of non-overlapping PWM waveforms, as described in the figure “H8/300H PWM modes”. As can be seen from this figure, in this mode TCNT3 and TCNT4 act as up/down counters, counting down from the point set by the compare match TCNT3 and GR3 and counting up from the point at which TCNT4 underflows.



The PWM waveforms are produced from compare matches with the general registers GRB3, GRA4 and GRB4. Using this mechanism, only three registers need to be reloaded to change the modulation ratio, keeping the CPU overhead to a minimum.

In an AC motor control system, it is necessary to insert some deadtime between the switching of the complementary phases to ensure that no short circuit condition occurs through the two drivers. The ITU supports this function using the difference in value between the two timer channels used. Thus a totally programmable deadtime is supported in this mode.

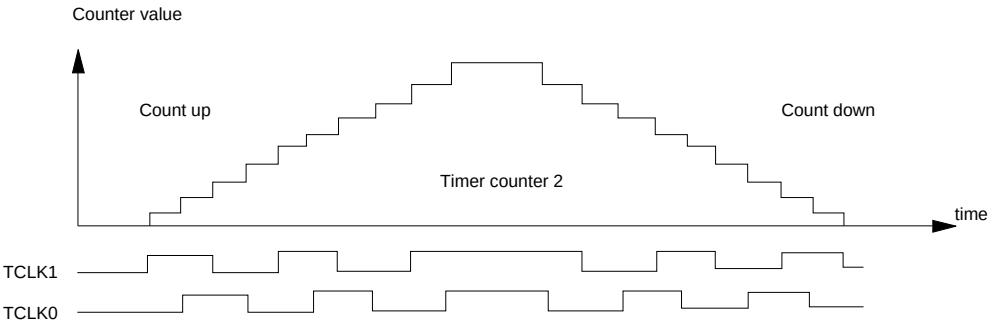


Reset Synchronised PWM

This output mode is shown in the figure “H8/300H PWM modes” and it provides three pairs of complementary PWM waveforms, all having one common waveform transition point. In this mode TCNT3 counts up until it is cleared by a match with GRA3. The output pins toggle at compare matches between GRB3, GRA4, GRB4 and TCNT3 and they all toggle when TCNT3 is cleared.

Phase Counting Mode

This mode finds use in servo control systems, where the position and speed feedback comes from a 2- phase quadrature encoder. In this type of encoder the waveforms output change their phase relationship depending on the direction of motion.



TCLK1 and TCLK0 are signals provided by a 2-phase quadrature encoder. The value of timer counter 2 reflects the position. Note: This mode is available only for timer counter 2

	Countdown				Countup			
TCLK0	↑	High	↓	Low	↑	Low	↓	High
TCLK1	Low	↑	High	↓	High	↑	Low	↓

Count condition: Count all edges

By utilising the phase counting mode on the ITU, TCNT2 will count up or down depending on the phase of the incoming signals. Therefore, the value of TCNT2 will reflect the positional changes experienced by the encoder.

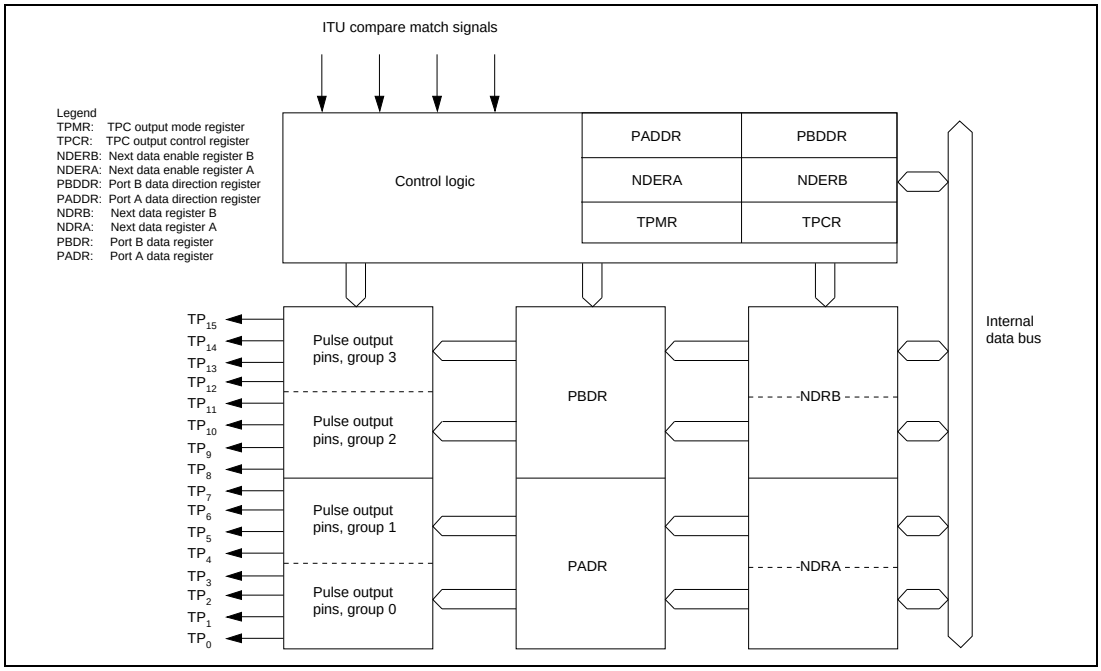
This facility removes the requirement for extra hardware or interrupt handlers for position monitoring. In this mode the comparators of channel 2 can also be used to generate interrupts, for example when a certain position is reached.

ITU Interrupts

The ITU can produce a total of 15 interrupts. This comprises 3 per channel, one for each general register and an overflow. Each interrupt has its own vector, making it unnecessary to poll flags, thus enabling fast reaction by the CPU.

Timing Pattern Controller (TPC)

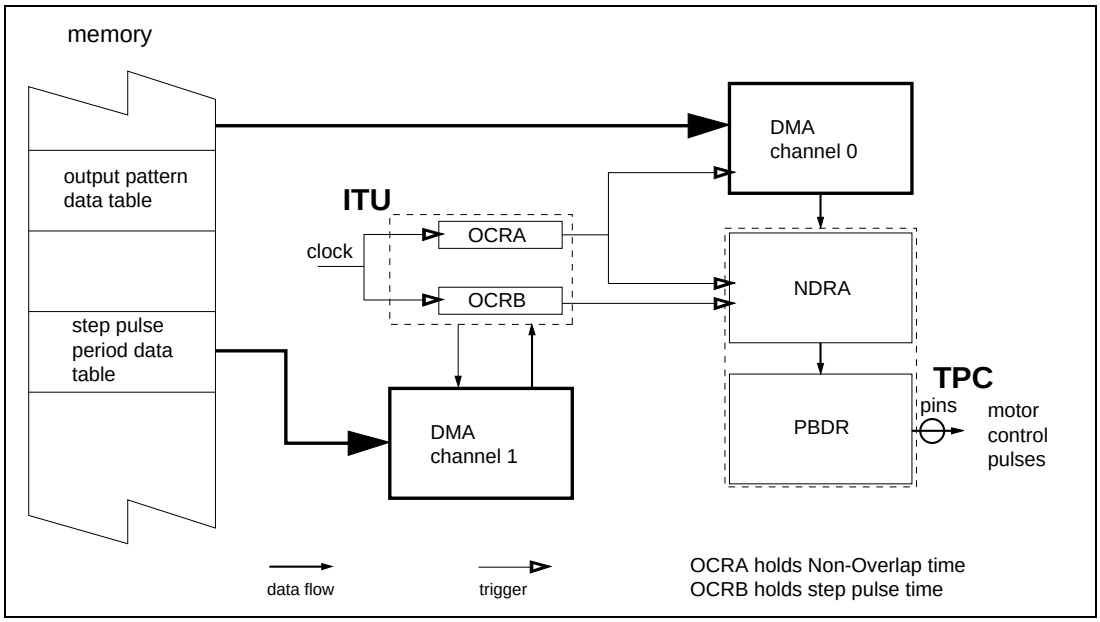
This peripheral can simultaneously output up to 16 waveforms, using a strobe signal produced by the ITU. It is useful for generating the necessary waveforms for driving stepper motors.



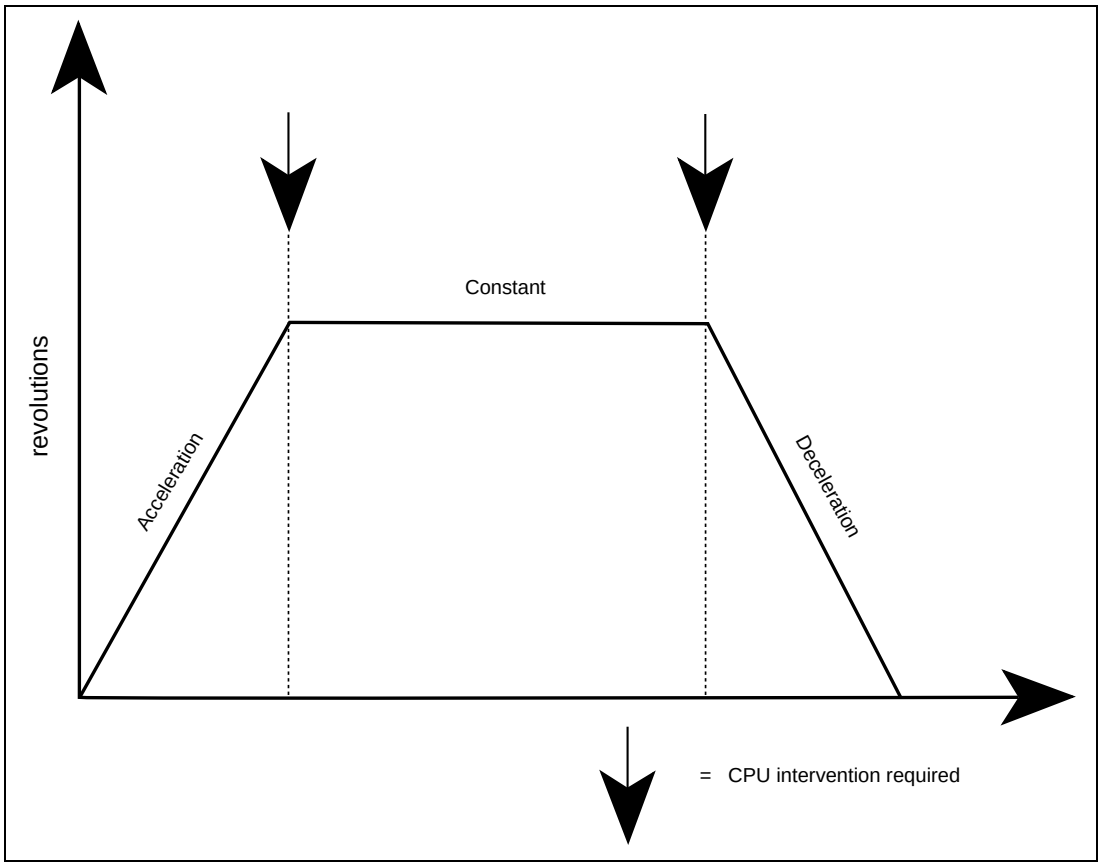
The main registers of the TPC are the next data registers (NDRA~D). These registers are each 4-bits wide and contain the next data which will be transferred into the port data registers PADR and PBDR. This data is transferred using a strobe signal generated by the selected timer channel output compare. A separate timer channel can be specified to synchronise each group of 4-bits. One of the main applications for this peripheral is the control of multiple stepper motors; if used in conjunction with the on-chip DMA controllers then this control can be performed with very little CPU supervision.

Stepper Motor Control with the TPC

A diagram showing how stepper motor control can be performed using the ITU, TPC and DMAC is shown below. In this example a two phase stepper motor is being driven, using complementary transistors. It is therefore necessary to provide a dead time between the switching of the phases to eliminate any short circuit conditions between the high side and low side drivers.



Using compare matches from the ITU to stimulate the DMAC, new pattern data is provided to the TPC. This pattern data represents the next phase drive pattern required, and is stored in a memory table. The DMAC uses its memory to I/O function to transfer this data on each compare match. The TPC also uses the stimulus from the ITU to transfer the contents of the NDR to the port. Using a TPC mode where transitions on the port from 0 to 1 (i.e. switching on a phase) are only made on compare match A, a dead time, equal to the value in GRA, is inserted.



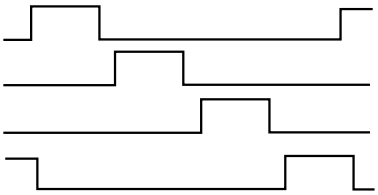
When controlling a stepper motor, providing the phase patterns onto the port pins is only part of the story. It is also necessary to modify the time between new patterns being output to allow acceleration and deceleration of the motor as shown by the velocity profile.

When the TPC, ITU and DMAC are working together, the acceleration and deceleration phases, as well as the steady speed phase can be controlled with minimal CPU overhead. The CPU need only get involved when a transition from one phase to another is made.

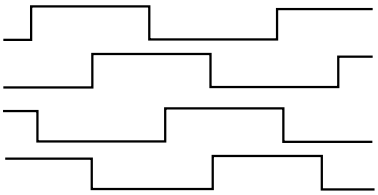
This is achieved by using a second memory to I/O DMA channel to reload the timer compare register after each new pattern has been output. Again the DMAC can take the next step period data from a table of values stored in the memory of the system. Therefore, by providing a table of increasing or decreasing values the motor can be decelerated or accelerated with no CPU intervention.

The flexibility of the H8/300H's stepper motor control functions will even allow multiple motors to be controlled simultaneously by one device. The figure shows examples of the stepper motors which can be controlled, and contrasts the resources required with the amount of resource available on the H8/3003.

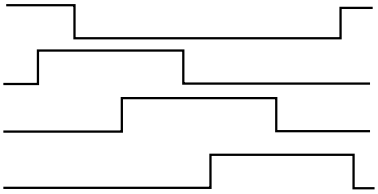
2-Phase 1-Excitation



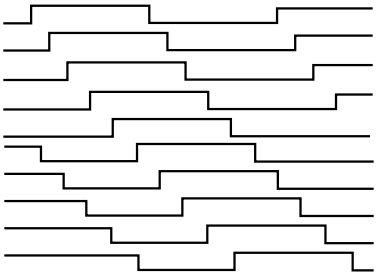
2-Phase 2-Excitation



2-Phase 1-2-Excitation



5-Phase 5-Excitation



	2-Phase 1-Excitation	2-Phase 2-Excitation	2-Phase 1-2-Excitation	5-Phase	H8 / 3003 Total
DMAC	2ch	2ch	2ch	2ch	8ch
TPC	4-bits	4-bits	4-bits	5-bits	16-bits
ITU	1ch	1ch	1ch	1ch	5ch

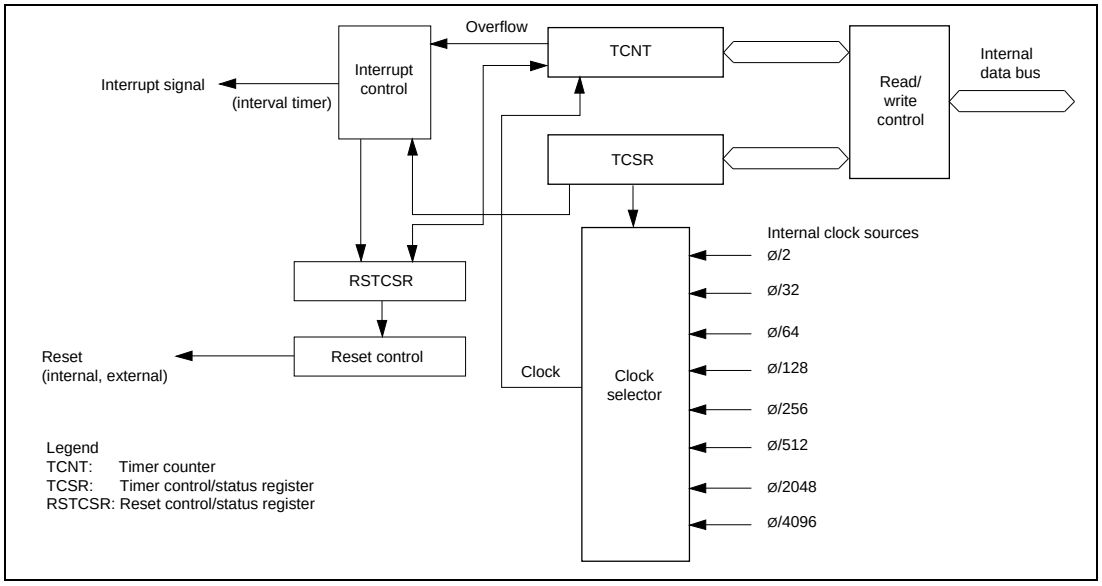
	2 -Phase I-Excitation	2-Phase 2-Excitation	2-Phase 1-2-Excitation	5-Phase	H8 / 3003 Total
DMAC	2 ch	2 ch	2 ch	2 ch	8 ch
TPC	4-bits	4-bits	4-bits	5-bits	16-bits
ITU	1 ch	1 ch	1 ch	1 ch	5 ch

Watchdog Timer (WDT)

Often, a watchdog timer is a very important feature in any embedded application. It is used to ensure that any “mishap” in the system (such as a noise induced software crash) is rectified as quickly as possible.

The principle behind a watchdog timer is very simple - a counter is constantly counting upwards, and correctly operating software ensures that this counter never overflows by continuously resetting the count. If the software crashes and the counter overflows, the watchdog “barks” and sends some stimulus to the microcontroller (normally a reset) to restart system operations in a controlled manner.

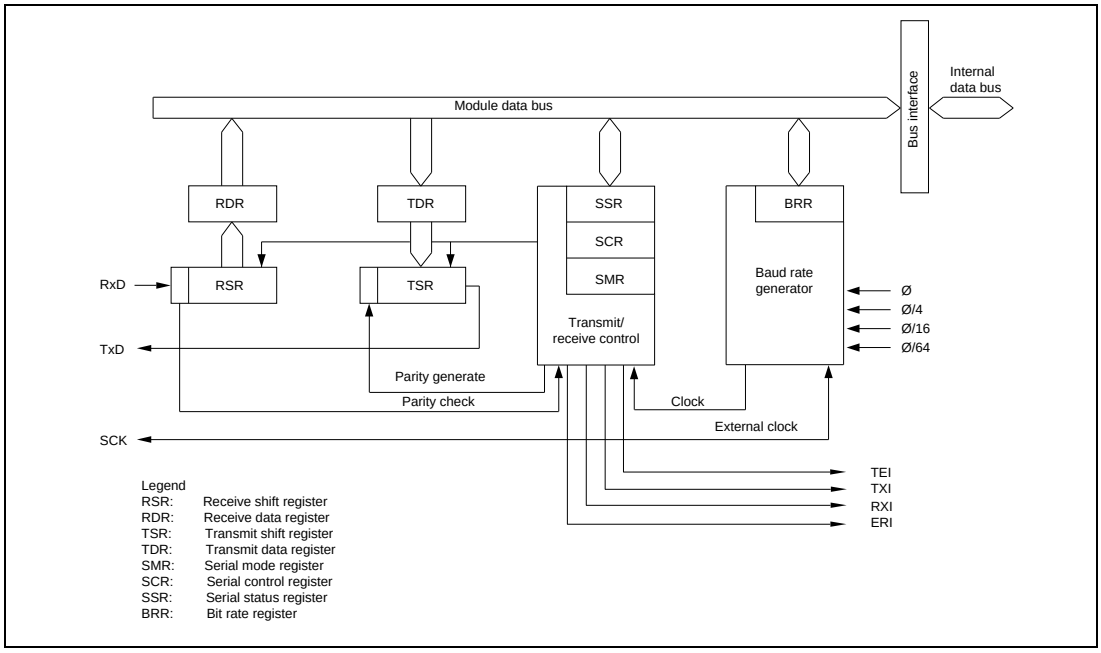
All H8/300H devices are equipped with a timer, which can be used either as a watchdog or as an interval timer. Its “bark” is a reset if it is used as a watchdog.



Serial Communications Interface (SCI)

This form of communication has many uses in microcontroller applications, such as interdevice communications, diagnostics, host communication, and even as an interface to peripherals. All H8/300H devices are equipped with at least one and very often two channels of serial communication interface (SCI). These channels can be used for either synchronous or asynchronous communications. This type of SCI is standard across the H8/300H range.

As shown in the block diagram each SCI channel has its own integral Baud rate generator, so many Baud rates can be produced from the microcontroller's internal clock, without using any other timers.



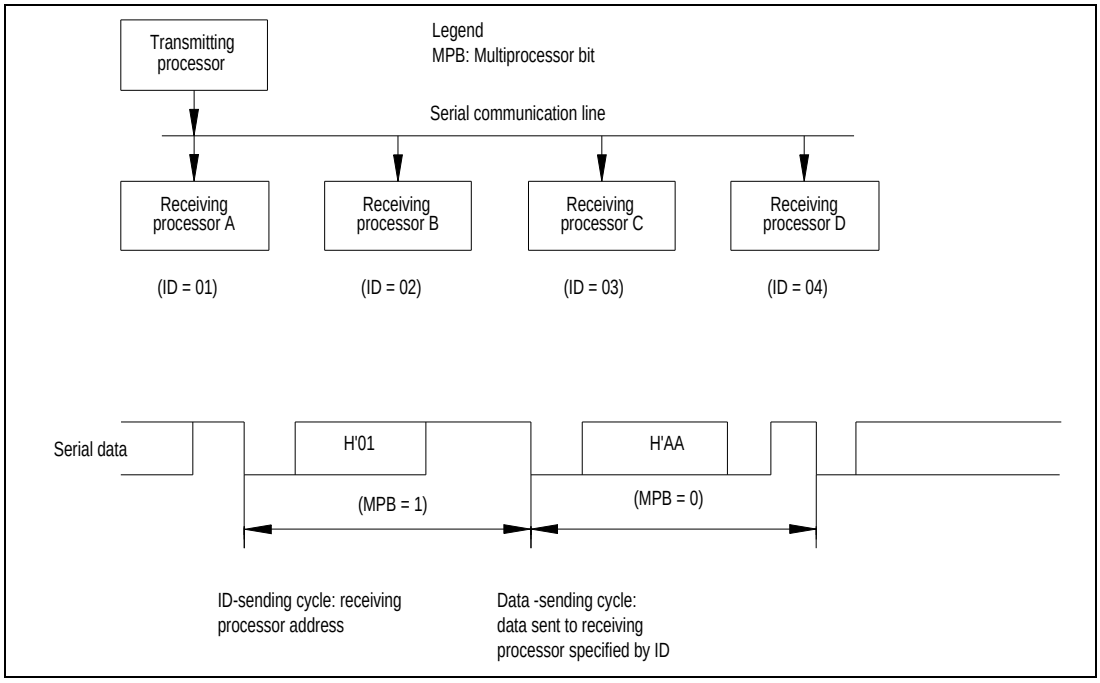
As well as using the integral Baud rate generator, each SCI can be configured to use an external serial clock.

To allow “back to back” transmission or reception of serial data, the SCI has double buffered transmit and receive shift registers.

The H8/300H serial ports also support multiprocessor communications using a master slave configuration in addition to the standard modes.

In this mode, communication between devices is performed using an additional multiprocessor bit (MPB) which is added to the data transmitted. This bit is used to differentiate between data

frames and address frames. Thus, any frame sent from the master with the MPB set to one can be used to activate the required slave.

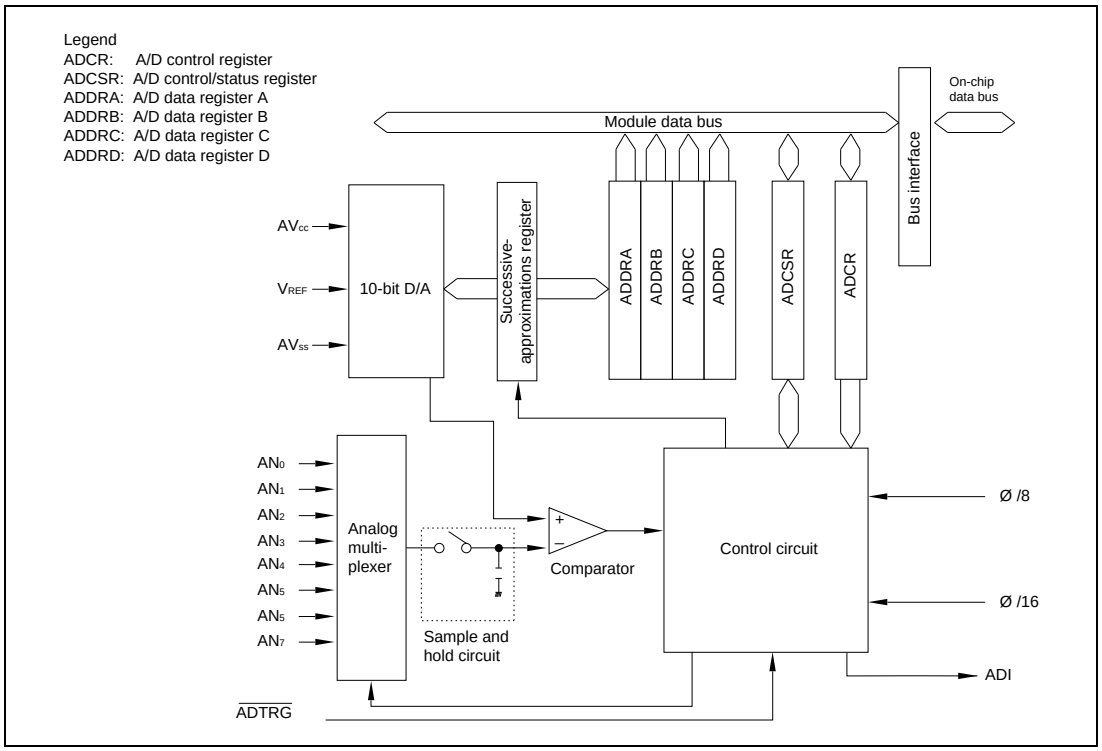


Slave devices on this network will only produce a receive interrupt when a frame is received with the MPB set, so the interrupt handler can check the address which has been transmitted.

Receive data errors are trapped using three error conditions -overflow,framing and parity. These three errors are indicated via one interrupt vector and three status flags in the serial status register.

Analogue to Digital Converter (ADC)

In many microcontroller based systems, some way of measuring analogue electrical values is necessary.With this in mind, all members of the H8/300H family are equipped with a 10-bit A/D converter.



The converter works using a successive approximation algorithm and conversions take 138 states (or 8.6µs if ϕ - 16MHz).

Up to 8 inputs can be converted; this is achieved using an 8 channel multiplexer between the input port and the A/D. A sample and hold capacitor is used to ensure that once a conversion begins, a change of the input value will not be reflected in a different conversion result.

As well as being able to perform single conversion (where only one channel is converted), the H8/300H A/D converter can also be used to scan up to four channels. To support this mode of operation, four A/D result registers are provided. Once the scan mode is selected, each channel specified is converted sequentially with the conversion value being stored in the appropriate result register. This mode of operation allows the user software to sample the current analogue value of an input by simply reading the appropriate data register.

Digital to Analogue Converter (DAC)

The H8/304X device incorporates an 8-bit digital to analogue converter which has a maximum conversion time of 6.2ms (ϕ = 16MHz). Two outputs are provided and multiplied with the analogue to digital inputs. The output voltage range is from 0V through to the A/D reference voltage.

H8/300H Summary

In the previous chapters we have given an overview as to why H8/300H gives the designers the technical benefits that are demanded by today's applications. Their purpose was to give you enough insight into the features of the H8/300H family to see how you can benefit from the performance, the rich set of peripherals and the memory options available. To summarize this:

- **high performance CPU** with architecture tailored for high level language software development
- many **low power,** low voltage options
- memory line up from 16K to 256K ROM and 512Byte to 4K RAM, also version with 128K Flash
- very **powerful and sophisticated peripherals** designed to offload the CPU and hence to increase system performance
- **full European technical and tool support**

The following chapters are provided for you to select the right derivative for your application from the H8/300H family.

Selection Guide (ROM-less)

Type No.	H8/3001	H8/3002	H8/3003	H8/3004	H8/3005
RAM (byte)	512	512	512	2k	4k
Vcc (V) / clock (MHz)	4.5-5.5/16 3.15-5.5/13 3.0-5.5/10 2.7-5.5/8	4.5-5.5/16 3.0-5.5/10 2.7-5.5/8	4.5-5.5/16 3.0-5.5/10 2.7-5.5/8	4.5-5.5/16 4.5-5.5/18 3.0-5.5/10 2.7-5.5/8	4.5-5.5/16 4.5-5.5/18 3.0-5.5/10 2.7-5.5/8
Address space (byte)	16M	16M	16M	16M	16M
External data bus (bit)	8/16	8/16	8/16	8	8
ITU (16-bit timer)	5	5	5	5	5
Watchdog timer	-	1	1	1	1
DMAC Memory to/from I/O Memory to Memory	 - -	 4 2	 8 4	 - -	 - -
SCI (async/sync)	1	2	2	1	1
TPC (bit)	12	16	16	-	-
ADC 10 bits External trigger Input	 4 -	 8 Yes	 8 Yes	 8 Yes	 8 Yes
Refresh controller	-	Yes	Yes	-	-
Chip select pins	-	4	8	-	-
Interrupts Internal External level	20 4	30 7	34 9	21 6	21 6
I/O	32	46	58	32	32
Package	FP-80A TFP-80C	FP-100A* FP-100B TFP-100B	FP-112 TFP-80C	FP-80A TFP-80C	FP-80A

* For availability of FP-100A please contact Hitachi or an authorized distributor

Selection Guide (ROM/ZTAT/Flash)

Type No.	H8/3030	H8/3031	H8/3032	H8/3040	H8/3041	H8/3042
ROM (byte)	16 k	32 k	64 k	32 k	48 k	64 k
RAM (byte)	512	1 k	2 k	2 k	2 k	2 k
ZTAT(OTP)	-	-	Yes	-	-	Yes
Vcc (V) / clock (MHz) Note 1	5/16 2.7-5.5/8	5/16 2.7-5.5/8	5/16 2.7-5.5/8	5/16 2.7-5.5/8	5/16 2.7-5.5/8	5/16 2.7-5.5/8
Address space (byte)	1 M	1 M	1 M	16M	16M	16M
External data bus (bit)	8	8	8	8/16	8/16	8/16
ITU (16-bit timer)	5	5	5	5	5	5
Watchdog timer	1	1	1	1	1	1
DMAC Memory to/from I/O Memory to Memory	 - -	 - -	 - -	 4 2	 4 2	 4 2
SCI (async/sync)	1	1	1	2	2	2
TPC (bit)	16	16	16	16	16	16
ADC 10 bits External trigger Input	 8 Yes	 8 Yes	 8 Yes	 8 Yes	 8 Yes	 8 Yes
8-bit DAC (channels)	-	-	-	2	2	2
Refresh controller	-	-	-	Yes	Yes	Yes
Chip select pins	-	-	-	4	4	4
Interrupts Internal External level	 21 6	 21 6	 21 6	 30 7	 30 7	 30 7
I/O	63	63	63	78	78	78
Package	FP-80A TFP-80C	FP-80A TFP-80C	FP-80A TFP-80C	FP-100A* FP-100B TFP-100B	FP-100A* FP-100B TFP-100B	FP-100A* FP-100B TFP-100B

Note 1: 5V with±10% tolerance

* For availabilty of FP-100A please contact Hitachi or an authorized distributor

Now available up to 18MHz at 5V:

H8/3035 256K ROM/OTP

H8/3034 192K ROM

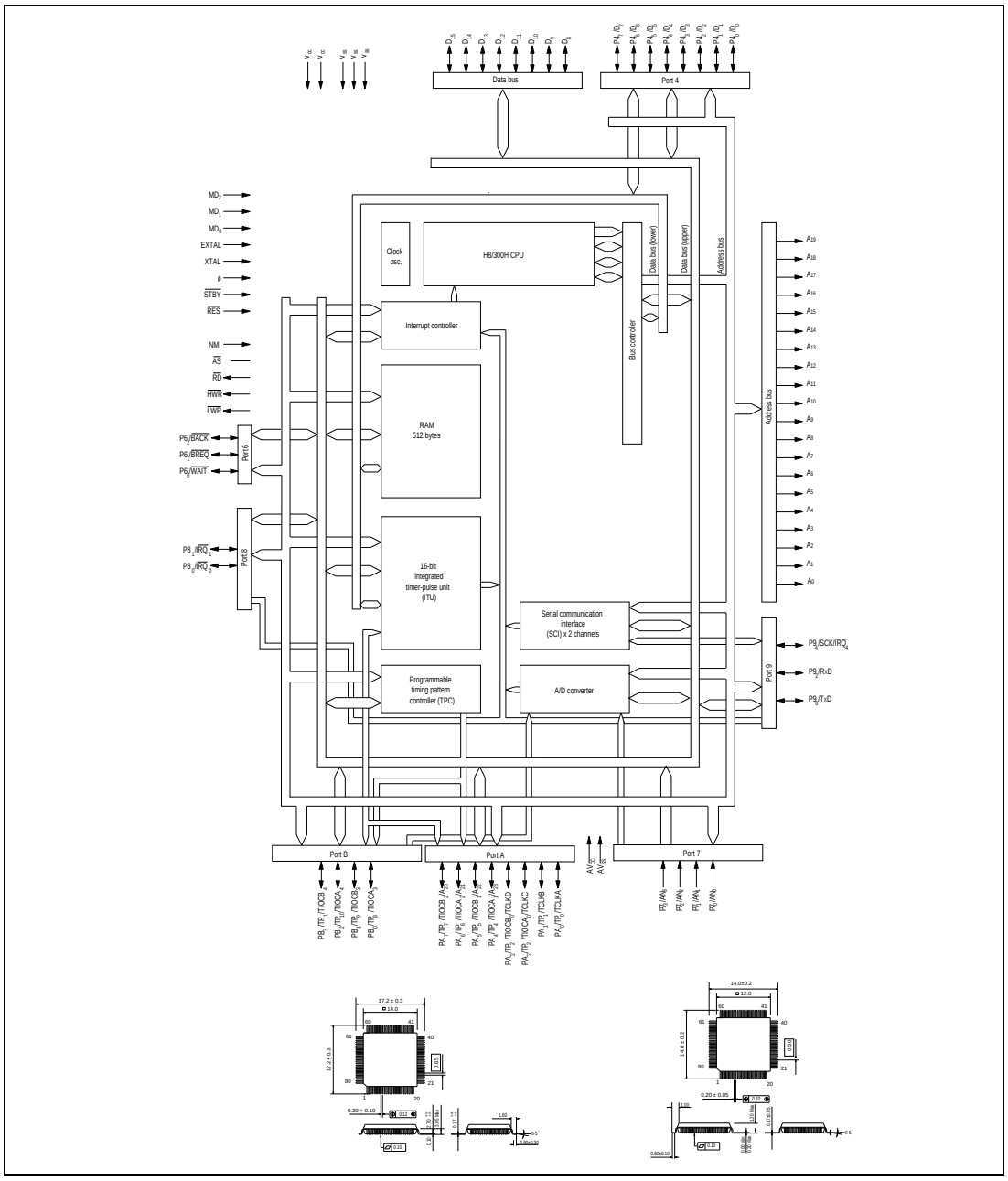
H8/3033 128K ROM

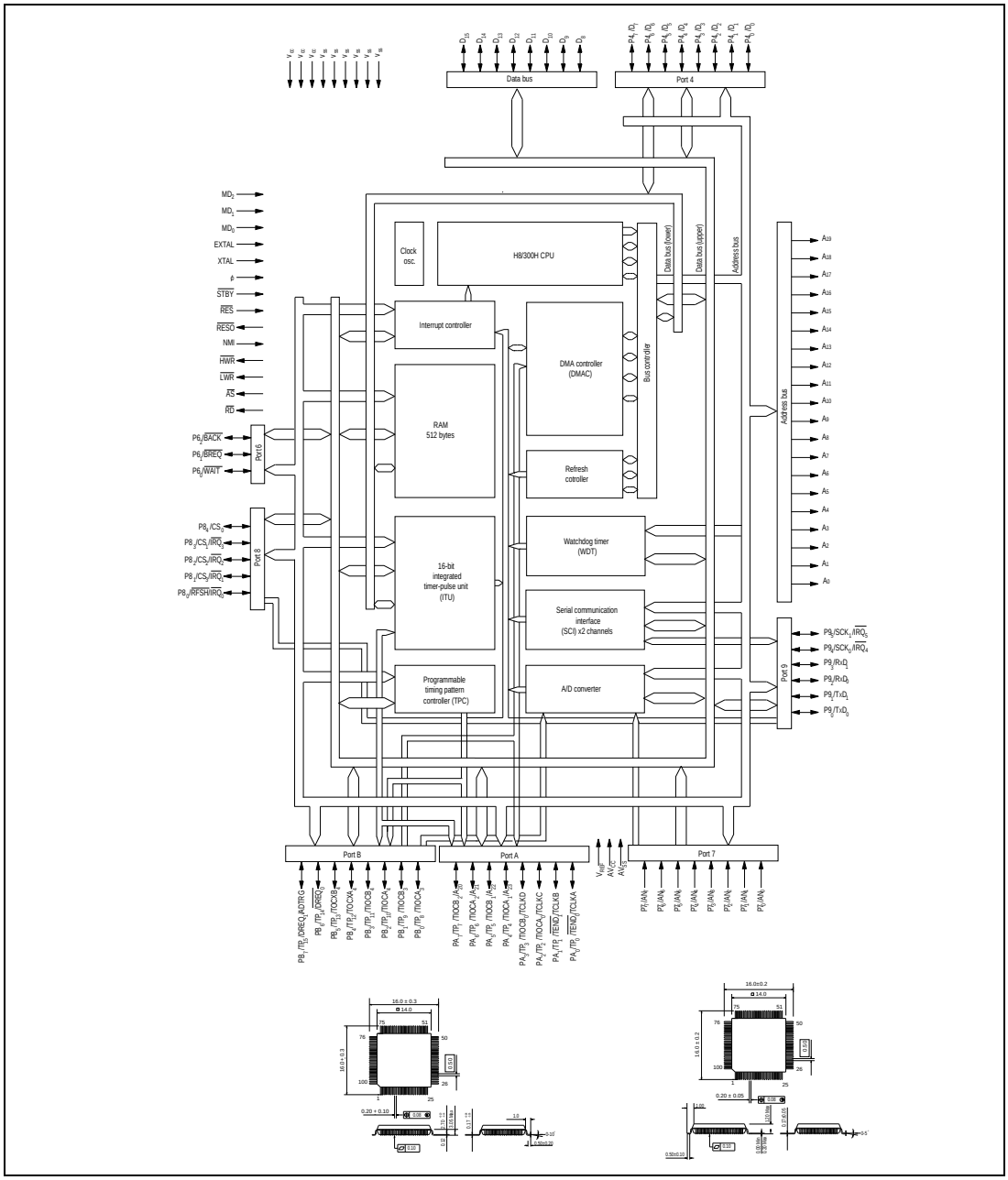
all with 4K RAM

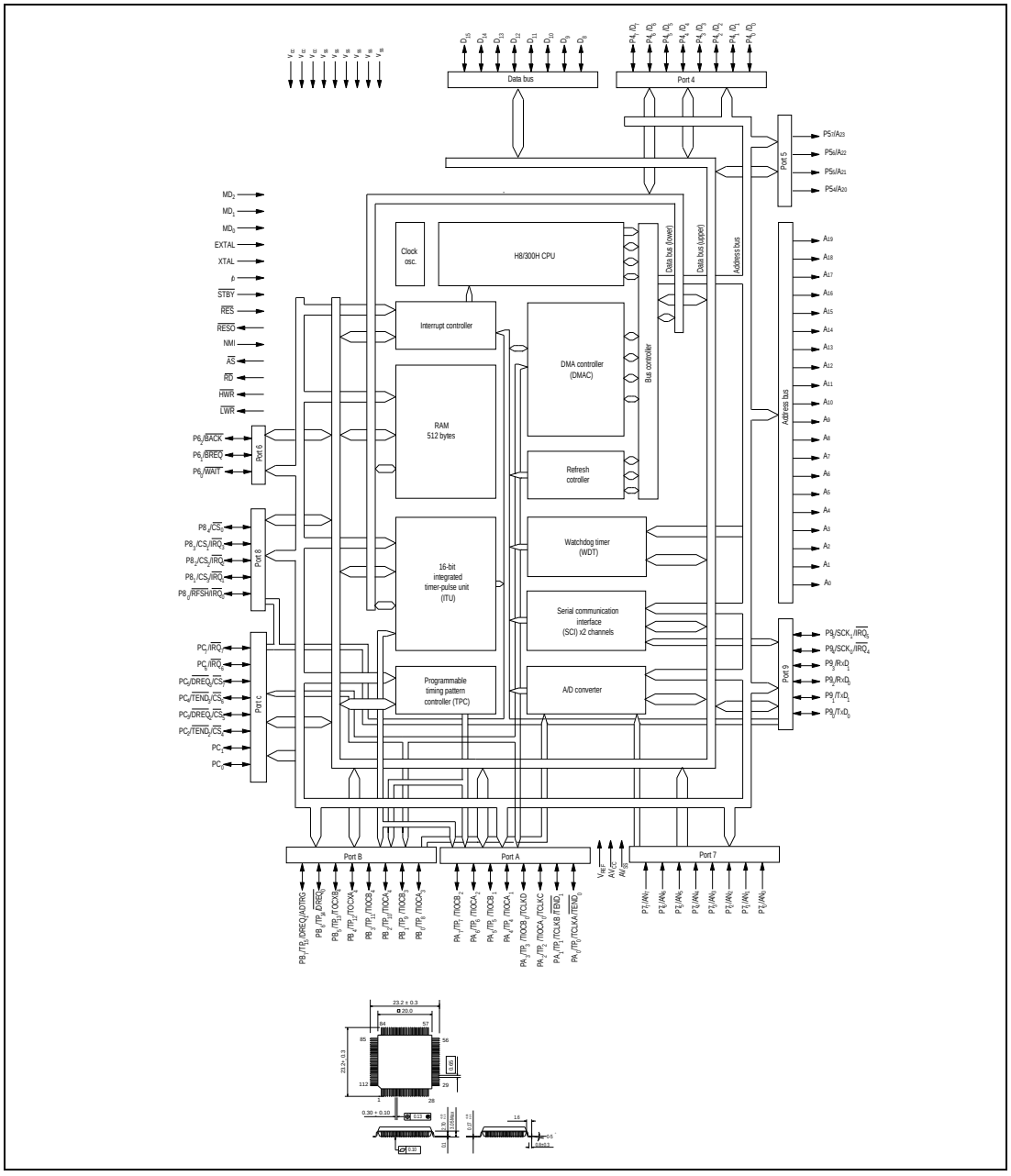
Selection Guide (ROM/ZTAT/Flash)

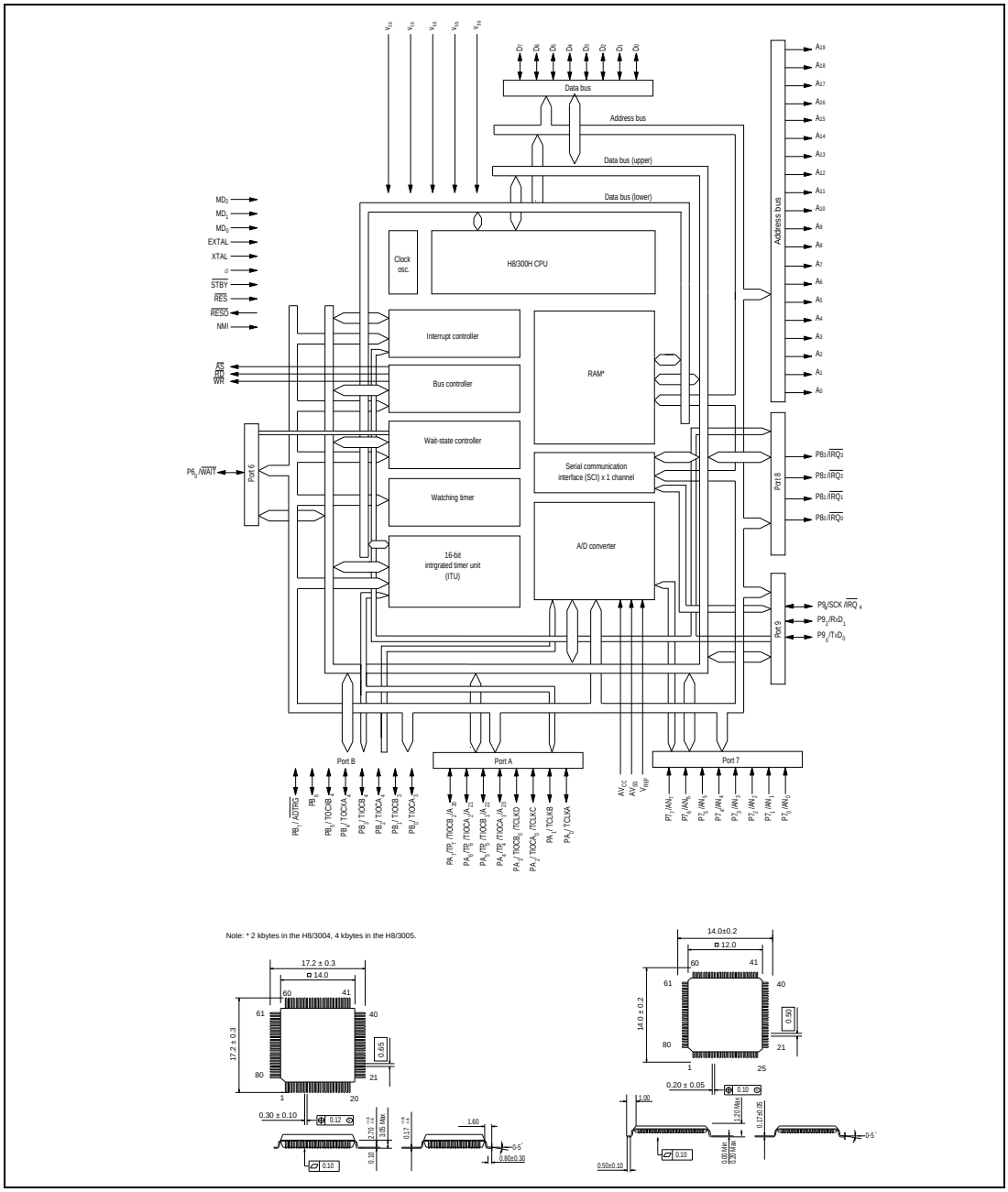
Type No.	H8/3044	H8/3045	H8/3047	H8/3048	H8/3048F
ROM (byte)	32 k	64 k	96 k	128 k	128 k
RAM (byte)	2 k	2 k	4 k	4 k	4 k
ZTAT(OTP)	-	-	-	Yes	Flash
Vcc (V) / clock (MHz) Note 1	5/16 5/18 3.3/13 2.7/8	5/16 5/18 3.3/13 2.7/8	5/16 5/18 3.3/13 2.7/8	5/16 5/18 3.3/13 2.7/8	5/16 2.7-5.5/8
Address space (byte)	16M	16M	16M	16M	16M
External databus (bit)	8/16	8/16	8/16	8/16	8/16
ITU (16-bit timer)	5	5	5	5	5
Watchdog timer	1	1	1	1	1
DMAC Memory to/from I/O Memory to Memory	 4 2	 4 2	 4 2	 4 2	 4 2
SCI (async/sync)	2	2	2	2	2
TPC (bit)	16	16	16	16	16
ADC 10 bits External trigger Input	 8 Yes	 8 Yes	 8 Yes	 8 Yes	 8 Yes
8-bit DAC (channels)	2	2	2	2	2
Smart card I/F	Yes	Yes	Yes	Yes	Yes
Refresh controller	Yes	Yes	Yes	Yes	Yes
Chip select pins	8	8	8	8	8
Interrupts Internal External level	 30 7	 30 7	 30 7	 30 7	 30 7
I/O	78	78	78	78	78
Package	FP-100B TFP-100B	FP-100B TFP-100B	FP-100B TFP-100B	FP-100B TFP-100B	FP-100B TFP-100B

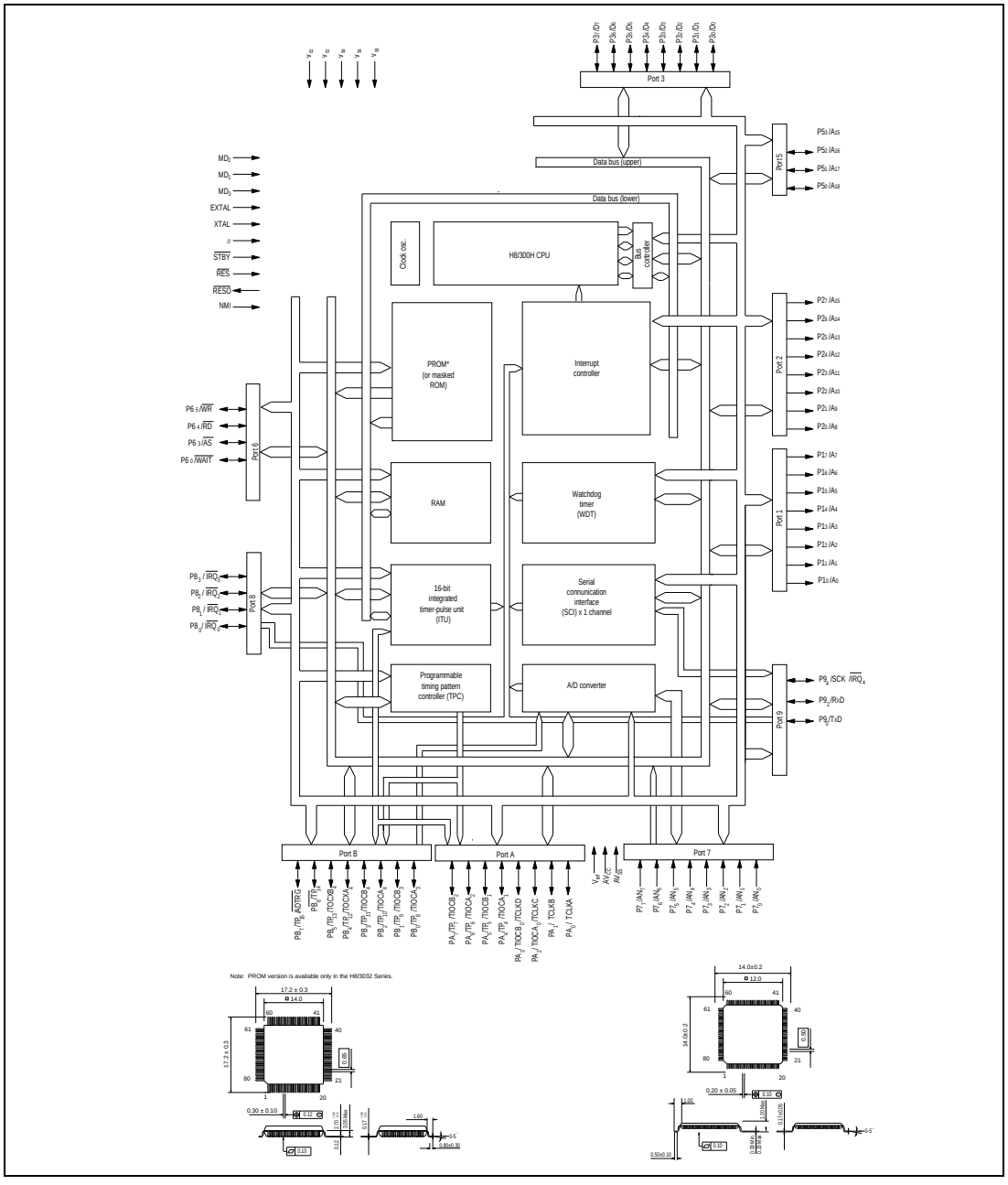
Note 1: 5V with±10% tolerance

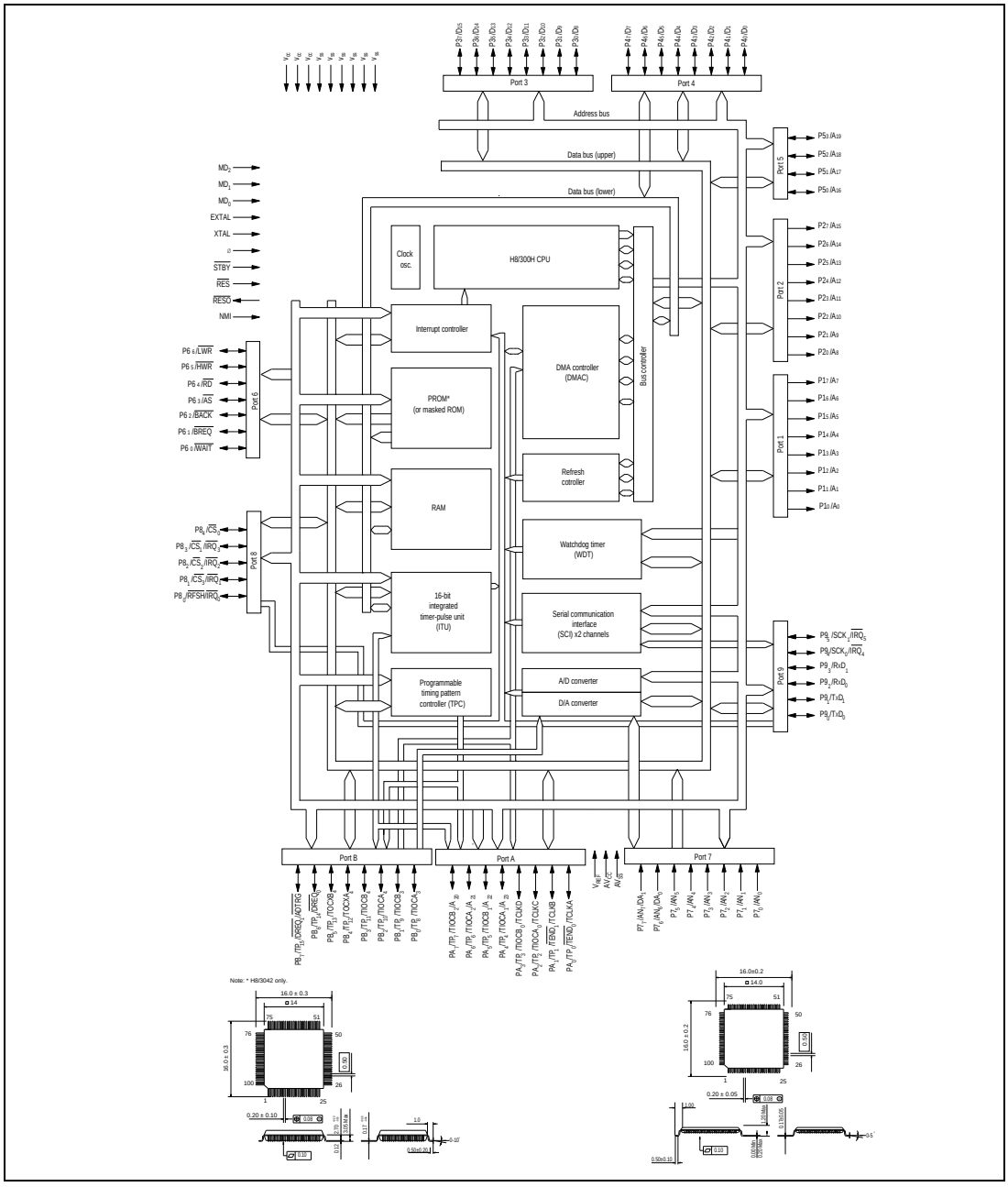

Data bus
Port 4
Clock osc.
H8/300H CPU
Interrupt controller
Bus controller
Data bus (lower)
Data bus (upper)
Address bus
RAM 512 bytes
16-bit integrated timer-pulse unit (ITU)
Serial communication interface (SCI) x 2 channels
Programmable timing pattern controller (TPC)
A/D converter
Port 6
Port 8
Port 9
Port B
Port A
Port 7

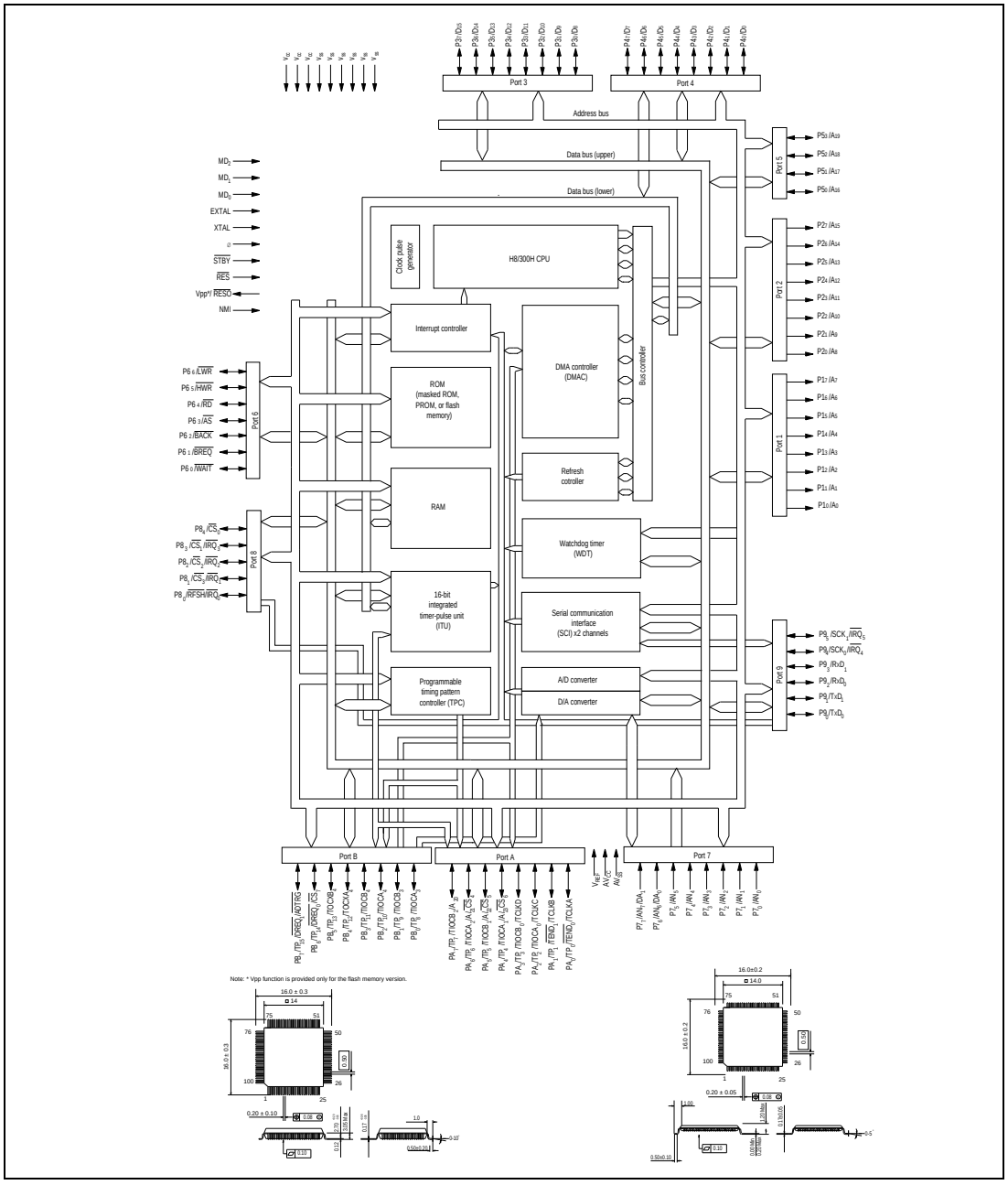

Data bus
Port 4
Clock osc.
H8/300H CPU
Interrupt controller
DMA controller (DMAC)
Bus controller
RAM 512 bytes
Refresh controller
Watchdog timer (WDT)
16-bit integrated timer-pulse unit (ITU)
Serial communication interface (SCI) x2 channels
Programmable timing pattern controller (TPC)
A/D converter
Data bus (lower)
Data bus (upper)
Address bus
Port 6
Port 8
Port 9
Port B
Port A
Port 7
MD2
MD1
MD0
EXTAL
XTAL
φ
STBY
RES
RESO
NMI
HWR
LWR
AS
RD
P66/BACK
P65/BREQ
P64/WAIT
P87/CS7
P86/CS6/IRQ7
P85/CS5/IRQ6
P84/CS4/IRQ5
P83/RFSH/IRQ4
P97/SCK1/IRQ3
P96/SCK0/IRQ2
P95/RxD1
P94/RxD0
P93/TxD1
P92/TxD0
P17/AN7
P16/AN6
P15/AN5
P14/AN4
P13/AN3
P12/AN2
P11/AN1
P10/AN0
AVCC
AVSS
VCC
VSS


Data bus
Port 4
Port 5
Clock osc.
H8/300H CPU
Data bus (lower)
Data bus (upper)
Address bus
Interrupt controller
DMA controller (DMAC)
Bus controller
RAM 512 bytes
Refresh controller
Watchdog timer (WDT)
16-bit integrated timer-pulse unit (ITU)
Serial communication interface (SCI) x2 channels
Programmable timing pattern controller (TPC)
A/D converter
Port 6
Port 8
Port C
Port 9
Port B
Port A
Port 7

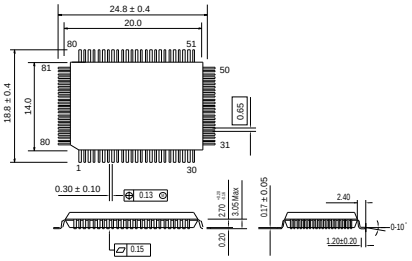

Data bus
Address bus
Data bus (upper)
Data bus (lower)
Clock osc.
H8/300H CPU
Interrupt controller
Bus controller
Wait-state controller
Watching timer
16-bit intrgrated timer unit (ITU)
RAM*
Serial communication interface (SCI) x 1 channel
A/D converter
Port 6
Port 8
Port 9
Port A
Port 7
Address bus
Note: * 2 kbytes in the H8/3004, 4 kbytes in the H8/3005.


Port 3
Data bus (upper)
Data bus (lower)
Port 5
Port 2
Port 1
Port 9
Port 6
Port 8
Port B
Port A
Port 7
Clock osc.
H8/300H CPU
Bus controller
PROM*
(or masked
ROM)
Interrupt
controller
RAM
Watchdog
timer
(WDT)
16-bit
integrated
timer-pulse unit
(ITU)
Serial
connunication
interface
(SCI) x 1 channel
Programmable
timing pattern
controller (TPC)
A/D converter
MD2
MD1
MD0
EXTAL
XTAL
STBY
RES
RESO
NMI
Note: PROM version is available only in the H8/3032 Series.

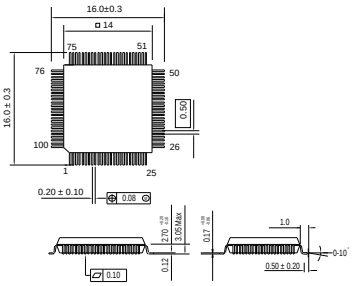

Port 3
Port 4
Address bus
Data bus (upper)
Data bus (lower)
Clock osc.
H8/300H CPU
Interrupt controller
DMA controller (DMAC)
Bus controller
PROM* (or masked ROM)
Refresh controller
RAM
Watchdog timer (WDT)
16-bit integrated timer-pulse unit (ITU)
Serial communication interface (SCI) x2 channels
A/D converter
D/A converter
Programmable timing pattern controller (TPC)
Port 1
Port 2
Port 5
Port 6
Port 8
Port 9
Port A
Port B
Port 7
MD2
MD1
MD0
EXTAL
XTAL
STBY
RES
RESO
NMI
Note: * H8/3042 only.
16.0 ± 0.3
14
16.0±0.2
14.0
0.20 ± 0.10
0.20 ± 0.05
1.0
0.50±0.10


Port 3
Port 4
Address bus
Data bus (upper)
Data bus (lower)
Port 5
Port 2
Port 1
Port 6
Port 8
Port 9
Port B
Port A
Port 7
Clock pulse generator
H8/300H CPU
Interrupt controller
DMA controller (DMAC)
Bus controller
ROM (masked ROM, PROM, or flash memory)
Refresh controller
RAM
Watchdog timer (WDT)
16-bit integrated timer-pulse unit (ITU)
Serial communication interface (SCI) x2 channels
A/D converter
D/A converter
Programmable timing pattern controller (TPC)
MD2
MD1
MD0
EXTAL
XTAL
STBY
RES
NMI
Note: * Vpp function is provided only for the flash memory version.

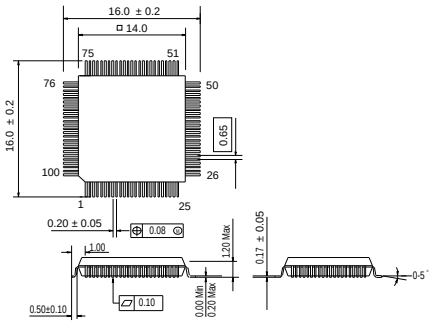
Packages



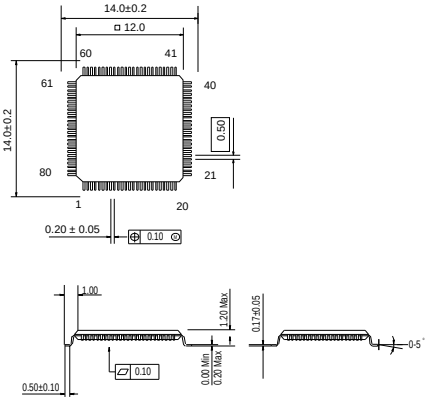
FP-100A



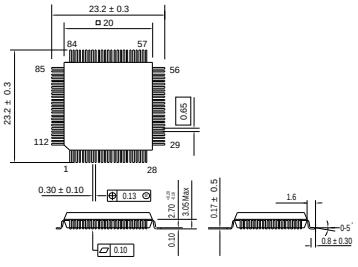
FP-100B



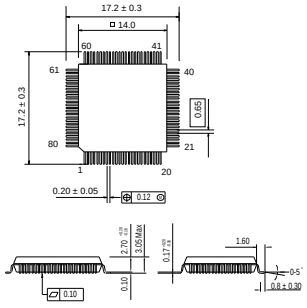
FP-100B



TFP-100B



TFP-100B



TFP-80C

Ordering Information

The following tables show the available derivatives for each series within the H8/300H family. To build the actual part name append the desired clock rate to the part name's body. Example: HD6413001F16 = H8/3001 in FP-80A package, 5V (±10%), 16mhz.

For details of the operating voltage range for ROM-less derivatives, please refer to the selection guides on page 27.

H8/3001 Body	**Package suffix**	**Package/Voltage/Temp.**	**available clock**
HD6413001	F	FP-80A/5V	16,18
	TF	TFP-80C/5V	16,18
	VF	FP-80A/low	8,10,13
	VTF	TFP-80C/low	8,10,13
	FI	FP-80A/5V/-40°C..85°C	16,18
	TFI	TFP-80C/5V/-40°C..85°C	16,18
	VFI	FP-80A/low/-40°C..85°C	8,10,13
	VTFI	TFP-80C/low/-40°C..85°C	8,10,13
H8/3002 Body	**Package suffix**	**Package/Voltage/Temp.**	**available clock**
HD6413002	F	FP-100B/5V	10,12,16,17
	FP	FP-100A*/5V	16,17
	TF	TFP-100B/5V	10,12,16,17
	VF	FP-100B/low	8,10
	VFP	FP-100A*/low	8,10
	VTF	TFP-100B/low	8,10
	FI	FP-100B/5V/-40°C..85°C	10,12,16,17
	TFI	TFP-100B/5V/-40°C..85°C	10,12,16,17
	VFI	FP-100B/low/-40°C..85°C	8,10
	VTFI	TFP-100B/low/-40°C..85°C	8,10
	FJ	FP-100B/5V/-40°C..85°C	10,12,16,17
	FQ	FP-100A*/5V/-40°C..85°C	16,17
H8/3003 Body	**Package suffix**	**Package/Voltage/Temp.**	**available clock**
HD6413003	RF	FP-112/5V	10,12,16
	TF	FP-112/5V	10,12,16
	RVF	FP-112/low	8,10
	TVF	FP-112/low	8,10
	RFI	FP-112/5V/-40°C..85°C	10,12,16
	TFI	FP-112/5V/-40°C..85°C	10,12,16
	RVFI	FP-112/low/-40°C..85°C	8,10
	TVFI	FP-112/low/-40°C..85°C	8,10
	RFJ	FP-112/5V/-40°C..85°C	10,12,16
	TFJ	FP-112/5V/-40°C..85°C	10,12,16
	Note: For H8/3003 derivatives with a package suffix starting “R”, a crystal with double frequency is required, because of an internal divider by 2.		
H8/3004 Body	**Package suffix**	**Package/Voltage/Temp.**	**available clock**
HD6413004	F	FP-80A/5V	16,18
	TE	TFP-80C/5V	16,18
	VF	FP-80A/low	8,10
	VTE	TFP-80C/low	8,10
	FI	FP-80A/5V/-40°C..85°C	16,18
	TEI	TFP-80C/5V/-40°C..85°C	16,18
	VFI	FP-80A/low/-40°C..85°C	8,10
	VTEI	TFP-80C/low/-40°C..85°C	8,10
	FJ	FP-80A/5V/-40°C..85°C	16,18
H8/3005 Body	**Package suffix**	**Package/Voltage/Temp.**	**available clock**
HD6413005	F	FP-80A/5V	16,18
	TE	TFP-80C/5V	16,18
	VF	FP-80A/low	8,10
	VTE	TFP-80C/low	8,10
	FI	FP-80A/5V/-40°C..85°C	16,18
	TEI	TFP-80C/5V/-40°C..85°C	16,18
	VFI	FP-80A/low/-40°C..85°C	8,10
	VTEI	TFP-80C/low/-40°C..85°C	8,10
	FJ	FP-80A/5V/-40°C..85°C	16,18

* For availability of FP-100A please contact Hitachi or an authorized distributor

For all of the parts with mask ROM, ZTAT or Flash the operating voltage range is 2.7..5.5V/8MHz, 3.0V..5.5V/10MHz, 3.15V..5.5V/13MHz and 4.5..5.5V otherwise.

H8/3032 (ZTAT)

Body	Package suffix	Package/Voltage/Temp.	available clock
HD6473032	F	FP-80A/5V	16,18
	TF	TFP-80C/5V	16,18
	VF	FP-80A/low	8,10
	VTF	TFP-80C/low	8,10
	FI	FP-80A/5V/-40°C..85°C	16,18
	TFI	TFP-80C/5V/-40°C..85°C	16,18
	VFI	FP-80A/low/-40°C..85°C	8,10
	VTFI	TFP-80C/low/-40°C..85°C	8,10
	FJ	FP-80A/5V/-40°C..85°C	16,18

H8/3042 (ZTAT)

Body	Package suffix	Package/Voltage/Temp.	available clock
HD6473042S	F	FP-100B/5V	10,12,16,17
	FP	FP-100A*/5V	16,17
	TF	TFP-100B/5V	10,12,16,17
	VF	FP-100B/low	8,10
	VFP	FP-100A*/low	8,10
	VTF	TFP-100B/low	8,10
	FI	FP-100B/5V/-40°C..85°C	10,12,16,17
	TFI	TFP-100B/5V/-40°C..85°C	10,12,16,17
	VFI	FP-100B/low/-40°C..85°C	8,10
	VTFI	TFP-100B/low/-40°C..85°C	8,10
	FJ	FP-100B/5V/-40°C..85°C	10,12,16,17
	FQ	FP-100A*/5V/-40°C..85°C	16,17

H8/3048 (ZTAT)

Body	Package suffix	Package/Voltage/Temp.	available clock
HD6473048S	F	FP-100B/5V	16,18
	TF	TFP-100B/5V	16,18
	VF	FP-100B/low	8,13
	VTF	TFP-100B/low	8,13
	FI	FP-100B/5V/-40°C..85°C	16,18
	TFI	TFP-100B/5V/-40°C..85°C	16,18
	VFI	FP-100B/low/-40°C..85°C	8,13
	VTFI	TFP-100B/low/-40°C..85°C	8,13
	FJ	FP-100B/5V/-40°C..85°C	16,18

H8/3048 (F-ZTAT= Flash memory)

Body	Package suffix	Package/Voltage/Temp.	available clock
HD64F3048	F	FP-100B/5V	16
	TF	TFP-100B/5V	16
	VF	FP-100B/low	8
	VTF	TFP-100B/low	8
	FI	FP-100B/5V/-40°C..85°C	16
	TFI	TFP-100B/5V/-40°C..85°C	16
	VFI	FP-100B/low/-40°C..85°C	8
	VTFI	TFP-100B/low/-40°C..85°C	8
	FJ	FP-100B/5V/-40°C..85°C	16

H8/3040 (pseudo ROM-less version)

This device is a mask ROM H8/3040 with a blank ROM, for which standard ROM-less commercial criteria apply (e.g. minimum order quantity). The part names comply with the masked ROM naming rule (please refer to next page).

Body	Vcc/clock suffix	Package suffix	Package/Temperature
HD6433040S	A, P, T, V	00F	FP-100B
	A, P, T, V	00FP	FP-100A
	A, P, T, V	00TF	TFP-100B
	A, P, T, V	00FI	FP-100B/-40°C..85°C
	A, P, T, V	00TFI	TFP-100B/-40°C..85°C
	A, P	00FJ	FP-100B/-40°C..85°C
	A, P	00FQ	FP-100A/-40°C..85°C

Vcc/clock suffix:

A	16MHz/5V
P	17MHz/5V
T	10MHz/3.0..5.5V
V	8MHz/2.7..5.5V

Example: HD6433040SA00FP is a pseudo ROM-less H8/3040 in a FP-100A package, 16MHz, 5V and standard temperature (-20°C..75°C).

* For availability of FP-100A please contact Hitachi or an authorized distributor

H8/3044 (pseudo ROM-less version)

This device is a mask ROM H8/3044 with a blank ROM, for which standard ROM-less commercial criteria apply (e.g. minimum order quantity). The part names comply with the masked ROM naming rule (please see below). It will be available in the first quarter 1997.

Body	Vcc/clock suffix	Package suffix	Package/Temperature
HD6433044s	A, M, S, V	00F	FP-I00B
	A, M, S, V	00TF	TFP-I00B
	A, M, S, V	00FI	FP-100B/-40°C..85°C
	A, M, S, V	00TFI	TFP-100B/-40°C..85°C
	A, M	00FJ	FP-I00B/-40°C..85°C

Vcc/clock suffix:

A	16MHz/5V
M	18MHz/5V
S	13MHz/3.15..5.5V
V	8MHz/2.7..5.5V

Naming rule for H8/300H mask ROM parts

HD643 + device body+ revision + code/speed + ROM code + package

device body	4-digit code of desired H8/300H derivative, e.g. 3042 for H8/3042	
revision	S for mask revision or omitted, for new inquiries always use “S”	
ROM code	2-digit number assigned to mask after customer code submission	
code/speed	A..E	16MHz/5V
	F..H	12MHz/5V
	K,L	10MHZ/5V
	M	18MHz/5V
	P	17MHz/5V
	S	13MHz/3.15..5.5V
	T,U	I0MHz/3.0..5.5V
	V,W	8MHz/2.7..5.5V

When the “ROM code” overflows (>99) the “code/speed” digit advances, e.g. from A99 to B00.

package	F	QFP	14mm x 14mm	
	Fl	QFP	14mm x 14mm	I-spec
	FJ	QFP	14mm x 14mm	J-spec
	FP	QFP	14mm x 20mm	
	FQ	QFP	14mm x 20mm	J-spec
	TF	TQFP	14mm x 14mm	
	TFI	TQFP	14mm x 14mm	I-spec

Example: HD6433032SMxxF is a mask ROM H8/3032S (mask revision), with 18MHz/5V in FP-80A package (14mm x 14mm) in standard spec. xx will be determined after ROM code submission.

Microcontrollers for wide temperature range (WTR)
All of Hitachi's microcontrollers are available in several temperature ranges. The standard temperature range is -20°C..75°C and is not indicated in the package suffix.

-40°C..85°C (I and J-spec)
Hitachi offers two specifications for -40°C..85°C. These are indicated in the package suffix by I and J. I-spec has a wider temperature range compared with standard spec, but has the same reliability. J-spec has improved reliability compared with standard spec. Hitachi recommends to use J-spec for critical applications, particularly in industrial and automotive applications.

-40°C..105°C (JE-spec)
JE-spec has the same reliability as J-spec but extends the operating temperature range to 105°C. This should - for example - be used in critical applications for automotive safety.

-40°C..125°C (K-spec)
K-spec further extends the temperature range to 125°C and also improves reliability over JE-spec. We recommend this for important safety features in the automotive market, where the product is installed e.g. near the engine.

If you require JE- or K-spec, please contact your Hitachi sales office or authorized distributor.