Freescale 68HC08LK ActivePOD rev. B

Ordering code	IC30488

Thank you for purchasing this product from iSYSTEM. This product has been carefully crafted to satisfy your needs. Should any questions arise, do not hesitate to contact your local distributor or iSYSTEM directly. Our technical support personnel will be happy to answer all your technical support questions.

All information, including contact information, is available on our web site www.isystem.com. Feel free also to explore our alternative products.
iSystem constantly yields for development and therefore certain pictures in this documentation may vary slightly from the actual product you received. The differences should be minor, but should you find more serious inconsistencies of the product with the documentation, please contact your local distributor for more information.

Solutions for Embedded Systems Development
www.isystem.com

HC08 Active POD Development System

HC08Active POD consists of a CPU specific Active POD module and either Active POD Emulator Module II HC08-M2 (IC30251) or Active POD Emulator Module II HC08-M4 (IC30252).

Active POD	Required CPU Specific module	Required Active POD Emulator Module II
68HC908AB32 II	IC30474	IC30251
68HC08GP(JL) II	IC30477	IC30251
68HC08GT II	IC30478	IC30251
68HC08AT II	IC30479 / IC30479L	IC30251
68HC08EY	IC30480	IC30252
68HC08LJ	IC30481	IC30252
68HC08JL8	IC30482	IC30252
68HC08AP	IC30483	IC30252
68HC08GR	IC30484	IC30252
68HC08GZ	IC30485	IC30252
68HC08QC	IC30486	IC30252
68HC08LD	IC30487	IC30251
68HC08LK	IC30488	IC30252

HC08Active POD is normally connected to iC3000(HS) or iC4000 through Active Emulator II iCARD (IC30202, IC30203).

The following elements of interest are located on all HC08 Active PODs:

- emulation CPU - acts on behalf of target's CPU. On some PODs you must use the same CPU on the POD as it is used on the target.
- green LED - lights when the Emulator is powered

Assembly Instructions

The following parts are necessary to assemble HC08 Active POD:

- CPU specific HC08 Active POD module (IC304xx)
- Adequate Active POD Emulator Module II (IC30251 or IC30251) depending on the CPU specific HC08 Active POD module
- two tiny screws delivered with Active POD Emulator Module II

1) Connect CPU specific HC08 Active POD module and Active POD Emulator Module II. Mictor connectors used for the connection don't allow incorrect insertion. Make sure that all four Mictor connectors firmly connect together.

2) Screw both modules together with two screws. Be cautious and don't screw to tight!

Trace

HC08 ActivePOD has integrated Trace capabilities that can be utilized if the Active Emulator II iCARD with trace capabilities is used (IC30203).

Final Target Application Test

After the application is being more or less debugged and final application test is performed, it is recommended to remove all breakpoints and to close all debug windows (memory, SFR, watch...) to eliminate any possible influence of the emulator on the CPU execution. There were cases where the target application has been behaving differently with the target CPU inserted or the POD connected. If the debugger is configured to update some debug windows in real-time, the user may not be aware of that the CPU execution may be slightly disturbed. However, when the monitor access type is configured to update debug windows while the CPU is running, the CPU execution is disturbed significantly, depending on the necessary number of memory accesses to update opened debug windows.

There are cases when internal peripheral device requires read access of the particular register during the device configuration. The user has had SFR window opened and the necessary read access was actually performed by the debugger and not by the application as it would be correct. Therefore, the application was working fine with the emulator, but a standalone application didn't work correctly, as the peripheral device was not configured properly.

ActivePOD Emulator Unit II HC08-M4

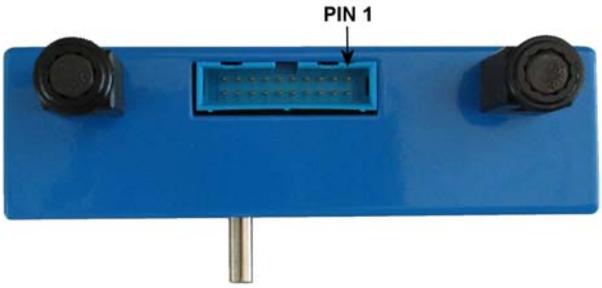
Ordering code	**IC30252**
Module dimensions (mm)	**98x89**

A green LED, present on the Emulator Unit, lights when the Emulator is powered.

The Emulator unit has a Target connector with the following signals:

19	17	15	13	11	9	7	5	3	1
AUX1	AUX3							TRIGOUT	GND
AUX0	AUX2							EXTBP	GND
20	18	16	14	12	10	8	6	4	2

Pin 1 is marked with an arrow on the Target Connector.



- AUX0 – 3 – available AUX inputs (3V LVTTL, connected to FPGA)
- TrigOut – Trigger or Qualifier output depending on the setting in the Trace Configuration dialog, active low when trigger condition occurs (3V LVTTL, connected to FPGA), TrigOut output pulse is approx. 160µs wide.
- EXTBP – External breakpoint input

Jumper Settings

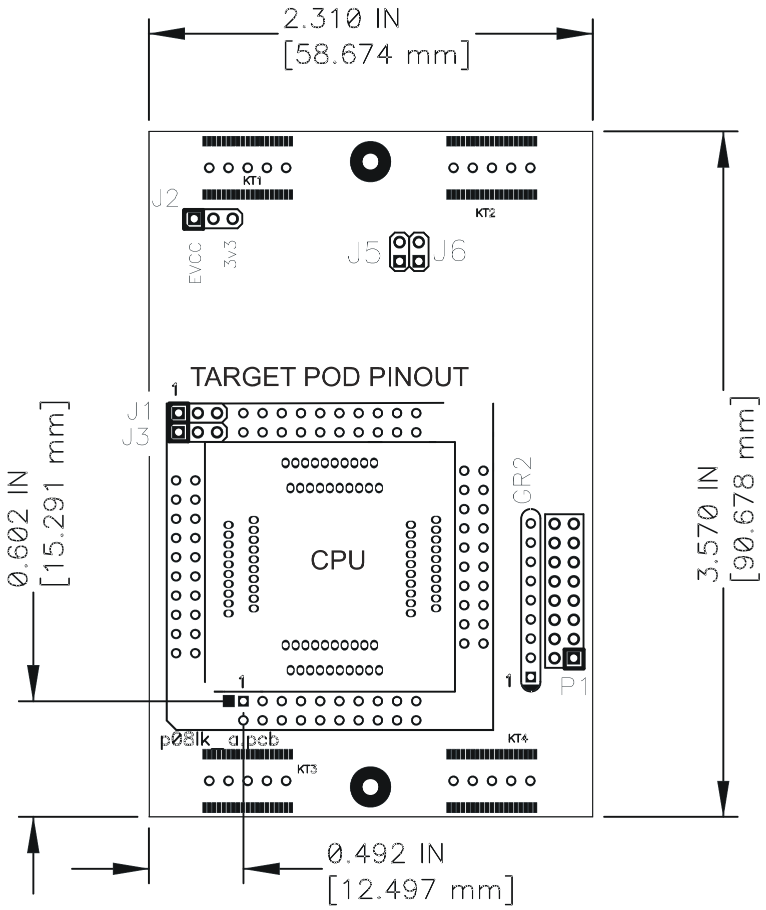
All jumpers on the Emulator Module are for production testing purposes only and should not be moved by the user.

Freescale 68HC08LK ActivePOD rev. B

Ordering code	**IC30488**
Maximum CPU Clock (MHz)	**32**
Trace Depth	**128k**
Exchange CPU	**YES**

Before connecting the POD, make sure you have read the technical notes on Freescale 68HC08 Family in a separate document.

This module connects to the ActivePOD II top module, Ordering Code IC30252.



Bottom board

Emulated CPU
68HC908LJ24
68HC908LK24

CPU Mask Information

A standard 68HC(9)08LK24 CPU is inserted in the POD. If you are using the 68HC(9)08LJ24 CPU, the CPU must be exchanged. If you are confronted with an unexpected application behaviour, which could be due to a different CPU mask being used in the POD and in the target, feel free to exchange the inserted CPU with the one being used in the target. Contact the CPU vendor for more details on the CPU mask differences.

Electrical Differences

In order to enable emulation, 1k Pull-up resistor is present on the Target Reset line.

The original ports A, B and D are used for the emulation and are rebuilt by the port replacement unit on the POD; therefore electrical characteristics are changed.

To enable emulation, the PTC1 line is connected to the Target through an analog switch. This is necesarry to enable entering the emulation mode. After reset the switch is always on.

The IRQ line is also used to start the emulation mode of the CPU. During the execution of the user program a 220E resistor and an 100nF capacitor are present on the line, which causes RC delays on this pin.

The PLL filter, connected to the CGMXFC, is constructed according to Motorola specifications.

The Analog/Digital Converter

The reference voltage for the ADC can be selected using jumpers J1 and J3 on the bottom board.

J1: VREFL reference voltage

The J1 jumper defines the VREFL source.

Position	VREFL connected to
1-2 (*)	GND
2-3	Target VREFL

Jumper J1 settings (- factory default)*

If the Target VREFL is selected, the VREFL from the target must be connected.

J3: VREFH reference voltage

The J3 jumper defines the VREFH source.

Position	VREFH connected to
1-2 (*)	CPU Power Supply
2-3	Target VREFH

Jumper J3 settings (- factory default)*

If the Target VREFH is selected, the VREFH from the target must be connected.

Voltage settings

J2

Jumper selector J2 on the bottom board determines the CPU operational voltage, if the power supply from the Emulator is used.

Position	Vcc level
1-2	3.3 V
2-3 (*)	5.0 V

Jumper J4 settings (- factory default)*

POD Clock limitations and settings

When target clock is used, only an oscillator can be used. A crystal can not be used, since the OSC2 pin is used for emulation purposes and a crystal can not be connected to it.

An 32.768kHz oscillator is built on the POD. If the Target clock is selected in the software, jumpers J5 and J6 select the target clock type selected. **Only one of these jumpers must be inserted.**

Clock selection	J5 Position	J6 Position
Target Clock	Set	Removed
32.768kHz Clock on POD	Removed	Set

Jumper J5 and J6 settings (- factory default)*

Ports

Original ports A, B and D are used for emulation and rebuilt by a port replacement unit. Therefore, electrical characteristics are changed. Because of different technology used in the original chip and the port replacement unit, the POD is not 100% electrically identical to the CPU. These (electrical) differences usually do not matter, until it is relied upon its limits (i.e. max. VIL, min. VIH, max. input current, input resistance, etc.).

Ports A and B also feature alternate functions, i.e. ADC and Keyboard interrupts. The ADC convertor is original, just the ADC selection is done using analog switches.

The Keyboard interrupt logic is emulated on the POD. The POD recognizes the KBSCR and KBIER register configuration and emulates it and generates interrupts to the CPU.

A Port Replacement Unit (PRU) is used to emulate the rebuilt ports. The rebuilt port A has pull-up registers constructed on the POD for the ports that can also be used as keyboard interrupts. The pull-up resistor value is defined by the resistor network, inserted into the GR2 socket. The default value for the resistor network is 22k.

The pull-ups must always be defined by setting the jumpers. P1 sets the pull-ups (P1.0..P1.4 for Port A PTA0..PTA4 ports and P1.4..P1.7 for Port D PTD4..PTD7 ports).

Whenever operating close to electrical limits and having problems with rebuilt ports, please check pull-up and pull-down resistors on the problematical port. They should be neither too strong nor too weak. Check the voltage level. Try to withdraw from voltage limits.

The P1 jumpers configure PORT A pull-up resistors GR2 (22k by default).

Configuration:

Jumper set to	Pull-up enabled on
P10	PA0
P11	PA1
P12	PA2
P13	PA3
P14	PD4
P15	PD5
P16	PD6
P17	PD7

Jumper P1 settings

The P10 is marked with a square on the PCB board.

Electrical differences

In order to enable emulation, 1k Pull-up resistors are present on the Target Reset line and on the IRQ line. Additionally, special logic is added to the IRQ line, required when the CPU is forced to the CTM mode during emulation. The Pull-up defines default inactive state of the IRQ.

General HC08 Emulation Notes

Internal RAM, Internal EEPROM

Note that the internal RAM of the 68HC08 CPU on the POD is disabled during the emulation. Thereby, associated memory area must be mapped as emulator RAM by the user. If the CPU provides a capability to write to the internal RAM or EEPROM via memory window (no specific programming sequence required), the download file can be loaded to the internal RAM or EEPROM using the ‘Target Download’ option. The debugger downloads the code to the internal memory after reset via the CPU. If the CPU (e.g. 68HC08AZ60A) requires some registers to be configured before the CPU is able to write in the EEPROM area, the user must configure the necessary registers respectively, using the initialization dialog. Any sequence, added in the initialization dialog, is executed immediately after reset, before the download is performed. Note that some HC08 derivatives (e.g. 68HC908AB32) don't allow writing to the internal EEPROM area via memory window since a special programming sequence is required. Consequently, the download file cannot be loaded to the internal EEPROM by the debugger. Therefore, the programming algorithm must be implemented by the user in his application. Refer to the CPU datasheets for more details.

Note that debugging is limited while executing the program in the internal EEPROM. At the HC08 family, the internal RAM is disabled during the emulation and the associated memory area overlaid by the in-circuit emulator (ICE). Thus, the debugging in the CPU internal RAM area has no limitations.

While the CPU accesses internal memory resources, the (ICE) loses the control over the CPU since the external bus is not active. Therefore, breakpoints cannot be set and the user's program cannot be stopped or stepped when executing in the internal EEPROM. Additionally, debug windows cannot be updated as well.

Normally, in the target application the CPU executes the program in the internal or external ROM. Using the ICE, ROM memory is overlaid by the emulation memory and consequently the program can be debugged without restrictions. But sometimes there is a need to execute some short routines in the CPU internal memory. Using the ICE, the user can run such a routine, but cannot debug it.

Checksum

When performing any kind of checksum in the emulated (code) area, note that all breakpoints must be removed before, otherwise the results are distorted. Note that the emulator forces "breakpoint" instruction on the data bus when executing the code at the address where breakpoint is set.

COP

Using any HC08 POD, the CPU's internal COP must be either disabled in the CONFIG-1 register (if the CPU has such an option) or serviced by the user's program, otherwise the emulation fails. While the user's program is stopped, the debugger updates the COP counter.

COP servicing by the user's program

Writing any value to location 0xFFFF before overflow occurs clears the COP counter and prevents reset. A user must be careful since reset vector and COP register are located at the same address.

COP update routine should be placed in the main program and not in an interrupt subroutine. Such an interrupt subroutine could keep the COP from generating a reset even while the main program is not working properly.

Clock

Clock source can be either used internal from the emulator or external from the target. It is recommended to use the internal clock when possible. When using the clock from the target, it may happen that the emulator cannot initialize any more.

Internal CPU Flash

Internal FLASH is overlaid by the emulation memory and disabled during the emulation and cannot be used in any way.

Target Adapters

iSYSTEM offers various adapter solutions for this POD. Please refer to the adapter documentation for more details.

POD Target Layout

The POD target layout is T_QFP80.

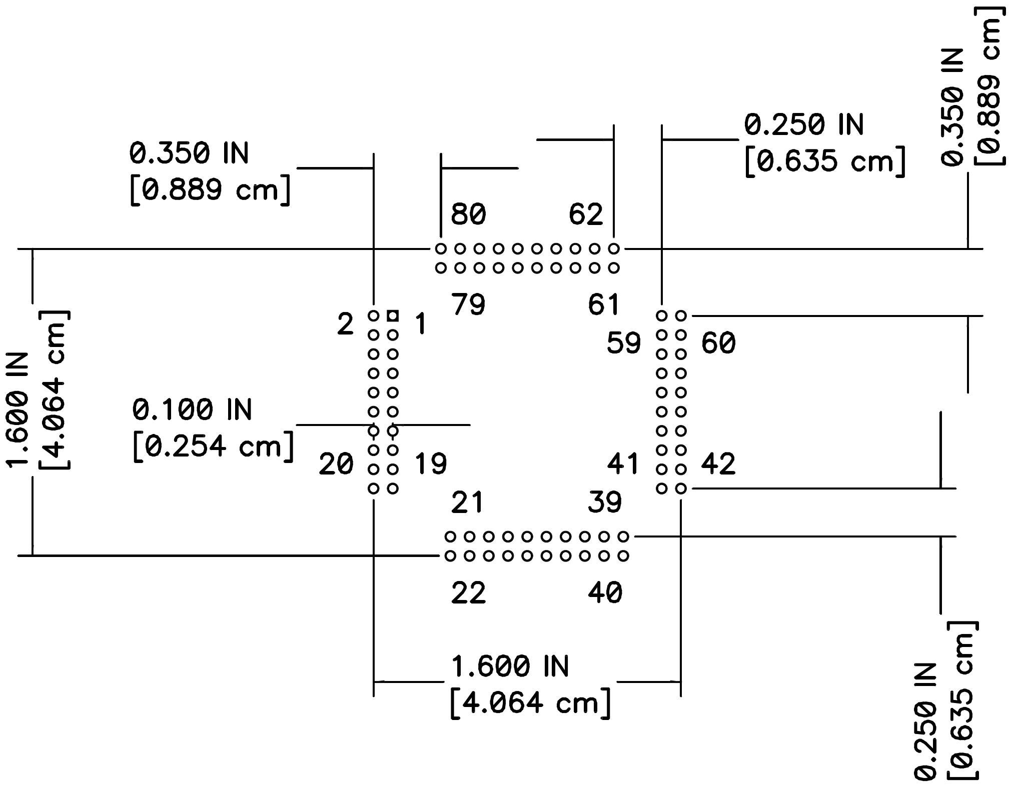
80	78	76	74	72	70	68	66	64	62
79	77	75	73	71	69	67	65	63	61

2	1
4	3
6	5
8	7
10	9
12	11
14	13
16	15
18	17
20	19

59	60
57	58
55	56
53	54
51	52
49	50
47	48
45	46
43	44
41	42

21	23	25	27	29	31	33	35	37	39
22	24	26	28	30	32	34	36	38	40

T_QFP80 – Top POD view



T_QFP80 – Dimensions

Notes: