
POD Hardware Reference

HC912GA32 ActivePOD II Rev. A

Ordering code	IC30402
---------------	---------



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.

This document and all documents accompanying it are copyrighted by iSYSTEM and all rights are reserved. Duplication of these documents is allowed for personal use. For every other case a written consent from iSYSTEM is required.

Copyright © 2001 iSYSTEM, GmbH.
All rights reserved.
All trademarks are property of their respective owners.

POD Hardware Reference

Active POD

Active PODs

Active PODs can be used connected to iC3000 or iC4000 Emulator through the iBUS Active Emulator/Trace interface.

The following elements of interest are located on all 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 (see your POD reference page). In such cases, remove the CPU from the POD and insert the CPU that you use in the target system, in its place.
- red LED - lit when CPU is running
- green LED - lit when Emulator is ready for emulation

For every POD following information is given:

- ordering code;
- information on available speed versions and required Emulator access time;
- size and position of PIN1 on the target adapter relative to bottom left corner.

The memory range specifies the range of addresses that a POD can address. If this specification is omitted the default 1MB is assumed.

Trace

The ActivePOD has integrated Trace capabilities that can be utilised if the interface iCard with trace capabilities is used. See the Trace section in the Hardware User's Guide for general information on Trace and POD information for specific information on trace signals (AUX, TrigOut, etc.).

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.

POD Hardware Reference

ActivePOD Emulator Unit II

Ordering code	IC30250
Board dimensions (mm)	98x82



This Emulator Unit is used on some ActivePODs and is used with an ActiveEmulator II iCard.

The following elements of interest are located on the Emulator Unit:

- Green LED LD1 - Emulation - lit when Emulator is ready for emulation
- Red LED LD2 – Running – lit when CPU is running
- LD3 – CPU Power LED
- LD4 – POD Power LED

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

19	17	15	13	11	9	7	5	3	1
AUX1	AUX3	AUX5	AUX7	AUX9	AUX11	AUX13	AUX15	TRIGOUT	GND
AUX0	AUX2	AUX4	AUX6	AUX8	AUX10	AUX12	AUX14	EXTBP	GND
20	18	16	14	12	10	8	6	4	2

- AUX0 – 15 – available AUX inputs (3V LWTTL, connected to FPGA)

Note: The number of AUX inputs depends of the ActivePOD.

- TrigOut – Trigger output, active low, when trigger condition occurred (3V LWTTL, connected to FPGA)

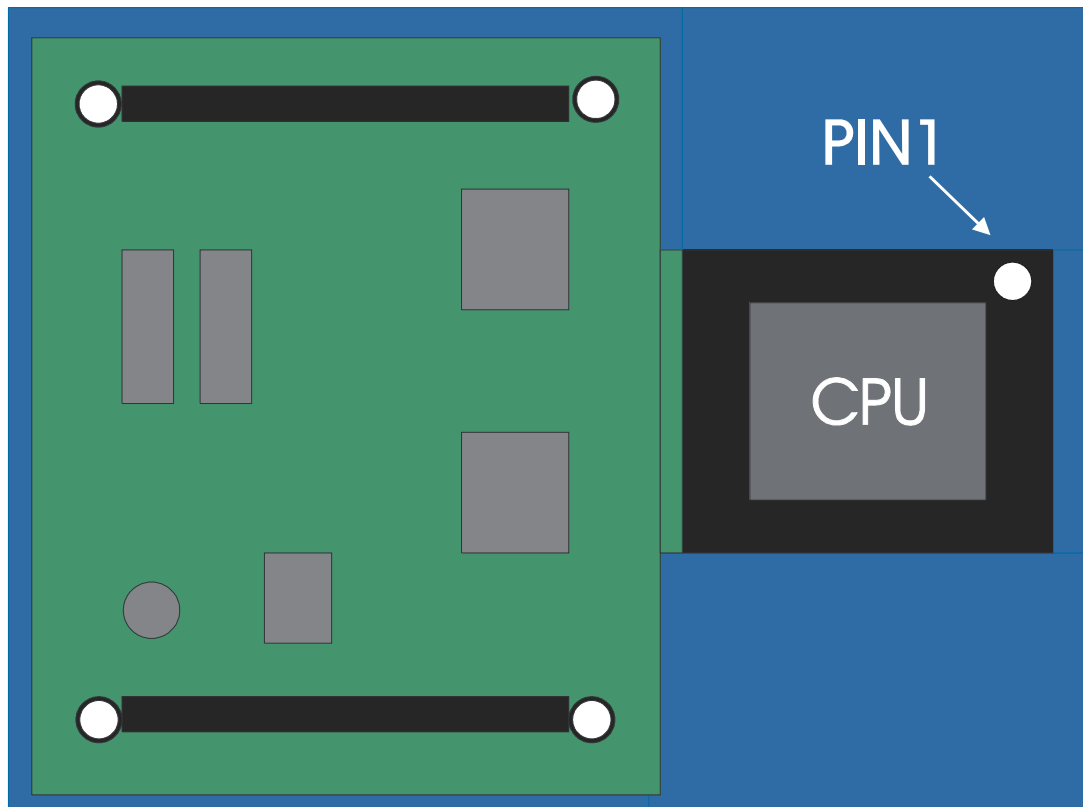
POD Hardware Reference

HC912GA32 ActivePOD II Rev. A

Ordering code	IC30402
POD ECLK Speed (MHz)	8
Exchange CPU	NO
Board dimensions (mm)	99x123
Pin 1 position (mm)	71x118

This POD is connected to the ActivePOD Emulator Unit II.

Before connecting the PODs, make sure you have read the technical notes on Motorola 68HC12 Family in the Hardware User's Guide.



ActivePOD

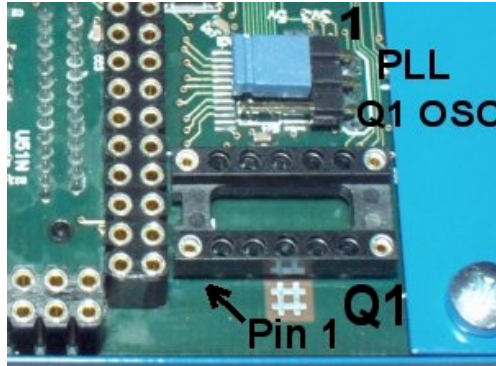
Emulated CPU
68HC912GA32

This POD is connected to the ActivePOD Emulation Unit II and this unit is connected to the Emulator with an iCARD ActiveEmulator II interface. Do not use the ActiveEmulator I iCard.

Clock Setting

Selecting in winIDEA™, the POD can use either internal or external clock from the target.

When internal clock is selected, the user can choose the clock from the emulator (marked as PLL) or a clock generated by the inserted oscillator (Q1) on the bottom side of the POD (see picture below). A jumper J1 selects internal clock source being used. An operating frequency is set in the winIDEA™, when using a clock from the emulator. An oscillator with 5V voltage levels must be used in this circuit.



J1 setting	Internal clock source
1-2 (*)	PLL (emulator)
2-3	Q1 Oscillator

Jumper J1 settings (- factory default)*

When external clock is selected, the POD uses a clock from the target. Motorola has defined very precise PCB and connection guidelines when using a crystal as a clock source. Since these guidelines cannot be considered on the POD completely, it is not recommended to use a crystal in the target as a clock source. The distance between the crystal in the target and the CPU on the POD is critical and thus the crystal may not oscillate at all. Therefore, when external clock is being used, it is recommended to use an oscillator in the target to assure initialization and operation of the system

AUX inputs

This ActivePOD supports 4 AUX inputs. The connector is available on the Emulation unit. Note that on the Emulation Unit II, AUX0 to AUX3 are active only. Any digital signal connected to the AUX input can be recorded by the analyzer, operating in the trace mode. Trigger on it can be set as well.

Synchronization of Two or More Emulators

When multiple emulators should operate synchronously, synchronization connector available on the side of the emulator must be used. All the emulators that should operate synchronously must have connected together (separately) SR (SYNC-RESET) pins, SS (SYNC-STOP) pins and GND (ground) pins.

2	4	6
GND	S-STOP	S-RESET
GND	S-STOP	S-RESET
1	3	5

Synchronization connector

Two lines for each signal are present to allow easier connection of multiple Emulators.

For more information, please refer to the "Synchronization of Two or More Emulators" section of the Hardware User's Guide.

The Port Replacement Unit and special logic on the POD

The CPU on the POD operates in the Expanded Wide Mode. Therefore, certain ports are not available from the CPU and must be reconstructed on the POD with a Port Replacement Unit (PRU). The PRU is logically identical to the ports on the CPU, whilst electrically it is not. The ports A, B, E and K are rebuilt on the POD with the PRU.

The most essential difference between the ports generated with the PRU and the CPU ports are the Input/Output levels. The CPU produces CMOS levels, but the PRU requires TTL levels for input. The output of the PRU still generates CMOS levels.

The support for pull-up resistors is present on the POD and the POD can generate pull-ups as defined in the PUCR (Pull-Up Control Register).

A special logic is present on the POD that checks whether the POD is inserted into the Target or not. If the POD is inserted into the Target, the power supply from the target is required.

For its operation, the POD requires the ECLK, RW, LSTRB, Pipe0 and Pipe1 signals. These settings are defined with PEAR (Port E Assignment Register) and are locked with the software. The only PEAR setting available is the ECLK, which can be defined in the software in the CPU Setup dialog.

General HC12 Emulation Notes

Clock

When using the external clock, the HC(S)12 application doesn't start. It works fine when I use internal clock. There are two major issues the user must pay attention to:

- Also when using external clock, the user must specify target clock frequency ($2 * ECLK$) in the 'Hardware/Emulation Options/Vcc/Clock' tab, like in the case when using internal clock. It is required by the debugger to be able to synchronize with on-chip BDM firmware which operates at CPU's system clock frequency.
- It is not recommended to use a crystal in the target as a clock source during the emulation. It is recommended that the oscillator is used instead. Normally, a crystal and two capacitors are connected to the CPU's clock inputs in the target application as stated in the CPU datasheets. A length of clock paths is critical and must be taken into consideration when designing the target. During the emulation, the distance between the crystal in the target and the CPU (on the POD) is furthermore increased, therefore the impedance may change in a manner that the crystal doesn't oscillate anymore. In such case, a standalone crystal circuit, oscillating already without the CPU must be built or oscillator used.

COP

Internal COP must be disabled when using the emulator, respectively while debugging

STOP Instruction

STOP instruction is completely supported by the emulator. After the STOP instruction is being executed, the CPU is stopped and the debugger displays HALTED status. Note that the debug windows cannot be updated while HALTED status is displayed. When the CPU is awoken either by interrupt or target reset, the emulation/execution proceeds normally.

Internal CPU Flash

Note that internal FLASH is disabled during the emulation and cannot be used in any way.

Internal RAM, Internal EEPROM

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 requires some registers to be configured before the CPU is able to write into 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.

Executing the program in the internal RAM or EEPROM has some limitation, depending on the emulation system.

The run, stop and single step debug commands can be used in the internal RAM or EEPROM. Single step can be performed either in the disassembly or source window. Breakpoints can be set as well.

PLL

The Active PODs support PLL use.

After the PLL is initialized, below equation defines new CPU system clock frequency:

1) after reset PLL is inactive

$Extal = 2 * ECLK$

2) PLL enabled

$ECLK = Extal * (SYNR + 1) / (REFDIV + 1)$

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.

Slow Mode

It is not recommended to debug the application running in slow mode, due to a CPU flaw.

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.

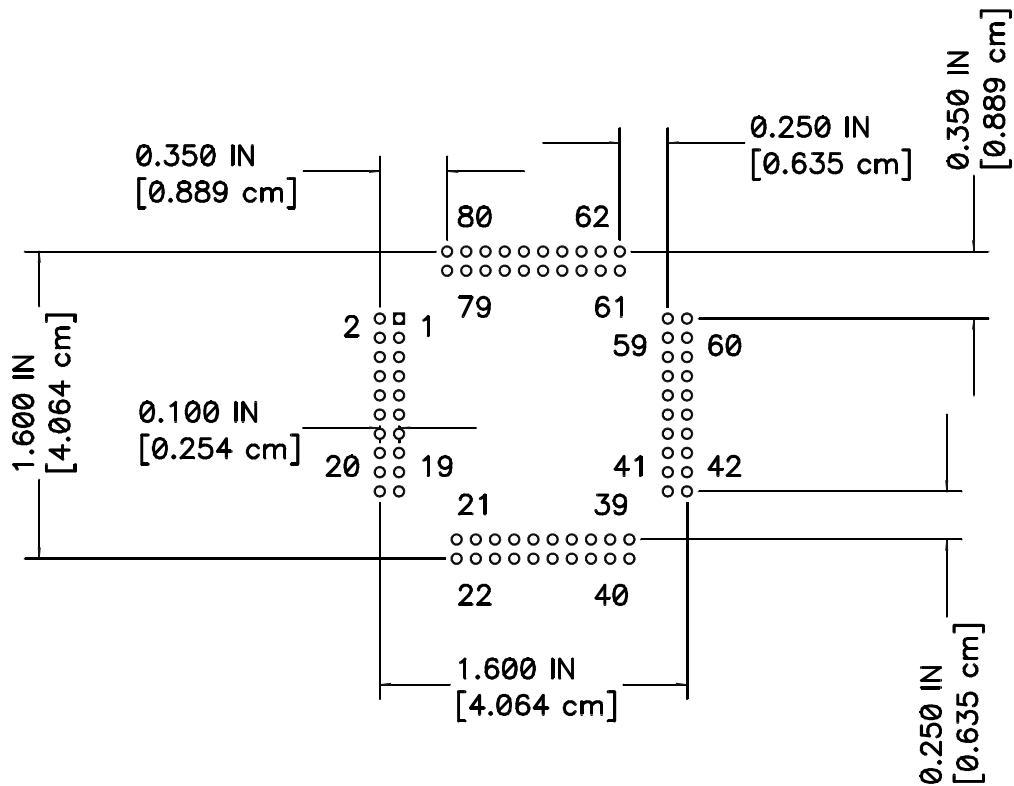
80	78	76	74	72	70	68	66	64	62
79	77	75	73	71	69	67	65	63	61

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