ICE Emulator for MC68000 and MC6830X

TRACE32 Online Help

TRACE32 Directory

TRACE32 Index

TRACE32 Documents ..... 🗁

ICE In-Circuit Emulator ..... 🗁

ICE Target Guides ..... 🗁

ICE Emulator for MC68000 and MC6830X 1

Warning 5

Quick Start 6

Troubleshooting 10

Hang-Up 10

Dualport Errors 11

FAQ 12

Configuration 14

Basics 15

Emulation Method “68302 Dual-Chip” 15

Emulation Method “Single-Chip” 15

Emulation Method “Hybrid-Solution” 16

Target Power Sense 16

Emulation Modes 17

SYStem.Clock Clock generation 18

Dualport Modes 19

DMA Modes 20

General SYStem Settings and Restrictions 21

General Restrictions 21

SYStem.Line BusSize Bus width 21

SYStem.Line BusReq Bus request when stopped 21

SYStem.Line WDOG WDOG line 22

SYStem.Option MOBAR Module controller base address register 22

SYStem.Option BAR Base address register 22

SYStem.Option BrkVector Break vector 22

SYStem.Option BreakWin Break window 23

SYStem.Option FAST High-speed emulation 23

SYStem.Option FCode Ignore function codes 23

SYStem.Option Freeze FREEZE control 24

Automatic Freeze 24
External FREEZE 24
Freeze by Analyzer 24
SYStem.Option TestClock Clockfail detection 25
SYStem.Option TrReset Tristate bus 25
SYStem.Option TrHalt Tristate bus 25
SYStem.Option V33 Powerfail detection 25
SYStem.Option FreezeExtern Target freeze enable 26
SYStem.Option TraceBank Bank trace enable 26

Special Settings 68302 Dual -chip and Restrictions 27
Restrictions 68302 Dual-chip 27
SYStem.RESetOut Reset target CPU 28
SYStem.Option BR0/OR0 CS0 line replacement 28
SYStem.Option BAR/BARI BAR registers 29
SYStem.Option IntChange Interrupt translation 31
Clock Generation 32
ONCE Mode Handling 33
Disable Target CPU 33
Use Target CPU 33

Special Settings 68302 Single Chip and Restrictions 36
Restrictions 68302 Single Chip 36
SYStem.Option MonBase Monitor relocation 36
SYStem.Option Dedicated Interrupt controller mode 37
ONCE Mode Handling 37

Special Settings 68356 38
SYStem.Line DRESET Reset configuration 38
SYStem.Line MODCK Clock mode selection 38
SYStem.Option MODCKx Clock mode selection 38
SYStem.Option CS Clock mode selection 39
SYStem.Option VCO/100 32KHz support 39

Special Settings 68PM302, 68LC302 40
SYStem.Option BRx/NOCSBASE A20..23 address creation 40
SYStem.Option DTACK DTACK configuration 40
SYStem.Option MODCLK Clock mode selection 41
SYStem.Option VCCSYN Clock mode selection 41
SYStem.Option ONCE ON-Circuit emulation (68PM302 only) 41
SYStem.Option PCEN PCMCIA (68PM302 only) 41
SYStem.Option ResetExt Reset configuration 42
SYStem.Option VCO/100 32KHz support 42

Special Settings 68EN302 43
ONCE Mode Handling 43
SYStem.Option MBCTL DRAM control 43

SYStem.Option BUSW Bus width selection 43

Special Settings GSC38393 and Restrictions 44
Restrictions GSC38939 44
SYStem.Option WSEL Reset configuration 44
SYStem.Option BUSW0 Reset configuration 44
SYStem.Option CKSL Clock selection 44

Special Settings 68328 and Restrictions 45
Restrictions 68328 45
SYStem.Option BBUSW Boot bus width 46
SYStem.Option MOCLK Clock selection 46
SYStem.Option PORTB Port B enable 46
SYStem.Option PRBASE Port replacement base address 47

Special Settings 68306 and Restrictions 48
Restrictions 68306 48
SYStem.Option AMODE Address mode 48
SYStem.Option ONCE On-circuit emulation 48
SYStem.Option XTAL Oscillator replacement 48
68306 Clock Generation 49

Special Settings 68307 and Restrictions 50
Restrictions 68307 50
SYStem.Option DISWT Watchdog disable 50
SYStem.Option DRAM DRAM mode 50
68307 Clock Generation 51

Exception Control 52
RESET Line 53
HALT Line 55
BERR Line 56
BUSREQ Line 57
VPA Line 57
Interrupt Control 58
Interrupt Stimulation 59
Trap Trigger 59

Monitor Extensions 60

Mapping 61
MAP.BUS8 Bus width mapping 61
MAP.OPFetch Code area mapping 61

Memory Classes 62

State Analyzer 63
Keywords for the Trigger Unit 64
Keywords for the Display 66

Dequeueing 66

Port Analyzer 67

Default Keywords for the Port Analyzer 67

68306 Keywords for the Port Analyzer 69

68307 Keywords for the Port Analyzer 70

Default Connector for Additional Trace Channels 72

Single Chip 68302 Connector for Additional Trace Channels 73

Compilers 74

3rd-Party Tool Integrations 76

Realtime Operation Systems 78

Emulation Frequency 79

Emulation Module 80

Module Overview 80

Order Information 81

Operation Voltage 83

Physical Dimensions 84

Adaption 111

ICE Emulator for MC68000 and MC6830X

Version 18-Nov-2020

For general informations about the In-Circuit Debugger refer to the **"ICE User's Guide"** (ice_user.pdf). All general commands are described in **"PowerView Command Reference"** (ide_ref.pdf) and **"General Commands and Functions"**.

Warning

NOTE:	Do not connect or remove probe from target while target power is ON. Power up: Switch on emulator first, then target Power down: Switch off target first, then emulator

Quick Start

Before debugging can be started, the emulator must be configured by hardware and software:

1. Check DIP-switch setting (chapter **Configuration**)
2. Create setup file (next)

Ready to run setup files for most standard compilers can be found on the software CD in the directory **…/Demo/M68K/Compiler**. All setup files are designed to run the emulator stand alone without target hardware.

The following description should make the initial setup (to run the emulator together with the target hardware) easier. It describes a typical setup with frequently used settings. It is recommended to use the programming language PRACTICE to create a batch file, which includes all necessary setup commands. PRACTICE files (*.cmm) can be created with the PRACTICE editor pedit (Command: **PEDIT <file name>**) or with any other text editor.

A basic setup file includes the following parts:

1. Set system options
2. Select dualport mode (optional)
3. Set mapper (optional)
4. Select frequency (optional)
5. Activate the emulator
6. Load application file (optional)
7. Initialize registers and chipselect units (optional)
8. Set breakpoints (optional)
9. Start application
10. Stop application (optional)

Now a typical example, how to setup the system:

1. Set **system** options

   The system window controls the CPU specific setup. Please check this window very carefully and set the appropriate options. Use the ↖? button in the main tool bar and click to the option check box (Command: **HELP.PICK**) to get online help in a pop up window.

```
system.down                  ; switch the system down
system.reset                 ; all system settings to default
system.option v33 on         ; on: if 3.3 V module is used
system.line bussize 16       ; select 16 bit mode
```

2. Select **dualport modes** (optional)

   Dualport allows access to emulation RAM, while emulation is running. This is necessary to display variables, set breakpoints or display the flag listings while the emulation is running. **System.access** selects how dualport access is done.

```
system.access request        ; request: BR-/BG- line is used for
                             ; dualport
```

3. Set mapper (optional)

   The mapper controls the memory access of the CPU. This means the use of internal or external memory, the number of wait states, the bus width etc.

```
map.reset                          ; reset mapper (all external)
map.mode fast                      ; use fast mode
map.ram 0x000000--0x0fffff         ; emulation RAM: use low 1MB
map.ram 0x300000--0x3fffff         ; emulation RAM: use top 1MB

map.intern 0x0000000--0x00fffff    ; memory: use low 1MB internal
                                   ;         use top 1MB external
                                   ;        use top 1MB for dualport
```

   Select **frequency** (optional)

   The CPU can be clocked by internal (emulator) or external (target). If the internal clock is used, the clock is provides by the VCO of the emulator. The setting of the internal clock is done by the **VCO** command.

   The current CPU frequency can be displayed in the counter window (Command: **Count**).

```
vco.clock 16.7               ; frequency: set to 16.7 MHz
                             ; (necessary if internal clock used)
```

4. Activate the emulator

When the emulator is activated a monitor program is loaded into hidden emulator memory. After the load and the falling edge of RESET the monitor program is started. This program allows access to user memory (data.dump, data.list) and register and gives control to start and stop the emulation.

```
system.mode emulext          ; system up: emulation external
                             ; (target, ext. clock)
                             ; or: system.mode aloneint
                             ; (stand alone, int. clock)
```

5. **Load** application file (optional)

Application can be loaded by various file formats. For information about the load command for your compiler see **Compiler.**

```
data.load.ieee file.abs /nocode    ; load application file (symbols
                                   ; only)
```

6. Initialize **registers** and chipselect units (optional)

For correct data.list and data.dump after RESET it could be necessary to initialize chipselect units. This can be done in the **PERipheral window** or by data.set commands to the chipselect registers. Stackpointer and program counter should be initialized by hand if debugging is started at RESET until it is initialized by the program. Stack is used for the emulator break system.

```
register.set pc  0x400       ; initialize program counter
register.set ssp 0x0fff0     ; initialize stack pointer to allow
                             ; debugging from begin of program
```

7. Set **breakpoints** (optional)

There are several ways to set breakpoints (Command: **Break.Set**). Breakpoints can be displayed using the **Break.List** command. Information regarding HLL lines (for HLL breakpoints) is loaded automatically when a HLL file is loaded.

```
breakpoint.set main /program        ; set program break on function
                                    ; main
breakpoint.set counter /write       ; set write break on variable
                                    ; counter
```

8. **Start** application

   Application can be started with giving a break address. For example **"go main"** starts the application and stops at symbol main.

```
go                          ; run application
```

9. **Stop** application (optional)

   Application can be breaked manually by using the **Break** command. If application executed a halt instruction the command **Break.HALT** should be used to terminate the application.

```
break                       ; break application by hand
```

It is recommended to check the following chapters for all questions regarding the correct setup:

- **Configuration**
- **General SYStem Settings and Restrictions**
- **Special Settings 68302 dual chip and Restrictions**
- **Special Settings 68302 single chip and Restrictions**
- **Special Settings 68356**
- **Special Settings 68PM302/68LC302**
- **Special Settings 68EN302**
- **Special Settings GSC38393 and Restrictions**
- **Special Settings 68328 and Restrictions**
- **Special Settings 68306 and Restrictions**
- **Special Settings 68307 and Restrictions**
- **Troubleshooting**

Hang-Up

If you are not able to stop the emulation, there may be some typically reasons:

Double Address Error

After a double address error the CPU is in halt state, use the **SYStem.Up** command to start again. Double address errors normally occur when the stack pointer is out of memory.

No DTACK Signal

If not **TIMOUT** is specified, the CPU cycle is not completed if the **DTACK** signal fails. On memory display windows **BERR** signals are not accepted. You can verify this state by checking the **CYCLE** signal with the counter function. When low, the CPU is stopped in the middle of the cycle. When request mode is selected, a dualport error occurs and the emulator system changes to reset state.

Clock Error

The clock lines between the target and the oscillator replacement are very short. Therefore normally no problems should occur when using an external crystal. Be sure that the capacitors on the target have a value of 20 pF minimum and are with short routes connected to the CPU socket.

Interrupt Request

If all IPL signals are active low at the same time (NMI request) you can not use an asynchronous break. This interrupt level in usually used for fatal errors in target systems only. If only program breakpoints are used, no restriction in using interrupt level 7 is known.

Analyzer Function

If you switch off the analyzer and the CPU has stopped operation, an invalid display occurs. Make a **SYStem.Up** command to see the true trace information.

Dualport Errors

To realize the dualport access (emulation memory) the BR-line of the CPU is used. Dualport accesses are allowed only while no external request to the bus occurs and the CPU cycle is completed. If the emulation CPU is in RESET state of the CPU the system controller may always access the emulation memory.

Dual-port errors may occur by the following conditions:

10. The length of the CPU cycle is extended by wait cycles, so that the request timeout signal is generated.
11. External DMA requests (single cycles) are too long.
12. The bus arbitration lines are not usable (e.g. 68307). See 'Restrictions' part for your processor.

To solve problems with dualport error first increase the **SYStem.TimeReq** value. Be sure that the **SYStem.TimeOut** value is bigger than the access time limit. If it is not possible to solve the problem by changing the values, you must switch to **DENIED** mode. In this mode no access to memory is possible while running realtime emulation. The internal dualport access can increase the reaction time for external DMA requests. The performance reduction by the dualport access is typically 1% with some data windows (dualported) on the screen and may be at max. 5% when using dynamic emulation memory.

Target Power Supply Switch
 	
Ref: 0103
	Is there a simple way to control target power supply via the ICE to prevent problems after the ICE has been powered off?
	
	Follow the sequence below.
	
	If you own an output probe COUT8, connect it to the STROBE output connector.
	
	Type PULSE2. and press F1. You will get the pin out of the output probe COUT8. Pin 13 (OUT6) delivers +5 V after the emulator has finished its initialization and 0 V if the emulator is powered off. This can be used to drive a relay via a transistor to switch the target power on and off automatically if the Pulse Generator is not used for other purposes. The schematic of the switching unit can be found in the file TARGETC.CMM.
	
	Additionally Pin 13 (OUT6) can be controlled by ICE commands.
	
	`Target power supply off. "PULSE2.P +"`
	`Target power supply on.  "PULSE2.P -"`
	
	The following PRACTICE command file creates 3 buttons in the Toolbox for:
	
	`Target power on`
	`Target power off`
	`Target power off and QUIT.`

Wrong Location after Break Ref: 0030	**Why is the location after break wrong?** Most emulators use some bytes of user stack for the break system. Therefore it is necessary to have valid stack, if single step or breakpoints are used.
68302D Wrong Exception Vectors Ref: 0033	**What could be the reason for wrong exception vectors?** The 68302 dual chip module uses two 68302 CPUs. Exception vectors are generated by the slave CPU which is in "disable CPU" mode. After reset no vectors will be generated because the VGE (Vector Generate Enable) of the System Configuration Register is disabled. This bit must be set by the user-program or by a data.set emulator command before the first interrupt is pending.

68LC/PM302
 	
Wrong Addresses in PC and Analyzer Listing
 	
Ref: 0032
	What could be the reason for wrong addresses in the program counter or analyzer listing?
	
	The 68LC302 and 68PM302 (if PCMCIA enabled!) only support addresslines A00--19. The emulator creates the missing addresslines A20..23 by using the chipselect configuration which must be entered to the SYStem window. If the emulator configuration does not match the CPU chipselect configuration, a wrong address will be created which will cause a wrong program counter address, a wrong analyzer display or trouble with mapping and breaking.
	To prevent such trouble the emulator and CPU configuration should be done before running program and should not be changed during program execution. If it is necessary to change the chipselect configuration, the program must be stopped to change the configurations by emulator commands and the program part for chipselect configuration must be skipped.
	Example for a PRACTICE command file:
	
	`; configure emulator CS0..3 baseaddresses`
	`SYStem.Option BR0 value1`
	`SYStem.Option BR1 value2`
	`SYStem.Option BR2 value3`
	`SYStem.Option BR3 value4`
	`; configure CPU CS0..3 baseaddresses`
	`Data.Set 0fff830 %Word value1`
	`Data.Set 0fff834 %Word value2`
	`Data.Set 0fff838 %Word value3`
	`Data.Set 0fff83c %Word value4`
	`end`

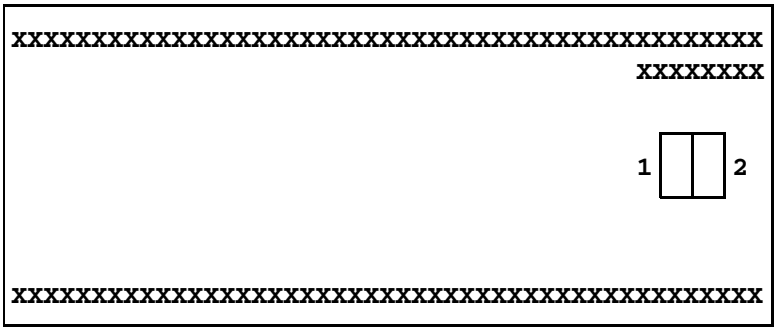
68XXX Watchdog/Timer Ref: 0036	**In which way timers and watchdog can be handled when emulation is stopped?** By default the timer- and watchdog counters are still enabled if emulation is stopped. To prevent watchdog resets or timer exceptions there are three ways to handle them. 1. Disable them for ever: Please have a look to the CPU Users Manual how to disable the watchdog and timers. 2. Watchdog and timers are enabled but stopped as long as emulation is stopped: There are some CPUs in the 68K family which have a FREEZE pin. The emulator handles this line, which can be used to stop timers. Stopping of timers, watchdog, SDMA or IDMA only works if this CPU option is enabled. The "System Control Register" of the 68302 contains the freeze bits for timers and watchdog. Please have a look to the CPU Users Manual for additional FRZ bits and have a look to the on line help "SYStem.Option FREEZE" command. 3. Exceptions are handled in a Background Task: Please have a look to the online help: "go.back" command, "break.enable back" command.

Configuration

The configuration of the different target CPUs and sockets is done by changing the probe or the sockets. The port analyzer is optional. The software is configured automatically.

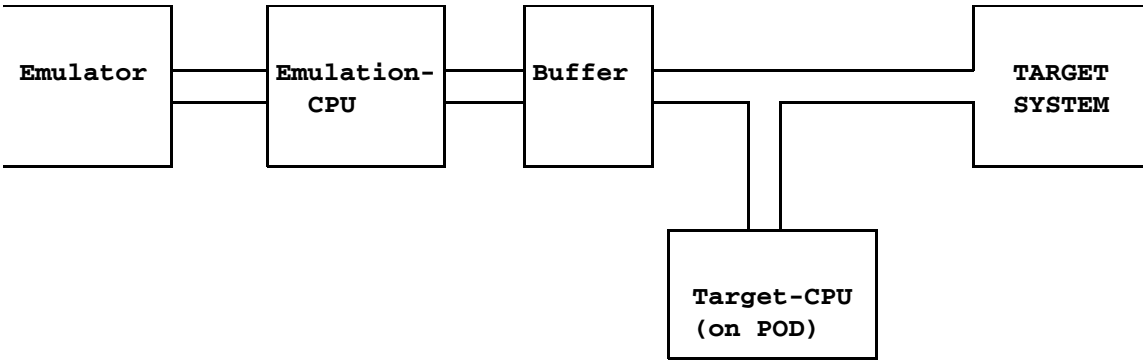
Emulation	CPU	DIP-SW 54321	X1 X2	Adapter	
68008 DIL48	68001	10000	-	-	DIL48
68008 PLCC52	68001	10001	-	-	PLCC52
68000 DIL64	68000/1	11111	-	-	DIL64
68000 PLCC68	68000/1	11111	0	0	PLCC68
68000 PGA68	68000/1	11111	0	0	PGA68
68001 PLCC68	68001	11111	1	1	PLCC68
68001 PGA68	68001	11111	1	1	PGA68
68010 DIL64	68010	01111	-	-	DIL64
68010 PLCC68	68010	01111	-	-	PLCC68
68301 QFP	68001	11111	-	-	68301
68303 QFP	68001	11111	-	-	68303
68307 QFP	68001	11111	-	-	68307

Adapter PLCC68/PGA68



Emulation Method “68302 Dual-Chip”

The 68302 dual-chip probe uses two 68302 CPUs to realize the in-circuit emulator. One CPU (called emulation CPU) is used as a CPU core, with deactivated internal peripherals (mapped to an unused memory area). The second CPU (called target CPU) is used as a peripheral device, with deactivated CPU-core. This dual CPU solution has many advantages, but also some disadvantages against a single CPU version.



Emulation Method “Single-Chip”

Single-chip probes uses one CPU to realize the in-circuit emulator. Internal DMAs of the CPU’s peripheral must be stopped by the FREEZE function while the emulation is stopped.

Emulation Method “Hybrid-Solution”

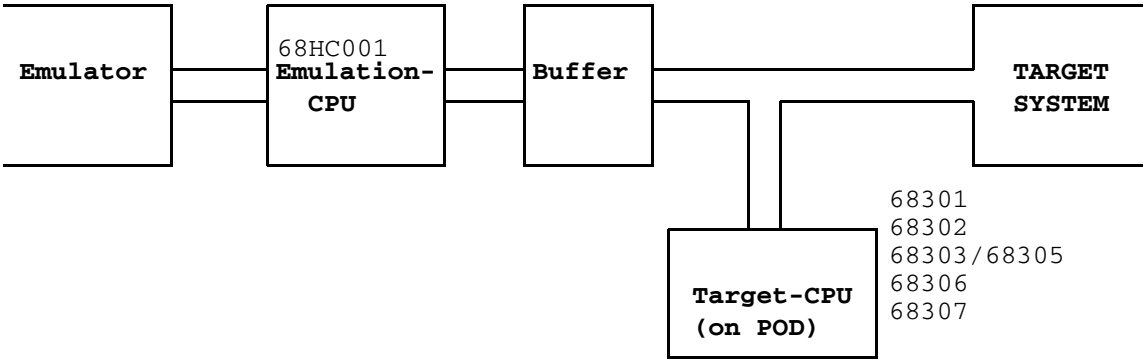
The 68001 probe uses a 68HC001 to realize the in-circuit-emulation for 68000, 68001 or 68008. The probe is also able to run with 68010.

With additional adapters the emulation of TMP68301, TMP68303, MC68302, MC68306, MC68307 or other Flex-Core products is possible. The 68001 CPU is used as emulation CPU, while the CPU core of target processor is switched off.

The 68306 emulation module has two 68306 processors, one is used as CPU code, the other as a peripheral chip.

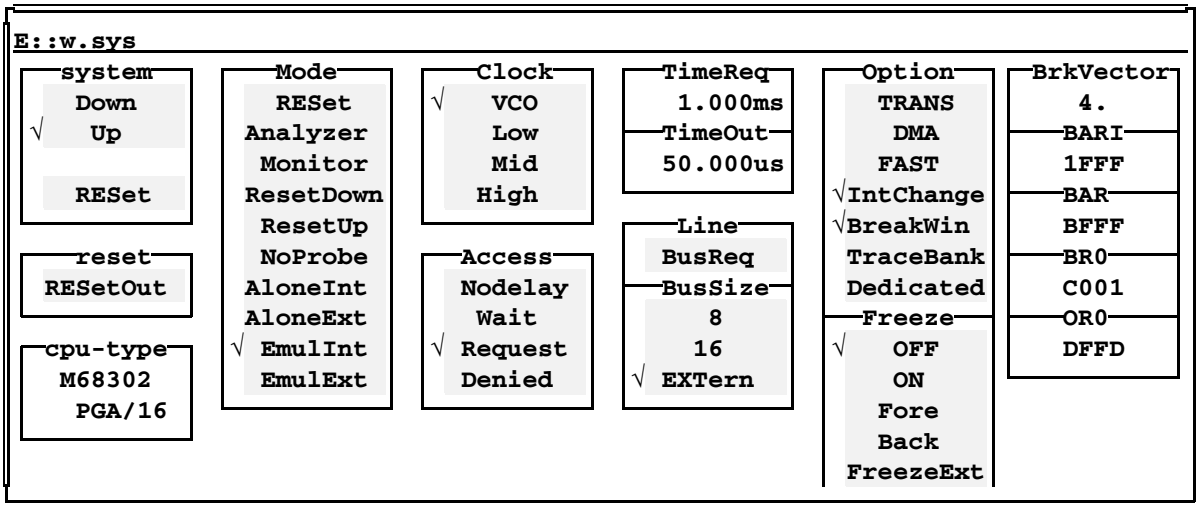
The MC68307 probe uses a Bond-Out chip to get access to the internal peripherals and the 8051 bus

The hybrid solution for the MC68302 combines the advantages and disadvantages of single-chip and dual-chip methods. The function of the RMC, CS0 and VPA lines cannot be used with this solution.



Target Power Sense

In active mode the target power is sensed by the emulator. By switching down the target power the emulator changes to **RESET** mode. The probe is not supplied by the target. When running without target, the target voltage is simulated by an internal pull-up resistor. For 3.3Volt targets the power fail detection must be adjusted with emulator command **SYStem.Option V33 ON**.



The emulation head can stay in 6 modes. The modes are selected by the **SYStem.Up** or the **SYStem.Mode** command.

Format: **SYStem.Mode** <*mode*>

<*mode*>: **ResetDown**
ResetUp
AloneInt
AloneExt
EmulInt
EmulExt

Reset Down Target is down, all drivers a in tristate mode.

Reset Up Target has power, drivers are logically in inactive state, but not tristate.

Alone Internal Probe is running with internal clock, driver inactive, the dual-chip target CPU is not activated.

Alone External Probe is running with external clock, driver inactive, the dual.chip target CPU is not activated.

Emulation Internal Probe is running with internal clock, strobes to target are generated.

Emulation External Probe is running with external clock, strobes to target are activated.

Format:	**SYStem.Clock** <*option*>
<*option*>:	**VCO** **High** **Mid** **Low**

VCO Variable frequency 1 … 35 MHz.

Low, Mid, High 2.5, 5.0 or 10.0 MHz.

Dualport Modes

Dualport allows access to emulation RAM, while emulation is running. This is necessary to display variables, set breakpoints or display the flag listings while the emulation is running. **System.Access** selects how dualport access is done.

Format:	**SYStem.Access** <*option*>
<*option*>:	**Nodelay** **Request** **Denied** **Idle** **Halt**

Nodelay Dualport access is done while the bus cycle is not active (AS- is HIGH). This access mode was made for CPU's which have no BR- line (68LC302, 68PM302). With 25 ns SRAM this mode works up to 19 … 20MHz.

Request Dualport access is done by forcing the BR- line to get access to the bus.

Denied There is no dualport access while emulation is running.

Idle Dualport access is done while the CPU is running idlecycles (AS- is HIGH for more than two clock cycles). This access mode was made for CPUs which have no BR- line (68LC302, 68PM302). With 25 ns SRAM this mode works up to 21 … 22 MHz. There will be dualport errors if the executed program has no idlecycles or if there are not many of them. With an increased **SYS.TimeReq** value the problem might be solved.

Halt Dualport access is done by forcing the HALT line to get access to the bus. This access mode was made for CPU's which have no BR- line (68LC302, 68PM302). It works up to the maximum CPU clock speed.

External DMA circuits and the IDMA circuit on the target CPU work in the same way: both request the main CPU with the BR signal. In realtime emulation the emulation CPU is stopped and the DMA can get control of the bus. When emulation is stopped, no BG signal is generated and the DMA is waiting till realtime emulation is started.

Format:	**SYStem.Line BusReq [OFF | ON]**
Format:	**SYStem.Option DMA [OFF | ON]**
Format:	**SYStem.Option TRANS [OFF | ON]**

SYStem.Line BusReq This option allows DMA access without running realtime emulation. External DMA circuits are not stopped on breakpoints.

SYStem.Option DMA DMA cycles may be traced and trigger system is also active on dma cycles.

SYStem.Option TRANS The DMA accesses make also writes to emulation memory. On read access to internal mapped memory the data information is driven to the target system. Be sure that there is no external memory to avoid bus conflicts.

General Restrictions

Memory Set-up	All 68000-type in-circuit-emulators need memory in the supervisor stack area (SSP) to break correctly. If you get an invalid PC value after stopping the program, the SSP register may be outside the memory area. This must be considered especially when debugging the start-up code of an application.
Register Set-up	The SR register trace flag must not be set to 1.
Emulation Memory with DMA	Emulation memory is accessible only in Supervisor and User Data and Program area. The DMA must use the function codes FC1, FC2, FC5 or FC6 to access emulation memory correctly. New ICE boards have the option to ignore function codes (see SYStem.Option Fcode).
disable CLKO	It is not allowed to disable the CLKO signal!

SYStem.Line BusSize — Bus width

Format: **SYStem.Line BusSize** <*option*>

option:
8
16
EXTern

Because the emulator control program must know the bus width, the BUSW signal is not checked when restarting from reset. The user must define the bus width before starting emulation. In external mode the input line BUSW is tested to define the internal state.

SYStem.Line BusReq — Bus request when stopped

Format: **SYStem.Line BusReq [OFF | ON]**

Enables the BR line when emulation is stopped.

Format: **SYStem.Option WDOG [OFF | ON]**

Disconnect WDOG pin from target. Disconnected if ON

SYStem.Option MOBAR Module controller base address register

Format: **SYStem.Option MOBAR** [<*value*>]

Defines the base address of the registers in the 4 kbyte Module Controller Block which are in addition to the register set of the CPU.

SYStem.Option BAR Base address register

Format: **SYStem.Option BAR** [<*value*>]

Defines the base address of the CPU internal peripherals. This base address information is used by the PERipheral window.

If the base address has changed the PERipheral window must be reprogrammed. Use command:**PER.REPROGRAM.**

SYStem.Option BrkVector Break vector

Format: **SYStem.Option BrkVector** <*vector*>

<*vector*>: **0 … 255.**

On asynchronous break a vector is generated internally. The setting of this vector defines which entry of the vector table is read (trap trigger) and which read flags (flag memory) are set. The vector is normally set to 4 (undef trap).

Format: **SYStem.Option BreakWin [OFF | ON]**

All program breakpoints are hardware-based. The operation is done by replacing the opcode by an UNDEF command. For not breaking on every **UNDEF** codes in the target program break sequencing is enabled only some cycles after the breakpoint cycle. In some cases it may be an advantage to switch off this feature and stop the emulation at each undefined opcode.

SYStem.Option FAST High-speed emulation

Format: **SYStem.Option FAST [OFF | ON]**

When running at frequencies above 20 MHz this switch must be set to ON. There is no effect on the target program, only the emulation monitor is running with waitstates.

SYStem.Option FCode Ignore function codes

Format: **SYStem.Option FCode [OFF | ON]**

By default the emulation memory is accessible for function code FC1, FC2, FC5 and FC6. Sometimes there might be problems to configure the DMA controller to generate DMA cycles with one of this function codes, so the DMA cycle has no access to emulation memory. With the following option the function codes can be ignored by the emulator.

ON: Function codes will be compared

OFF: Function codes will be ignored. FC0..6 cycles have access to the emulation memory.

If function codes are ignored and **MAP.SPLIT** is active DMA cycles with

- FC0 cycles will access FC2(UserProg) emulation memory
- FC3 cycles will access FC1(UserData) emulation memory
- FC4 cycles will access FC6(SupProg) emulation memory

Defines Freeze Line handling.

Automatic Freeze

Format: **SYStem.Option Freeze** <*option*>

<*option*>: **OFF**
Fore
Back
ON

As the dual-chip probe works with two CPUs, the peripherals must not be stopped when stopping the CPU core. The peripherals may be active only on program running, always or never.

OFF The peripherals of the target CPU are never stopped by the emulator. This option is available with the dual-chip probe only.

Fore The peripherals are frozen if the emulation is stopped, or if the foreground program is running.

Back The peripherals are frozen if the emulation is stopped, or if the background program is running.

ON The peripherals are always stopped.

External FREEZE

The line FRZ- on the emulation pod is ored together with the internal freeze control logic. An additional freeze signal may be added by the bank input probe. If the command **SYStem.Option FreezeExt** is switched ON, PIN 7 of the BANK-probe is used as an additional freeze signal (active low).

Format: **SYStem.Option FreezeExt [OFF | ON]**

Freeze by Analyzer

The function **AUX.A** can be used to generate a freeze signal.

Format: **SYStem.Option TestClock [OFF | ON]**

Switch for clock fail detection.

SYStem.Option TrReset Tristate bus

Format: **SYStem.Option TrReset [OFF | ON]**

Many CPUs tristate their bus if there is a RESET or HALT asserted.

TrReset Trestates the bus if a system reset (RESET and HALT are Low) is asserted.

SYStem.Option TrHalt Tristate bus

Format: **SYStem.Option TrHalt [OFF | ON]**

Many CPUs tristate their bus if there is a RESET or HALT asserted.

TrHalt Trestates the bus if HALT is asserted.

SYStem.Option V33 Powerfail detection

Format: **SYStem.Option V33 [OFF | ON]**

The emulator has logic to detect a power fail. This logic has to be adjusted for 3.3 V CPUs.

Format: **SYStem.Option FreezeExtern [OFF | ON]**

ON enable target freeze input

OFF disable target freeze input

Enable target freeze input.

SYStem.Option TraceBank Bank trace enable

Format: **SYStem.Option TraceBank [OFF | ON]**

ON enable tracing of bank signals

OFF disable tracing of bank signals

Enable tracing of bank signals.

Restrictions 68302 Dual-chip

Emulation Mode	It is not possible to access the CPU peripherals if SYStem.Mode AloneInt or AloneExt is used. Please use SYStem.Mode EmulInt or EmulExt.
DTACK on BAR address	The Target DTACK line should not be connected to GND as this will also cause a DTACK when the BAR register of the target CPU should be accessed. The **MAP.Ack** command should be not used on the address of the BAR register. No emulation memory should be mapped internal at the address of the SCR register.
Write to BAR Register	Writing the **BAR** register is not allowed. The contents of the register can be defined by the **SYStem.Option BAR** command.
CS0 Operation	The CS0 set-up cannot be changed dynamically. Boot sequences must be skipped over.
Interrupt Level 4	Interrupts from the internal peripherals are changed to Level 5 by the emulation hardware. Software must be aware of this, i.e. blocking absolute interrupt levels should be avoided. The **VGE** bit of the 68302 interrupt controller must be set.
Interrupt Level 5	The interrupt level 5 is used for internal interrupts. It cannot be used for interrupt requests from the target system.
Interrupt Vector Generation Enable	The VGE bit in the target CPU must be set on when using vector interrupts (Bit 5 of 0F5H).
Access to area 0f0--0ff	Write to this area go to both CPUs, a write to the BAR register is not allowed. Reads from this area are satisfied by the core CPU. Reading status register values (e.g. Write Protect Detection) will read from the wrong CPU.
CPU Reset in Realtime Emulation	If using external reset signals, the peripherals of the emulation CPU must be switched OFF again. So set a breakpoint to allow the monitor program to do this.

Format: **SYStem.RESetOut**

The peripheral CPU must be reset before starting emulation. When running with a target system this reset is usually done automatically at power on.

In stand-alone mode the user must reset the target CPU by the SYStem.RESetOut command.

SYStem.Option BR0/OR0 CS0 line replacement

Format: **SYStem.Option** <*option*>

<*option*>: **BR0** [<*value*>]
OR0 [<*value*>]

The CS0 line is supported by the emulation CPU. The programming of the CS0 line may not be changed dynamically by the program. The definition must be done by the commands **SYStem.Option BR0** and **SYStem.Option OR0**.

When the program is running from the emulation memory, the CS0 line has no effect and may be switched off. The wait-state definition is done by the target CPU. A value of **-1** disables the **CS0** line.

Format:	**SYStem.Option BARI** [<*value*>]
Format:	**SYStem.Option BAR** [<*value*>|<-1>]

As the emulator must disable the peripherals of the emulation CPU, it is not allowed to write to the BAR register by the target software. The switch-off position is defined by the **SYStem.Option BARI** command (called BAR Internal).

The value of **BAR** is written to the target CPU. It defines the base address of the CPU peripherals, used by the peripheral display window.

The value of **BARI** is written to the emulation CPU to “disable” it’s peripherals.

All settings with BAR, BARI, BR0 and OR0 fields are acknowledged on **SYStem.Up** commands only.

NOTE:	Application programs may read back the BAR register to get the base address of the internal peripherals. For this reason, the BARI should be programmed very similar to the BAR register.

e.g. BAR = BFFF BARI = 0FFF (same address only FC0:)

BAR = B800 BARI = 0800

There are some methods to solve this problem:

1. Automatic ICE detection

   The target program automatically recognizes whether the program is running on the emulator or original CPU. This is done by first reading the BAR register. When running on the emulator the CPU reads the information from the emulation CPU. As the BAR register contains the value defined by **BARI** it can be distinguished from the regular reset value. The mapping of the peripherals is done only, when the BAR register has the reset value (0BFFFH).

   Program example

```
            ....                #$0bfff,$0f2
            ....                skip
            cmpi.w              #$xxxx,$0f2
            bne
            move.w
  skip:
            ....
            ....
```

2. Automatic un-mapping

   The peripherals of the emulation CPU are always mapped off by the monitor program. So when setting a breakpoint or single stepping trough the program the target peripherals may be mapped without affecting the peripherals of the emulation CPU.

3. Setting BAR manually

   The BAR register of the peripheral CPU may be programmed directly by Data.Set commands.

4. Setting BAR automatically

   On setup the target peripherals are automatically mapped, if the **SYStem.Option BAR** command if active (switch off by value = -1). When active, on **SYStem.Up** the target CPU is mapped to the defined position.

All settings with BAR, BARI, BR0 and OR0 fields are acknowledged only on **SYStem.Up** commands.

```
system.option bar 0x00100               ; choose configuration of target
system.option or0 0x09f00
system.option br0 0x0c001

system.up                               ; power up target

data.set 100832 %word 0x09f00 /verify   ; configure registers for
data.set 100830 %word 0x0c001 /verify   ; CS0-line in target-CPU
```

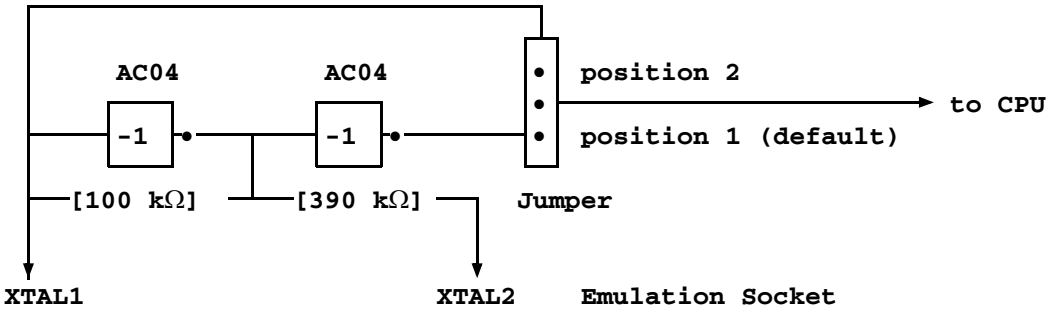
Format: **SYStem.Option IntChange [OFF | ON]**

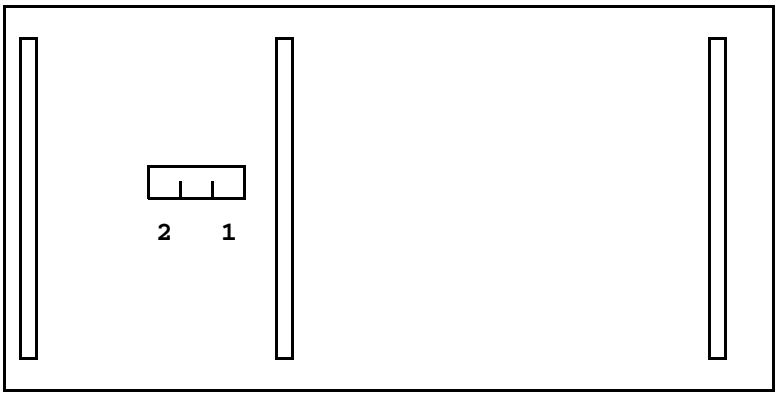
Because the emulation CPU blocks all level 4 interrupts, the interrupt level must be changed to level 5. It is not allowed to use the level 5 by other interrupt controllers on the target system.

All level 4 interrupts are automatically changed to level 5 and a level 5 acknowledge is changed to level 4.

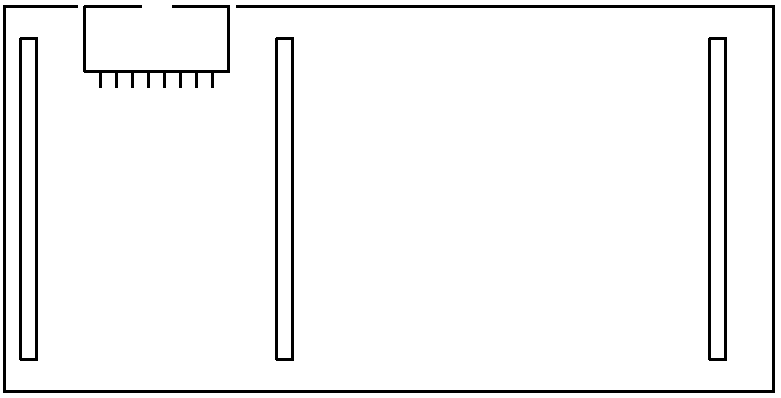
Because both CPUs must have the same clock signal, an oscillator replacement logic is on the adapter board. Jumpered to default position, this circuit can run with crystals and clock signals from TTL logic. In synchronous target systems where the time relation between the clock signal on the target board and the emulation CPU is very critical, the clock input signal may be connected directly to the CPU (Jumper in Position 2).

Oscillator Replacement





Version A



Version B

Position 1 Jumper 5-7 (see chapter On-Circuit Emulation)
Position 2 Jumper 1-3

ONCE Mode Handling

The on-circuit emulation allows to emulate in boards, which use a soldered QFP chip. The adaption is made by a 'Clip-Over' adapter. Two methods can be used to achieve on-circuit emulator:

Disabling the CPU core of the target or disabling the whole target processor.

Disable Target CPU

As the MC68302 processor was not designed for on-circuit emulation, some modifications on the board must be made. The CPU on the target is in RESET by pulling-down the RESET and HALT line. The connection of the RESET and HALT lines between the emulator and the target system must be opened. Both lines must be disabled by the exception control system. Additionally some lines must be cut from the target CPU:

WDOG-

RTS1-

TXD1

RTS3-

NC1

NC3

BRG1

Use Target CPU

The CPU on the target is used for the peripherals instead of the second CPU on the emulation pod. Some modifications become necessary as a result of changed functionality of some pins when the processor is switched from active to peripheral mode (DISCPU).

The CPU on the emulation probe (on bottom PCB) has to be removed. Three lines between the clip over adaptor and the module's PGA target adaptor must be opened.

This lines are:

CS0-/IOUT2- (PGA-K1),

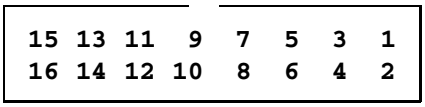
RMC-/IOUT1- (PGA-K2),

AVEC-/IOUT0- (PGA-L9).

It can be done by a PGA socket in between the clip over adaptor and the module's PGA target adaptor which pins are removed for this lines.

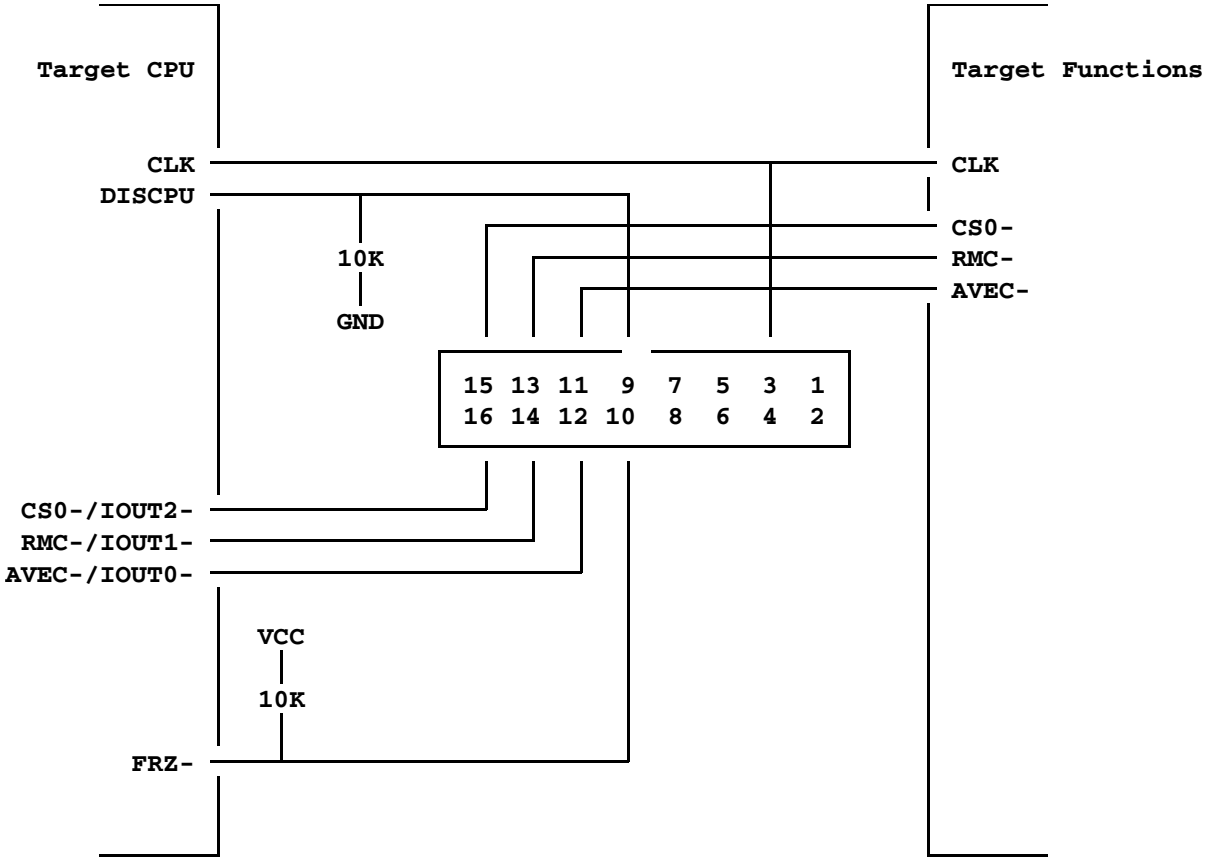
The same lines must be opened on the target and the CPU on the target system must be disabled (DISCPU). The emulation module (Rev. B) contains a connector for on-circuit emulation which has pins for connecting the IOUT0..2- lines of the target CPU and has pins which support the CS0-, RMC- and AVEC-functions of the target system.

Connector on the emulation module



1	EXTAL from target
2	GND
3	Clock to emulator
4	GND
5	Clock from osc. replacement
6	GND
7	Clock to emulator
8	GND
9	DISCPU to target
10	FRZ- to target
11	AVEC- to target
12	IOUT0- from target CPU
13	RMC- to target
14	IOUT1- from target CPU
15	CS0- to target
16	IOUT2- from target CPU

Connections to the target system



Restrictions 68302 Single Chip

FREEZE	All FREEZE bits for DMA and TIMER must be activated. Otherwise the emulation monitor will crash down.
Peripheral Relocation	The emulation control monitor must be relocated to an internal unused address. If internal peripherals are set to 800000, the emulation monitor must be relocated to an other address (**SYStem.Option MonBase** command).
Monitor in Read-Only chip select area	The emulation control monitor will crash, when a read-only chip select is mapped in this range and the WPE (Write Protect Enable) bit is set. The emulation monitor must be relocated to an other address ((**SYStem.Option MonBase** command) in this case.
Interrupt Disable during Single Step	To prevent the single-chip probe from entering pending interrupt routines during single step, the commands **SETUP.IMASKASM** or **SETUP.IMASKHLL** can be used. An alternative can be to use an emulation monitor extension to disable or reset the timer when the emulation is stopped (command **SYStem.MonFile**).
Stepping in Dedicated Interrupt Mode	Single stepping will only work, if the operation mode of the processor and the mode selected by **SYStem.Option Dedicated** are the same. When running in dedicated mode it is recommended to configure the processor to dedicated mode before single stepping.

SYStem.Option MonBase

Monitor relocation

Format: **SYStem.Option MonBase** [<*address*>]

If the emulation monitor address collides with a read-only chip select or with the internal peripherals, it can be relocated to an other address.

The address can be any 64K boundary. The default address is 800000.

Format: **SYStem.Option Dedicated [OFF | ON]**

The operation mode of the interrupt controller must be configured in the same way as the **SYStem.Option Dedicated**.

In dedicated mode it is not possible to enable or disable single interrupt levels.

ONCE Mode Handling

The on-circuit emulation allows to emulate in boards, which use a soldered QFP chip. The adaption is made by a 'Clip-Over' adapter.

As the MC68302 processor was not designed for on-circuit emulation, some modifications on the board must be made. The CPU on the target is in RESET by pulling-down the RESET and HALT line. Both lines must be disabled by the exception control system. Additionally some lines must be cut from the target CPU:

WDOG-

RTS1-

TXD1

RTS3-

NC1

NC3

BRG1

SYStem.Line DRESET

Reset configuration

Format: **SYStem.Line DRESET [OFF | ON]**

If ON, each system reset (HALT and RESET are Low) forces DRESET to low level.

SYStem.Line MODCK

Clock mode selection

Format: **SYStem.Line MODCK [OFF | ON]**

If sys.line MODCK is *ON* the CPU gets MODCKx and CSELECT as set in the system.option, if *OFF* the CPU gets MODCKx and CSELECT of the target system.

SYStem.Option MODCKx

Clock mode selection

Format: **SYStem.Option MODCKx [OFF | ON]**

The clock mode of this CPU is selected by the levels on MODCK0/1 and CSELECT. This lines can be set or reset by the emulator.

Format: **SYStem.Option CS [OFF | ON]**

The clock mode of this CPU is selected by the levels on MODCK0/1 and CSELECT. This lines can be set or reset by the emulator.

SYStem.Option VCO/100 32KHz support

Format: **SYStem.Option VCO/100 [OFF | ON]**

For stand alone emulation with 32KHz use a VCO clock of 3.2 MHz and select option VCO/100. This results a clock speed of 32KHz.

SYStem.Option BRx/NOCSBASE — A20..23 address creation

Format:	**SYStem.Option BR**x [<*value*>]

The emulator creates address A20..23 lines in conjunction with the chip-select-lines. The system window contains chip-select-base fields which must be initialized with the Chip-Select-Baseaddresses.

<value>: The current values of the CPU Chip-Select-Base registers

For not active chipselect areas a special base address can be defined.

Format:	**SYStem.Option NOCSBASE** [<*value*>]

<value>: A 24bit base address.

NOTE:	68PM302: If PCMCIA is disabled this command has no effect. In this mode the CPU supports address lines A20..A23.

SYStem.Option DTACK — DTACK configuration

Format:	**SYStem.Option DTACK [OFF | ON]**

Enables emulator DTACK logic for *timeout detection, wait mapping*... if set to *ON* (target DTACK is input only). Disables emulator DTACK logic but connect target and CPU DTACK if set to *OFF*.

SYStem.Option MODCLK — Clock mode selection

Format:	**SYStem.Option MODCLK [OFF | ON]**

The clock mode of this CPU's is selected by the levels on VCCSYN and MODCLK. The lines can be set or reset by the emulator.

NOTE	The **MODCLK** configuration must match with the target hardware.

SYStem.Option VCCSYN — Clock mode selection

Format:	**SYStem.Option MODCLK [OFF | ON]**
Format:	**SYStem.Option VCCSYN [OFF | ON]**

The clock mode of this CPU's is selected by the levels on VCCSYN and MODCLK. The lines can be set or reset by the emulator.

SYStem.Option ONCE — ON-Circuit emulation (68PM302 only)

Format:	**SYStem.Option ONCE [OFF | ON]**

This option tristates a target CPU. On reset the target CPU's multi function pin TRIS is forced to high level. Don't assert any signal to this pin during reset.

SYStem.Option PCEN — PCMCIA (68PM302 only)

Format:	**SYStem.Option PCEN [OFF | ON]**

NOTE:	The **PCEN** configuration must match with the target hardware.

PCMCIA can be enabled or disabled by the emulator. Additional address lines and bus control lines are available if PCMCIA is disabled.

Format: **SYStem.Option ResetExt [OFF | ON]**

At reset the CPU fetches the target reset configuration if *ON* or emulator reset configuration if *OFF*.

SYStem.Option VCO/100 32KHz support

Format: **SYStem.Option VCO/100 [OFF | ON]**

For stand alone emulation with 32 kHz use a VCO clock of 3.2 MHz and select option VCO/100. This results a clock speed of 32 kHz.

ONCE Mode Handling

On reset the emulator asserts a LOW-signal to the target multi function pin PARITYE- to tristate the target CPU. Don't assert any signal to this pin during reset.

SYStem.Option MBCTL DRAM control

Format:	**SYStem.Option MBCTL** [<*value*>]

The content of this optionfield must match the content of the CPU register MBCTL! The emulator uses this information for DRAM handling.

SYStem.Option BUSW Bus width selection

Format:	**SYStem.Option BUSW**

Defines the voltage level asserted to the multi function pin PARITY1(BUSW) during reset.

NOTE:	Pay attention to the right bus width mapping for the first 8Kbyte of memory (use **MAP.BUS8** or **MAP.NOBUS8** command)!

If the bus width mapping does not match the CPU bus width configuration the emulator monitor program will not work because it gets a wrong reset-vector.

Restrictions GSC38939

GSC38939 The address area 0f0--0ff must me mapped external. It is not possible for the emulator to stop timers when emulation is halted. The watchdog has to be disabled or handled by a background task.

SYStem.Option WSEL — Reset configuration

Format: **SYStem.Option WSEL [OFF | ON]**

During an IIP reset signal pins A0..7 temporarily become mode selection pins. For emulation the WSEL and BUSW0 configuration has to be done.

SYStem.Option BUSW0 — Reset configuration

Format: **SYStem.Option BUSW0 [OFF | ON]**

During an IIP reset signal pins A0..7 temporarily become mode selection pins. For emulation the WSEL and BUSW0 configuration has to be done.

SYStem.Option CKSL — Clock selection

Format: **SYStem.Option CKSL [OFF | ON]**

NOTE:	The **WSEL,BUSW0,CKSL** configuration must match with the target hardware.

Clock selection is done by the CPU signal CKSL. This line can be set or reset by the emulator.

Restrictions 68328

Address range

Only the first 16 MByte address range is supported by the emulator. Address lines A24..31 can be traced **(not with module revision EKLU-1!)** if there is a port analyzer. Programs which use the address range >16 MByte may cause break and mapping problems because of address mirroring!

Address lines A16..23

This address lines must be enabled always! Port function is supported by a port replacement (see later).

LCD DMA-Cycles

All DMA cycles generated by the CPU's LCD controller are mapped to target RAM. The LCD controller has no access to emulation memory! DMA cycles are not visible for the trace.

Watchdog and Timers

The 68328 has no FREEZE option like 68302 to stop timers dynamically. The CPU internal watchdog should be disabled, timers can not be stopped while emulation is halted! Timer interrupts can be handled by a **background task**.

OPFETCH

The 68328 has no Function Code Lines. Without any additional information no program or data fetch detection would be possible. All program memory areas must be defined by the command **MAP.OPFETCH**. Address 10h always is decoded as data fetch. No program break can be done at address 10h!

Dynamic Bussing

All 8-Bit memory areas must be defined by the command **MAP.BUS8**.

NMI/Port C4

The NMI is used for asynchronous break, so this CPU function must be handled with care.

- The NMI/Port-C4 pin must be configured as NMI. Port function is supported by a port replacement (see later).
- it is not allowed to mask the NMI during program execution.
- no asynchronous break will work as long as the NMI status bit is not reset while a target NMI is executed.
- after reset the NMI is masked. It has to be enabled with one of the first instructions after reset.

UDS-/Port C1
LDS-/Port C2

This pins must be configured as UDS- and LDS-. Port function is supported by a port replacement (see later).

DTACK-/Port C5	This pin must be configured as DTACK. Port function is supported by a port replacement (see later). **NOTE:** module revision EKLU-1: For port function one wire on the module's PCB has to be cut. Please contact Lauterbach company.
Port replacement Port A0..7, Port C1,2,4,5	The port replacement can be set to any 256Byte range in the first 16MByte address range. It must not overlap the CPU register range! It has registers for pin assignment, data-direction and data. Please refer to **SYS.Option PRBASE** for more details. **NOTE:** module revision EKLU-1: The port replacement does not work in 8bit-only bus mode!

SYStem.Option BBUSW — Boot bus width

Format:	**SYStem.Option BBUSW [OFF | ON]**

Defines the voltage level asserted to the pin BBUSW. It must match the target BBUSW voltage level.

NOTE:	For applications which run with 8 bit only or 8/16bit mixed bus cycles the **SYStem.Option FAST** must be set. Pay attention to the right bus width mapping. Use **MAP.BUS8** or **MAP.NOBUS8** command! If the bus width mapping does not match the CPU bus width configuration the emulator monitor program will not work.

SYStem.Option MOCLK — Clock selection

Format:	**SYStem.Option MOCLK [OFF | ON]**

Defines the voltage level asserted to the pin MOCLK/PC0 for clock mode selection. It must match the target MOCLK voltage level.

SYStem.Option PORTB — Port B enable

Format:	**SYStem.Option PORTB [OFF | ON]**

Enables Port B for 8bit-only targets.

Format: **SYStem.Option PRBASE [value | OFF]**

Defines the base address of the port replacement for port PA0..7, PC1, PC2, PC4 and PC5.

If no port function is required please set this option to OFF.

The port replacement can be set to any 256Byte range in the first 16MByte address range. The complete 256Byte range is used by the port replacement. It must not overlap the CPU register range! It has registers for pin assignment, data-direction and data. This register must be accessed with byte accesses!

Addresses of the replaced ports:

	module revision EKLU-2...	**module revision EKLU-1**
PADIR	PRBASE+00h	PRBASE+00h
PADATA	PRBASE+02h	PRBASE+01h
PASEL	PRBASE+04h	PRBASE+03h
PCDIR	PRBASE+10h	PRBASE+10h
PCDATA	PRBASE+12h	PRBASE+11h
PCSEL	PRBASE+14h	PRBASE+13h

Port function of port C0 and C6 is still supported by the original CPU registers, not by the port replacement!

Restrictions 68306

Access to BAR/SCR register

The addresses of this registers should be mapped to external, otherwise the current values of these registers cannot be read by the core CPU.

SYStem.Option AMODE — Address mode

Format: **SYStem.Option AMODE [OFF | ON]**

Defines the level on the AMODE line of the MC68306.

SYStem.Option ONCE — On-circuit emulation

Format: **SYStem.Option ONCE [OFF | ON]**

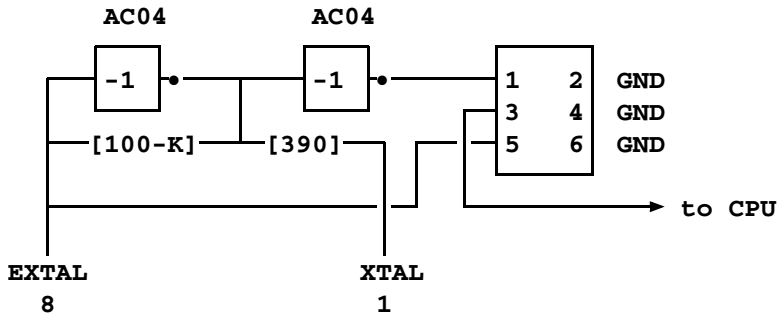
The BERR line is pulled down on target reset. This function disables the MC68306 on the target board. The target CPU only checks this condition after a reset. This requires resetting the target after the option has been activated and before activating the emulation.

SYStem.Option XTAL — Oscillator replacement

Format: **SYStem.Option XTAL [OFF | ON]**

The oscillator replacement is activated.

68306 Clock Generation



Mode	Jumper	Position
Oscillator Replacement	1-3	SYS.O XTAL On
Direct Clock	3-5	SYS.O XTAL Off

Restrictions 68307

Access to BAR/SCR register	The addresses of this registers should be mapped to external, otherwise the current values of these registers cannot be read by the core CPU.
Watchdog	The watchdog on the 68307 bond-out chip should be switched off. Otherwise the watchdog forces a reset while realtime emulation is stopped. Use command **SYStem.Option DISWT ON** and set the **SYStem.Option BAR** value before activating the emulation system.
Port A5..A7	If port pins A5 to A7 are used as i/o ports, the BR- and BGACK-lines of the emulator should be switched off (**X.E BR OFF**)

SYStem.Option DISWT — Watchdog disable

Format: **SYStem.Option DISWT [OFF | ON]**

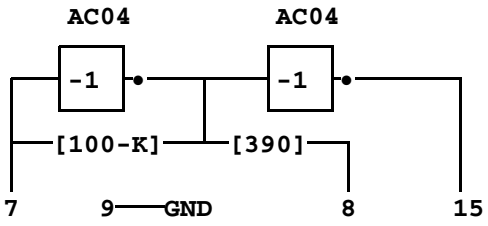
Disable watchdog if *ON*.

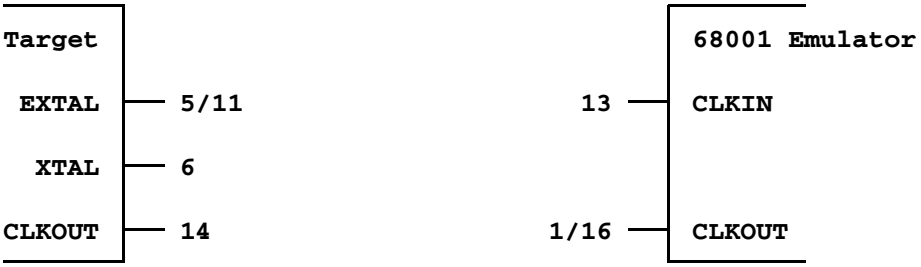
SYStem.Option DRAM — DRAM mode

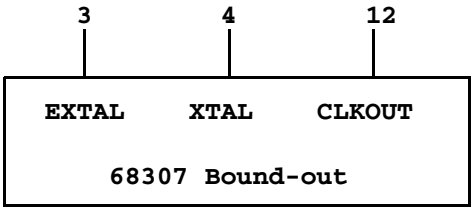
Format: **SYStem.Option DRAM [OFF | ON]**

The Chip Select logic is programmed for DRAM accesses. The internal memory system is switched-over. The emulation memory access is slowed-down.

68307 Clock Generation

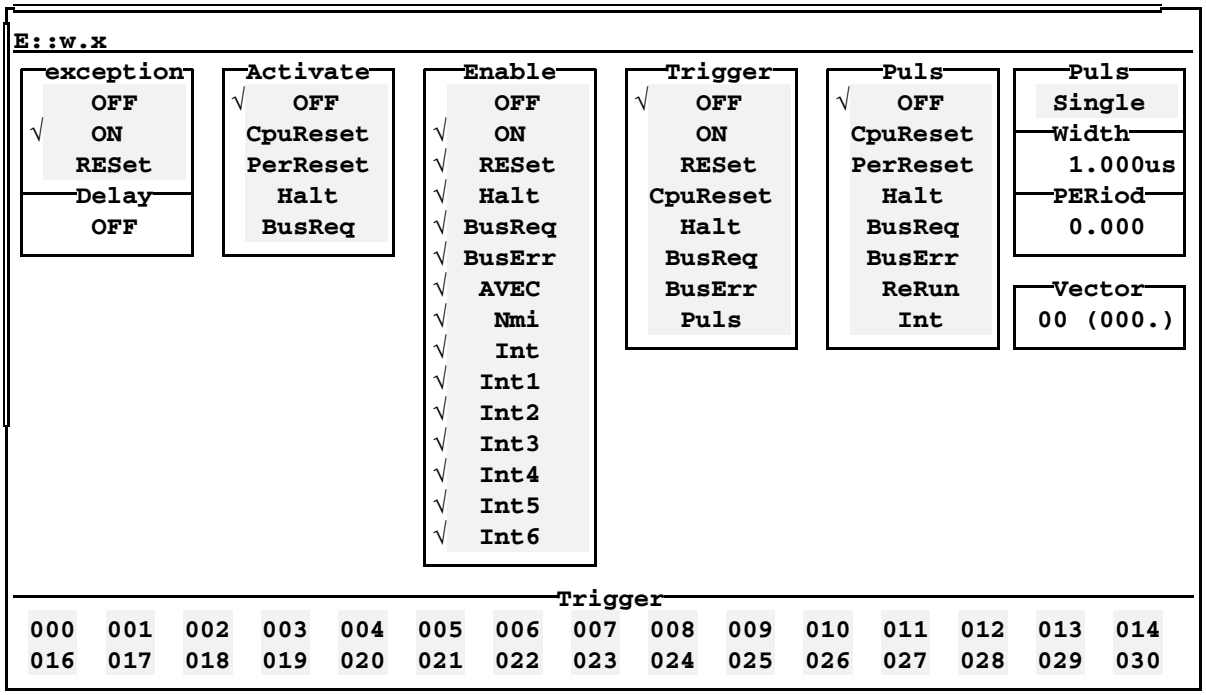






Mode	Jumper Position
Oscillator Replacement	5-7 6-8 1-3 13-15 14-16
Direct Clock	3-5 4-6 7-9 11-13 12-14

Exception Control



Format:	**eXception.Enable ON**
Format:	**eXception.Enable OFF**
Format:	**eXception.Activate OFF**
Format:	**eXception.Enable OFF**

Enable ON Enable all exception lines.

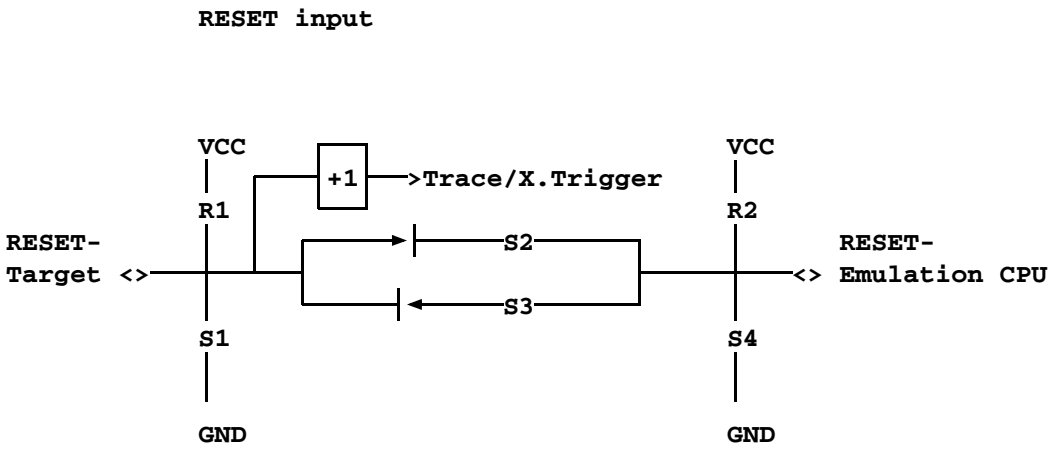
Enable OFF Disable all exception lines.

Activate OFF Inactivate all exception lines.

Pulse OFF Disable all pulse exceptions.

RESET Line

The reset line (input and output) is controlled by a bridge with analog switches and diodes.



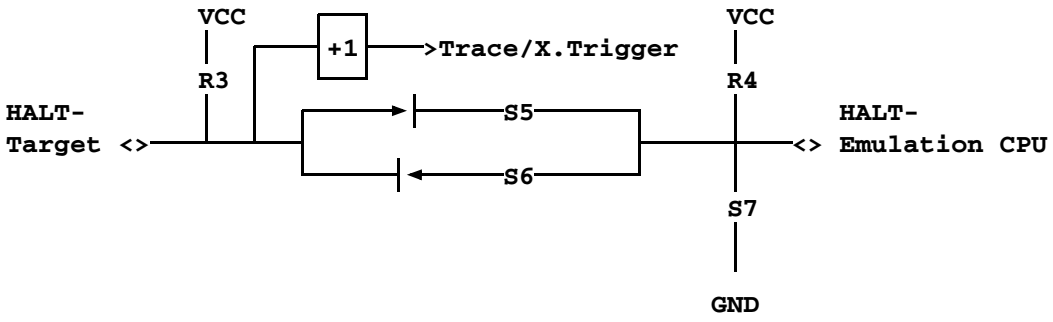
R1 = R2 = 2.7 K

S1	Reset Target	X.Activate PerReset X.Puls PerReset
S2	Reset Out	SYStem.RESetOut Running
S3	Reset In	X.Enable Reset
S4	Internal Reset	Emulator Control X.Activate CpuReset X.Puls CpuReset

Format:	**SYStem.Option RESetOut [OFF | ON]**
Format:	**eXception.Enable RESet [OFF | ON]**
Format:	**eXception.Activate PerReset [OFF | ON]**
Format:	**eXception.Activate CpuReset [OFF | ON]**
Format:	**eXception.Pulse PerReset [OFF | ON]**
Format:	**eXception.Pulse CpuReset [OFF | ON]**
Format:	**eXception.Pulse ReRun [OFF | ON]**

Enable RESet	Enables the Reset line.
Activate PerReset	Activates the Target Reset line.
Activate CpuReset	Activates the CPU Reset line.
Pulse PerReset	Force a pulse to the Target Reset line.
Pulse CpuReset	Force a pulse to the CPU Reset line.
Pulse ReRun	Force a ReRun pulse sequence.

HALT Line



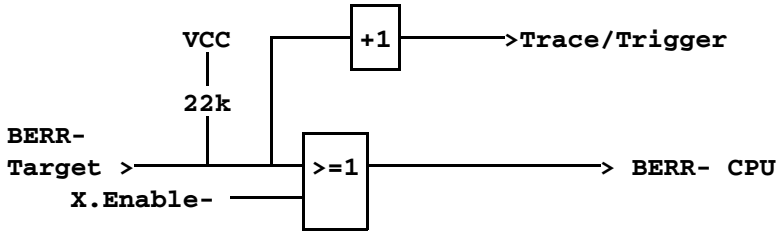

```
R3 = R4 = 2.7 K

S5   HALT Out        Running

S6   HALT In         X.Enable HALT

S7   Internal Halt   Emulator Control
                     X.Activate HALT
                     X.Activate CpuReset
                     X.Puls HALT
                     X.Puls CpuReset
```

Format:	**eXception.Enable Halt [OFF | ON]**
Format:	**eXception.Activate Halt [OFF | ON]**
Format:	**eXception.Pulse Halt [OFF | ON]**

Enable Halt Enables HALT line.

Activate Halt Activates the Halt line.

Pulse Halt Force a pulse to the CPU Halt line.

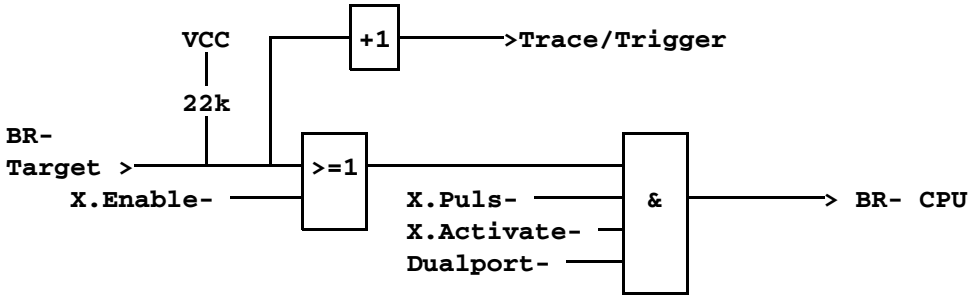


Format:	**eXception.Enable BERR [OFF | ON]**
Format:	**eXception.Pulse BErr [OFF | ON]**

Enable BErr Enables BERR line.

Pulse BErr Force a pulse to the CPU BERR line.

BUSREQ Line



Format:	**eXception.Enable BusReq [OFF | ON]**
Format:	**eXception.Activate BusReq [OFF | ON]**
Format:	**eXception.Pulse BusReq [OFF | ON]**

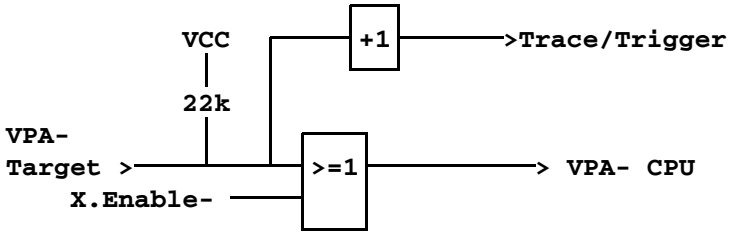
The **BusReq** line must be switched off, when the processor lines are used as an I/O ports (68307 only).

Enable BusReq Enables BusRequest line.

Activate BusReq Activates the **BusReq** line.

Pulse BusReq Force a pulse to the **BusReq** line.

VPA Line



Format:	**eXception.Enable Vpa [OFF | ON]**

Enable VPA Enables VPA line.

Interrupts may be enabled separately for every interrupt level with the dual-chip solution. The single-chip solution allows level-sensitive interrupt control only, when the processor is in non-dedicated mode.

Format:	**eXception.Enable** <*option*>
<*option*>:	**Nmi [ON | OFF]** **INT6 [ON | OFF]** **INT5 [ON | OFF]** **INT4 [ON | OFF]** **INT3 [ON | OFF]** **INT2 [ON | OFF]** **INT1 [ON | OFF]** **INT [ON | OFF]**
Format:	**eXception.Enable AVEC [OFF | ON]**
Format:	**eXception.Pulse Int [OFF | ON]**

Enable INT Enables all interrupt levels.

Enable INT1 Enables interrupt levels 0 to 1.

Enable INT2 Enables interrupt levels 0 to 2.

Enable INT3 Enables interrupt levels 0 to 3.

Enable INT4 Enables interrupt levels 0 to 4.

Enable INT5 Enables interrupt levels 0 to 5.

Enable INT6 Enables interrupt levels 0 to 6.

Enable AVEC Enables AVEC line.

Pulse Int Force an interrupt pulse.

Interrupt Stimulation

The interrupt stimulation is always done with level 7 (NMI level). The pulse width must be at minimum 2 CPU cycles.

Trap Trigger

Every trap can be used as a separate trigger point. Trap trigger is detected by executing a trap only, not by accessing the vector table!

Format:	**eXception.Trigger** <*vector*>
<*vector*>:	**0. … 255.**

Monitor Extensions

A monitor extension is a piece of code that is extending the emulation control monitor. The emulation monitor is responsible for starting and stopping the target program and accessing memory and registers when the target program is stopped. This monitor is running in a hidden memory inside the ECU unit. Extensions must be made available in a binary program. This program must be loaded before activating the emulation by the following command:

Format:	**SYStem.MonFile** <*file*>

The program can contain the following extensions:

Start Target	This part is executed before the target program is started. It can enable timers in the target or reset watchdogs.
Stop Target	This part is executed after the emulation in the target has stopped. It can disable timers or external watchdogs.
Read Memory	User specific memory read. Allows access to special memories, e.g. serial connected EEPROMs. The access is made by the USR: memory class.
Write Memory	User specific memory write. Allows write access to special memories, e.g. programming EEPROM or FLASH memories. The access is made by the USR: memory access class.
Read FPU, Write FPU	Program to read a peripheral 68881/68882 FPU unit. This allows to use the FPU commands to access the unit.

For more details about the definition of the monitor extension and parameter passing see the example file './demo/m68k/etc/monext.asm'.

MAP.BUS8 Bus width mapping

Format: **MAP.BUS8** [<*range*>]

Format: **MAP.NOBUS8** [<*range*>]

Every block in the address space of the CPU has either an 8 or 16 bit bus width. The emulator breakpoint and trace system need this information in realtime in order to work correctly. On setting up the emulator system all areas are defined with 16 bit bus width as default.

```
map.bus8 0x0--0x0fffff          ; maps first 1 MB block for 8 bit

map.nobus8                      ; remaps all to 16 bit
```

The **MAP.RESet** command sets the bus width definition to 16 bit.

MAP.OPFetch Code area mapping

Format: **MAP.OPFetch** [<*range*>]

Format: **MAP.NoOPFetch** [<*range*>]

```
map.opfetch  0x0--0x0fff        ; maps first 4K block for code
```

For the correct operation of the break and trace system the emulator needs to know if an access to memory is an opfetch or a data read cycle. Since the CPU does not provide this information during the cycle this information must be given this way.

Memory Class	Description
FC0	Function-Code 0
FC1	USER-DATA
UD	USER-DATA
FC2	USER-PROGRAM
UP	USER-PROGRAM
FC3	Function-Code 3
FC4	Function-Code 4
FC5	SUPERVISOR-DATA
SD	SUPERVISOR-DATA
FC6	SUPERVISOR-PROGRAM
SP	SUPERVISOR-PROGRAM
FC7	Function-Code 7
CPU	CPU Function-Code
U	User
S	Supervisor
D	Data
P	Program
C	Memory access by CPU
E	Emulation memory access
USR	User defined memory access
	(monitor extension)

When the bussize of the processor is changed, an existing analyzer recording, sampled with a different bussize, will be displayed wrong.

Keywords for the Trigger Unit

Input Event	Meaning	Analyze Hardware			
		ECC8	HAC	HA120	SA120
AutoVECtor	Reading interrupt vector from table (**FC7** and **R** and **VMA**)			X	X
BYTE	Byte transfer			X	X
CPU, FC7	Interrupt acknowledge		X	X	X
Data	Data access (**UD** or **SD**)		X	X	X
DMACycle	DMA cycle			X	X
FC0	Function code 0		X	X	X
FC1,UserData	User data area		X	X	X
FC2, UserProgram	User program area		X	X	X
FC3	Function code 3		X	X	X
FC4	Function code 4		X	X	X
FC5,Supervisor-Data	Supervisor data area		X	X	X
FC6,Supervisor-Program	Supervisor program area		X	X	X
FC7, CPU	Interrupt acknowledge		X	X	X
IACK	Interrupt acknowledge (**FC7** and **Read**)		X	X	X
IPL0 .. IPL2	Interrupt priority level lines			X	X
IR	Interrupt request (**IPL0** or **IPL1** or **IPL2**)			X	X
IR1 .. IR6	Interrupt request 1 to 6			X	X
IR7, NMI	Interrupt request 7, or NMI			X	X
LDS	Data to D0 - D7 (LowerDataStrobe)			X	X
LONG	Double word transfer			X	X
NMI, IR7	Interrupt request 7, or NMI			X	X
Program	Program access (**UP** or **SP**)		X	X	X
TRIPLE	3-byte transfer			X	X

Read	CPU read cycle	X	X	X
ReadData	Data access read (**Read** and **Data**)	X	X	X
Supervisor	Supervisor program or data access (**SP** or **SD**)	X	X	X
SupervisorData, FC5	Supervisor data area	X	X	X
SupervisorProgram, FC6	Supervisor program area	X	X	X
TimeOut	DTACK Timeout			X
UDS	Data to D8 - D15 (UpperDataStrobe)		X	X
User	User program or data access (**UP** or **UD**)	X	X	X
UserData, FC1	User data area	X	X	X
UserProgram, FC1	User program area	X	X	X
VMA	VMA cycle		X	X
Wait0 .. Wait6	Waitstates 0 .. 6		X	X
WaitX	Waitstates greater 6		X	X
WORD	Word transfer		X	X
Write	CPU write cycle	X	X	X
WriteData	Data access write (**Write** and **Data**)	X	X	X

For not CPU-specific keywords, see **non-declarable input variables** in **"ICE/FIRE Analyzer Trigger Unit Programming Guide"** (analyzer_prog.pdf).

WR	Write line
LDS	Lower Data Strobe
UDS	Upper Data Strobe
DMA	A (maybe not visible) DMA cycle was between this and the last record
DMAT	DMA Transfer cycle (only when SYStem.Option DMA is ON)
VMA	VMA cycle
IR	Interrupt request level
IPL.0	Interrupt request line 0
IPL.1	Interrupt request line 1
IPL.2	Interrupt request line 2
BR	Bus request
BG	Bus grant
BGACK	Bus grant acknowledge
BERR	Bus access error
AVEC	Auto vector cycles
HALT	Halt cycle
RES	Reset cycle
IAC	Access to internal peripherals
Wait	Number of inserted wait cycles, for more than 6 a 'X' appears.

Dequeueing

The disassembled lines in the analyzer are displayed prior to the resulting data cycles. This dequeueing fails for commands which have not a constant number of data cycles.

Problems with Prefetches:

- short forward conditional branches to addresses already prefetched

Default Keywords for the Port Analyzer

IPL0	external interrupt request lines
IPL1	
IPL2	
BCLR	
CS0	Chip select outputs
CS1	
CS2	
CS3	
PB8	parallel port
PB9	
BP10	
BP11	
DREQ	DMA control lines
DACK	
DONE	
RMC	RMC cycle lines
IACK7	External interrupt lines
IACK6	
IACK1	
TIN1	timers
TOUT1	
TIN2	
TOUT2	
WDOG	
RXDx	Communication channel lines (1,2,3)
TXDx	
RCLKx	

TCLKx

CTSx

RTSx

CDx

BRGx

X0 .. **X7**	External trace inputs
BNK0 .. **BNK7**	External bank probe inputs

CPU	CPU control lines
AS	Address Strobe
LDS	Lower Data Strobe
UDS	Upper Data Strobe
R/W	Read/Write
DTACK	DTACK
BR	Bus Request
BG	Bus Grant
BGACK	Bus Grant Acknowledge
RESET	RESET line
HALT	HALT line
INT	Interrupt Lines
IRQ1	Interrupt Request 1
IRQ4	Interrupt Request 4
IRQ7	Interrupt Request 7
IACK1	Interrupt Ackn. 1
IACK4	Interrupt Ackn. 4
IACK7	Interrupt Ackn. 7
UART	UART lines
RXDA	Receiver Data A
TXDA	Transmitter Data A
RXDB	Receiver Data B
TXDB	Transmitter Data B
RTSA	Request to send A
CTSA	Clear to send A
RTSB	Request to send B
CTSB	Clear to send B
A	Port A
A0 .. **A7**	
B	Port B

B0 .. **B7**

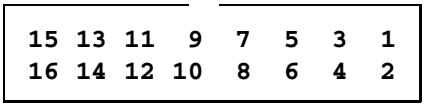
X0 .. **X7**	External Connector
Y0 .. **Y7**	External Connector
BNK0..**BNK7**	Bank Probe

68307 Keywords for the Port Analyzer

CPU	CPU control lines
AS	Address Strobe
LDS	Lower Data Strobe
UDS	Upper Data Strobe
R/W	Read/Write
DTACK	DTACK
BR	Bus Request
BG	Bus Grant
BGACK	Bus Grant Acknowledge
RD	Read Strobe 8051
WR	Write Strobe 8051
ALE	Address Latch Enable
IRQ7	Interrupt Request 7
RESET	RESET line
HALT	HALT line
CS	Chip selects
CS0 .. **CS3**	Chip select 0..3
INT	Interrupt lines
INT1 .. **INT8**	Interrupt Input
A	Port A
A0 .. **A4**	
B	Port B
B0 .. **B7**	

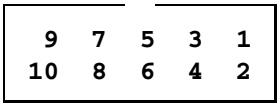
X0 .. **X7**	External Connector
Y0 .. **Y7**	External Connector
BNK0..**BNK7**	Bank Probe

Default Connector for Additional Trace Channels



1	X0
2	X1
3	X2
4	X3
5	X4
6	X5
7	X6
8	X7
9	BNK0
10	BNK1
11	BNK2
12	BNK3
13	BNK4
14	BNK5
15	BNK6
16	BNK7

Single Chip 68302 Connector for Additional Trace Channels



Two connectors,
connector B is located
on the Pin-A1 side of the CPU

A1	X0
A2	X1
A3	X2
A4	X3
A5	X4
A6	X5
A7	X6
A8	X7
A9	GND
A10	GND

B1	BNK0
B2	BNK1
B3	BNK2
B4	BNK3
B5	BNK4
B6	BNK5
B7	BNK6
B8	BNK7
B9	GND
B10	GND

Language	Compiler	Company	Option	Comment
ADA	ALSYS-ADA		IEEE	limited support (IEEE)
ADA	TELESOFT-ADA	Telesoft	IEEE	limited support (IEEE)
ASM	RTOS	IEP GmbH	SYM/LOC	Source level debugging
ASM	ASM68K	Mentor Graphics Corporation	IEEE	Source level debugging
ASM	VERSADOS-ASM	NXP Semiconductors	VERSADOS	symbols only
ASM	OS-9-ASSEMBLER	Radisys Inc.	ROF	Source level debugging
ASM	AS68	TASKING	IEEE	
C	HP-64000-C		HP	no type/locals info
C	ORGANON	CAD-UL ElectronicServices GmbH	BOUND	
C	C68K	Cosmic Software	COSMIC	
C	GNU-C	GNU Compiler Collection	ELF/DWARF	
C	GNU-C	GNU Compiler Collection	COFF	
C	GNU-C	GNU Compiler Collection	ELF/DWARF	
C	GREEN-HILLS-C	Greenhills Software Inc.	COFF	
C	ICC68K	Introl Corporation	ICOFF	
C	MCC	Mentor Graphics Corporation	IEEE	
C	HT-68K	Microchip Technology Inc.	HITECH	
C	SCC68K	Microsemi Corp.	COFF	
C	HICROSS-68K	NXP Semiconductors	HICROSS	
C	CC68K	NXP Semiconductors	COFF	
C	ULTRA-C	Radisys Inc.	ROF	OS/9 compilers
C	OS/9-C	Radisys Inc.	ROF	
C	CROSSCODE-C	SDSI	SDS	
C	SUN3-CC	Oracle Corporation	DBX	
C	ICC68K	TASKING	COFF	
C	ICC68K	TASKING	IEEE	
C	TT-68K	TASKING	IEEE	
C	TCC68K	TASKING	AOUT	only source and syms
C	TEKTRONIX-C	Tektronix	COMFOR	

Language	Compiler	Company	Option	Comment
C	D-CC	Wind River Systems	IEEE	
C	D-CC	Wind River Systems	ELF/DWARF	
C++	ORGANON-C++	CAD-UL ElectronicServices GmbH	BOUND	
C++	GNU-C++	GNU Compiler Collection	DBX	
C++	GNU-C++	GNU Compiler Collection	ELF/DWARF	
C++	CCC68K	Mentor Graphics Corporation	IEEE	
C++	HICROSS-68K	NXP Semiconductors	HICROSS	
C++	CODEWARRIOR	NXP Semiconductors	ELF/DWARF	
C++	CROSSCODE-C++	SDSI	SDS	
C++	D-C++	Wind River Systems	ELF/DWARF	
MODULA	MOD68K	Introl Corporation	ICOFF	
MODULA	MCS2	Multichannelsystems GmbH	COFF	
MODULA	MCDS	NXP Semiconductors	MCDS	
PASCAL	MPC	Mentor Graphics Corporation	IEEE	
PEARL	RTOS	IEP GmbH	SYM/LOC	no type/locals info

CPU	Tool	Company	Host
	WINDOWS CE PLATF. BUILDER	-	Windows
	CODE::BLOCKS	-	-
	C++TEST	-	Windows
	ADENEO	-	
	CODEWRIGHT	Borland Software Corporation	Windows
	CODE CONFIDENCE TOOLS	Code Confidence Ltd	Windows
	CODE CONFIDENCE TOOLS	Code Confidence Ltd	Linux
	EASYCODE	EASYCODE GmbH	Windows
	ECLIPSE	Eclipse Foundation, Inc	Windows
	RHAPSODY IN MICROC	IBM Deutschland GmbH	Windows
	RHAPSODY IN C++	IBM Deutschland GmbH	Windows
	CHRONVIEW	Inchron GmbH	Windows
	LDRA TOOL SUITE	LDRA Technology, Inc.	Windows
	UML DEBUGGER	LieberLieber Software GmbH	Windows
	TRACEANALYZER	LUXOFT	Windows
	SIMULINK	The MathWorks Inc.	Windows
	ATTOL TOOLS	MicroMax Inc.	Windows
	VISUAL BASIC INTERFACE	Microsoft Corporation	Windows
	LABVIEW	NATIONAL INSTRUMENTS Corporation	Windows
	TPT	PikeTec GmbH	Windows
	X-TOOLS / X32	PTC	Windows
	CANTATA	QA Systems Ltd	Windows
	RAPITIME	Rapita Systems Ltd.	Windows
	TESSY	Razorcat Development GmbH	Windows
	DA-C	RistanCASE	Windows
	ECU-TEST	TraceTronic GmbH	Windows
	UNDODB	Undo Software	Linux
	TA INSPECTOR	Vector	Windows

CPU	Tool	Company	Host
	VECTORCAST UNIT TESTING	Vector Software	Windows
	VECTORCAST CODE COVERAGE	Vector Software	Windows
68K	OS68 DEBUGGER	Enea OSE Systems	-
68K	SDT CMICRO	IBM Deutschland GmbH	Windows
68K	DIAB RTA SUITE	Wind River Systems	Windows

Company	Product	Comment
PTC	AdaWorld ARTK	
KadakProducts Ltd.	AMX	
Oracle Corporation	ChorusOS	
CMX Systems Inc.	CMX-RTX	
Synopsys, Inc	MQX	2.40 and 2.50, 3.6
	MTOS-UX	
Mentor Graphics Corporation	Nucleus PLUS	
Radisys Inc.	OS-9	
Enea OSE Systems	OSE Classic	(OS68)
Enea OSE Systems	OSE Delta	4.x and 5.x
	RealTime Craft	(XEC68k)
Quadros Systems Inc.	RTXC 3.2	
IBM Corp.	SDT-Cmicro	
-	uCLinux	Kernel Version 2.4 and 2.6, 3.x
Mentor Graphics Corporation	VRTX32	
Mentor Graphics Corporation	VRTXmc	
Mentor Graphics Corporation	VRTXsa	
Wind River Systems	VxWorks	5.x and 6.x

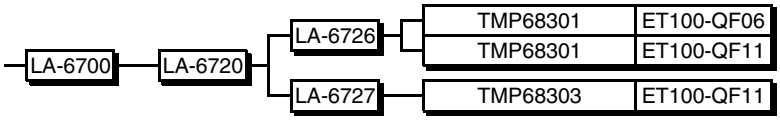
Emulation Frequency

The emulation probe is designed for running with CPUs up to 33 MHz. The max. speed is limited by the memory speed and the wait states used for memory access.

No adapters necessary !

Module	CPU	F-W0-15	F-W0-35	S-W0-15	S-W0-35	S-W1-15	S-W1-35	DRAM
LA-6720	TMP68301	16.7+	16.7+	16.7+	16.7+	16.7+	16.7+	16.7
LA-6720	TMP68303	16.7+	16.7+	16.7+	16.7+	16.7+	16.7+	16.7

Module Overview



Order Information

Order No.	Code	Text
LA-6700	ICE-68300	ICE-68300 Base Module
LA-6710	M-MC68302-D	Module MC68302 PGA (Dual-Chip Version)
LA-6711	M-MC68302-S	Module MC68302 PGA (Single-Chip Version)
LA-6703	M-MC68302-S-3.3V	Module MC68302 PGA (Single-Chip Version) 3.3V
ET-1008	ET-68302-C	PGA to QFP 68302 Clip-Over Adapter
ET-1010	ET-68302-S	PGA to QFP 68302 Surface Mountable Adapter
LA-1007	68302-EYA-144	ET132 to ET144 for 68302
LA-1046	A-68302-ET144	68302 Adapter PGA132 to ET144-QF63
LA-6712	M-MC68302-D-3.3V	Module MC68302 TQFP (Dual-Chip Version) 3.3V
LA-1045	A-68302/3.3V-PGA	68302/3.3V Adapter ET144-QF63 to PGA132
LA-6707	M-MC68EN302	Module MC68EN302
LA-1047	A-68EN302-PGA	68EN302 Adapter ET144 to PGA
LA-6708	M-MC68PM302	Module MC68PM302 (Single-Chip Version)
LA-6706	M-MC68PM302-3.3V	Module MC68PM302 (Single Chip Version) 3.3 V
LA-1048	A-68PM302-PGA	68PM302 Adapter ET144 to PGA
LA-6709	M-MC68LC302	Module MC68LC302
LA-6704	M-MC68LC302-3.3V	Module MC68LC302 3.3V
LA-1049	A-68LC302-PGA	68LC302 Adapter ET100 to PGA
LA-6705	M-MC68356	Module MC68356
LA-6714	M-MC68328	Module MC68328
LA-6717	M-KELVIN	Module KELVIN
LA-6720	M-MC68HC001	Module MC68HC001
LA-6721	A-MC68HC001-PLCC	Adapter MC68HC001-PLCC68
LA-6722	A-MC68HC001-DIL64	Adapter MC68HC001-DIL64
LA-6723	A-MC68HC001-DIL48	Adapter MC68HC001-DIL48
LA-6724	A-MC68HC001-PLCC52	Adapter MC68HC001-PLCC52
LA-6725	A-MC68HC001-PGA	Adapter MC68HC001-PGA
LA-6729	A-MC68EC000-PLCC	Adapter MC68EC000-PLCC68
LA-6767	M-MC68SEC000	Module MC68SEC000
LA-6726	A-68301	Adapter MC68HC001-301
LA-6727	A-TMP68303	Adapter MC68HC001-303
LA-6716	A-MC68307	Adapter MC68HC001-MC68307

Order No.	Code	Text
LA-6718	**M-MC68306**	**Module MC68306**
LA-6890	**ET-68306-144-S**	**ET144-Surface Mountable Adapter for MC68306**
LA-6891	**68306-YA-144**	**YAMAICHI-Adapter to ET144 for MC68306**
LA-6719	**M-MC68308**	**Module MC68308**
Additional Options		
LA-6450	PA64	Port Analyzer

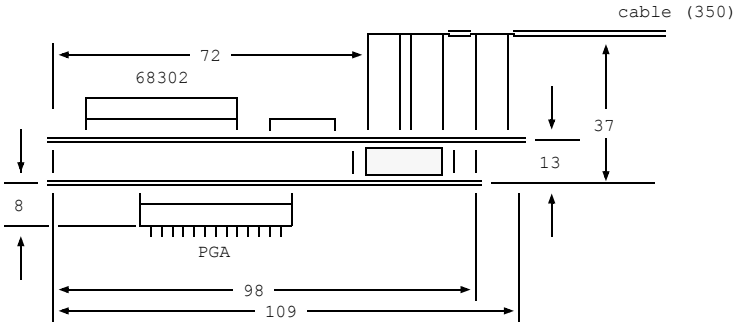
Operation Voltage

No other voltage ranges available !

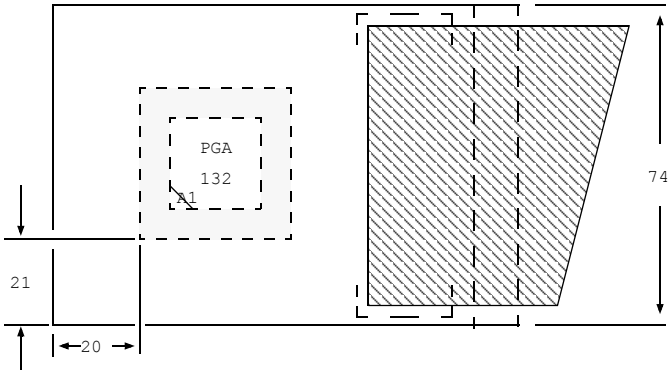
Physical Dimensions

Dimension

LA-6703 M-MC68302-S-3.3V



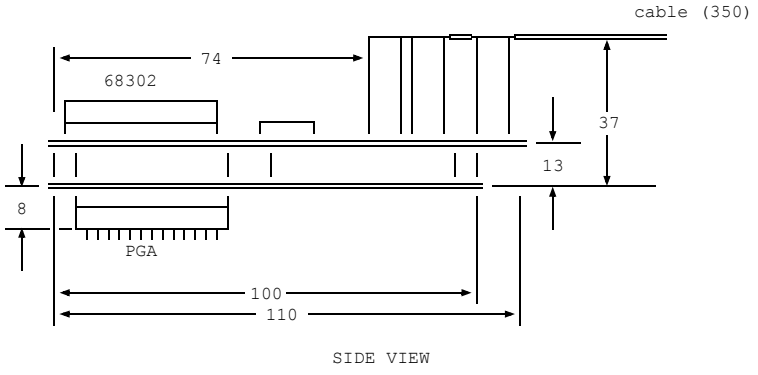
SIDE VIEW



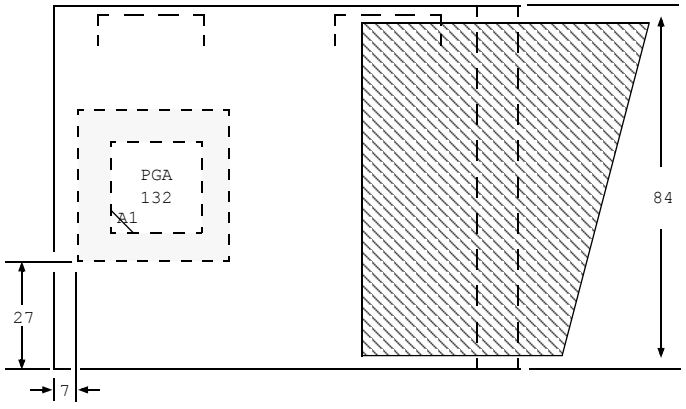
TOP VIEW (all dimensions in mm)

Dimension

LA-6710 M-MC68302-D



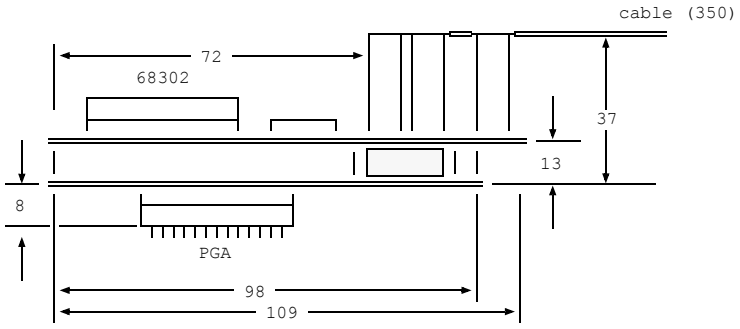
SIDE VIEW



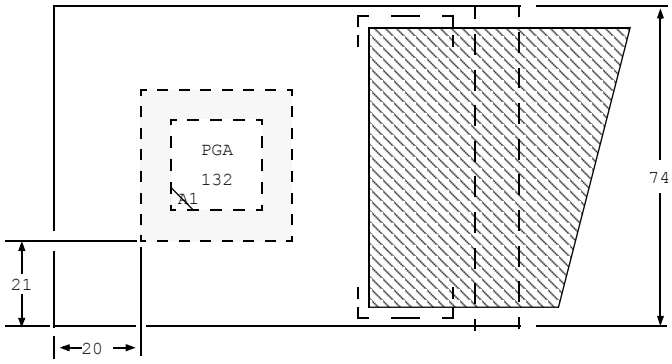
TOP VIEW (all dimensions in mm)

Dimension

LA-6711 M-MC68302-S



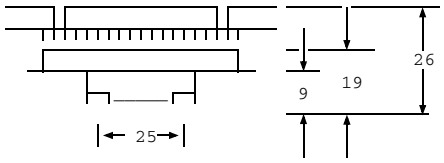
SIDE VIEW



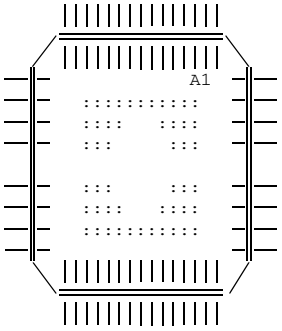
TOP VIEW (all dimensions in mm)

Dimension

ET-1008 ET-68302-C

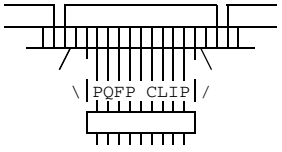


SIDE VIEW

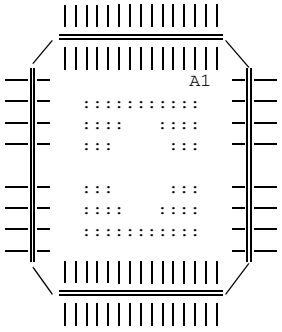


TOP VIEW (all dimensions in mm)

ET-1009 ET-68302-CF



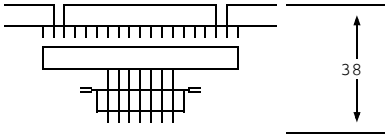
SIDE VIEW



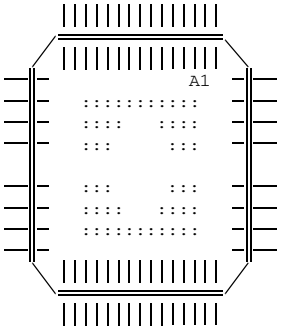
TOP VIEW (all dimensions in mm)

Dimension

ET-1010 ET-68302-S



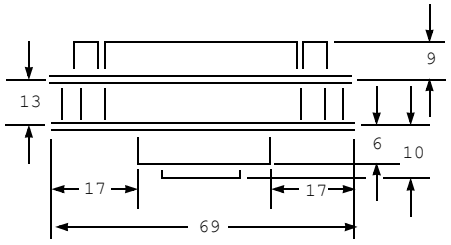
SIDE VIEW



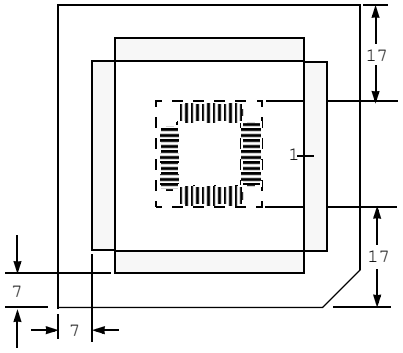
TOP VIEW (all dimensions in mm)

Dimension

LA-1007 68302-EYA-144



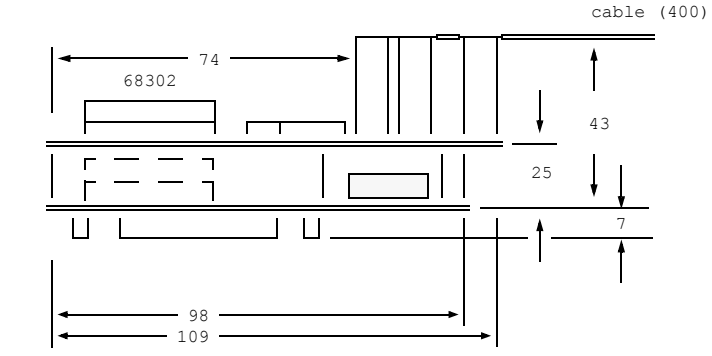
SIDE VIEW



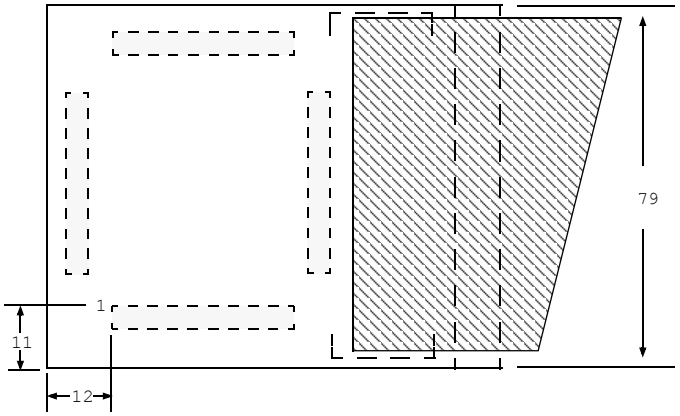
TOP VIEW

Dimension

LA-6712 M-MC68302-D-3.3V



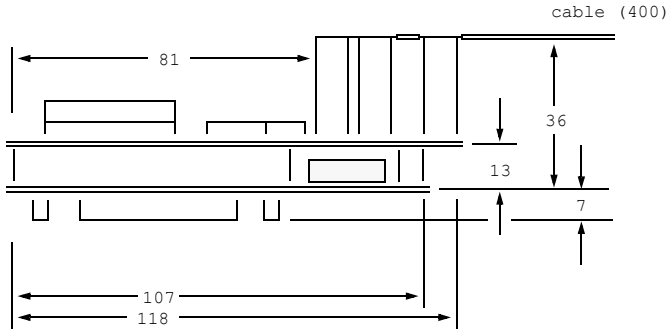
SIDE VIEW



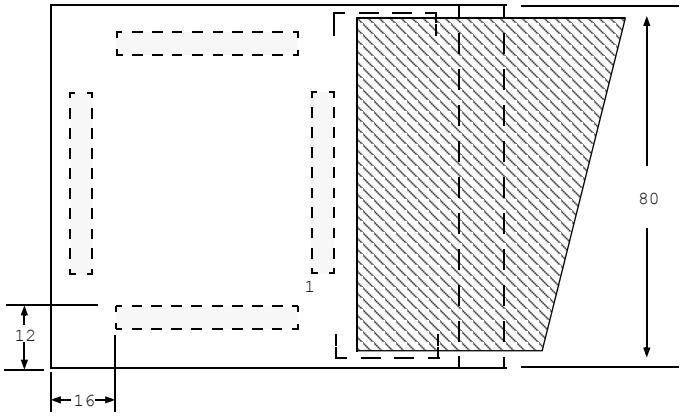
TOP VIEW (all dimensions in mm)

Dimension

LA-6707 M-MC68EN302



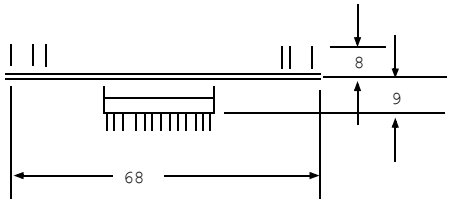
SIDE VIEW



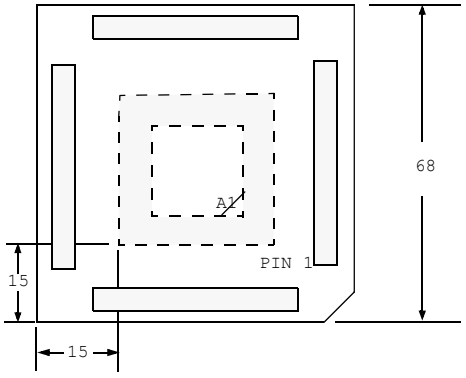
TOP VIEW (all dimensions in mm)

Dimension

LA-1047 A-68EN302-PGA



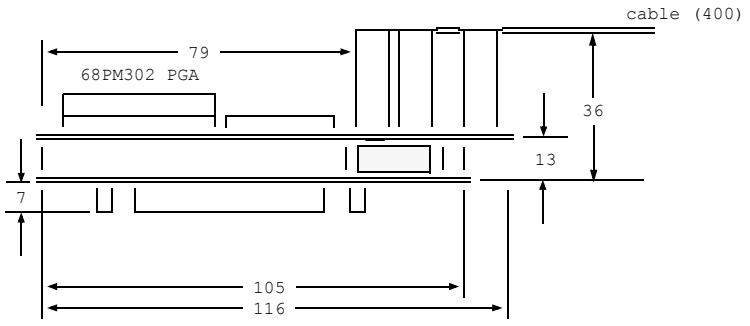
SIDE VIEW



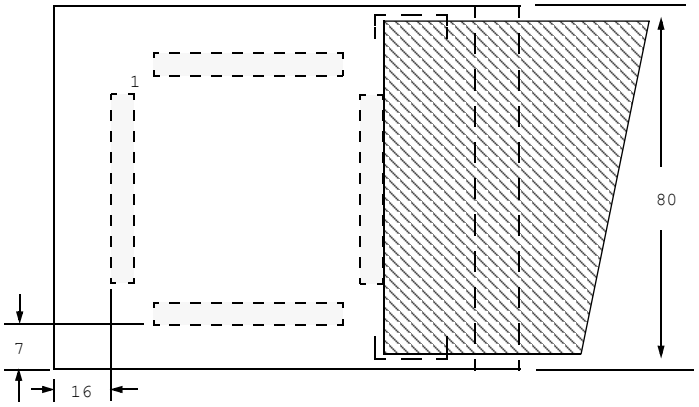
TOP VIEW (all dimensions in mm)

Dimension

LA-6708 M-MC68PM302
LA-6706 M-MC68PM302-3.3V



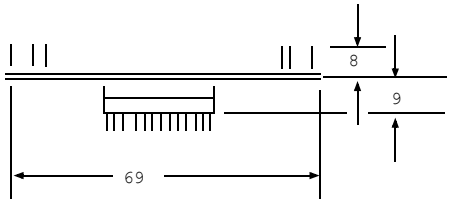
SIDE VIEW



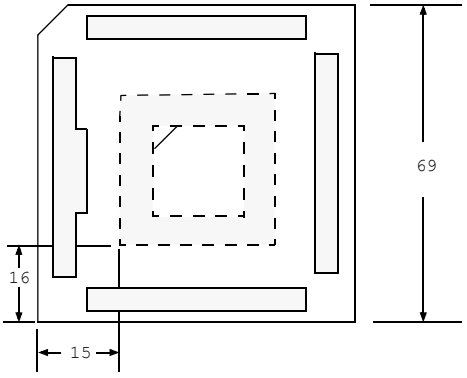
TOP VIEW (all dimensions in mm)

Dimension

LA-1048 A-68PM302-PGA



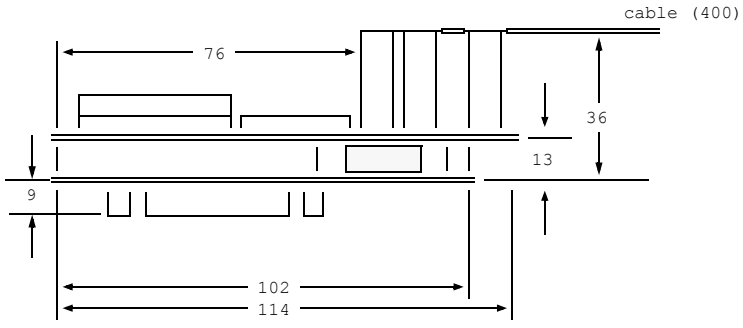
SIDE VIEW



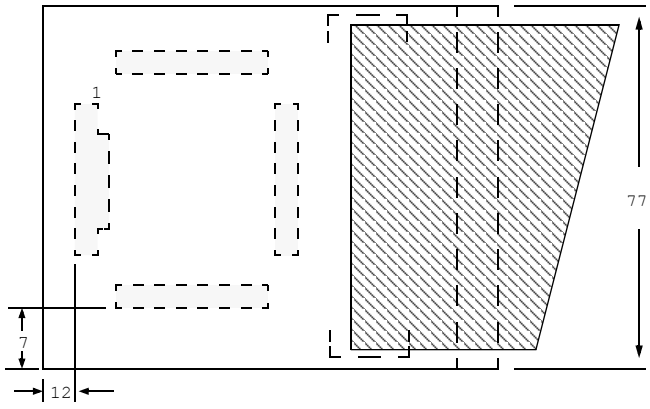
TOP VIEW (all dimensions in mm)

Dimension

LA-6709 M-MC68LC302
LA-6704 M-MC68LC302-3.3V



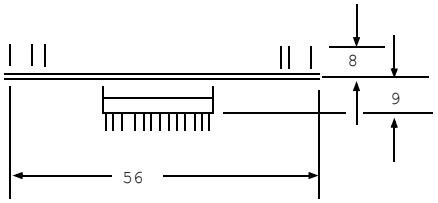
SIDE VIEW



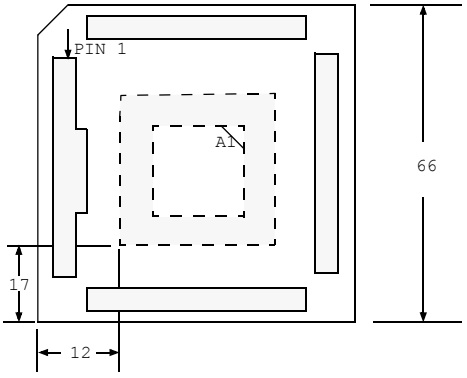
TOP VIEW (all dimensions in mm)

Dimension

LA-1049 A-68LC302-PGA



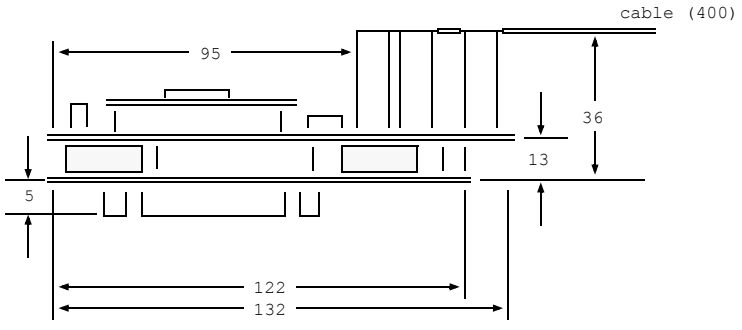
SIDE VIEW



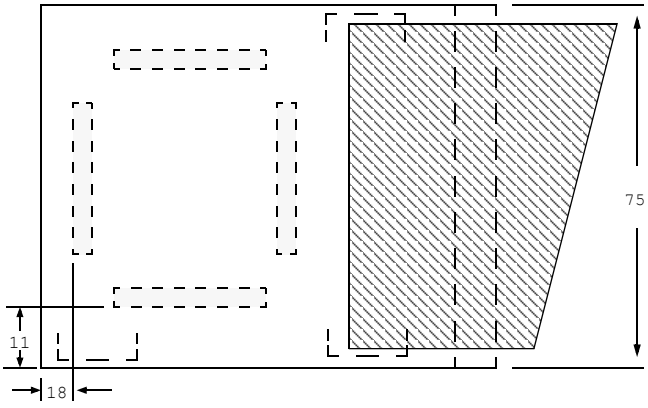
TOP VIEW (all dimensions in mm)

Dimension

LA-6705 M-MC68356



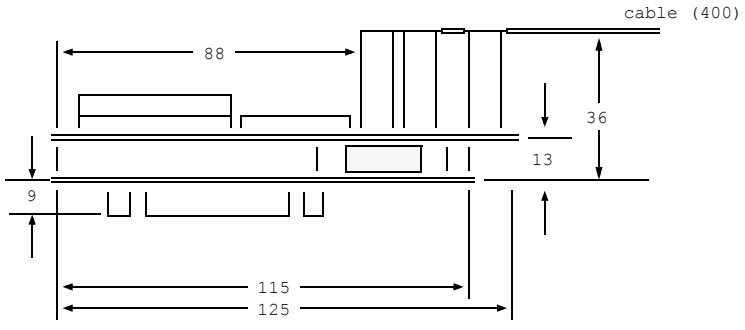
SIDE VIEW



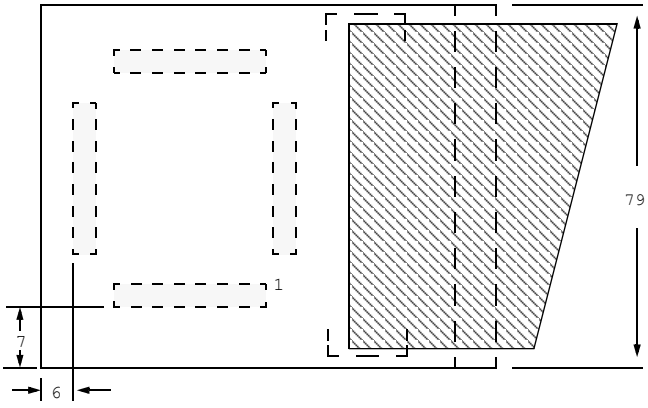
TOP VIEW (all dimensions in mm)

Dimension

LA-6714 M-MC68328



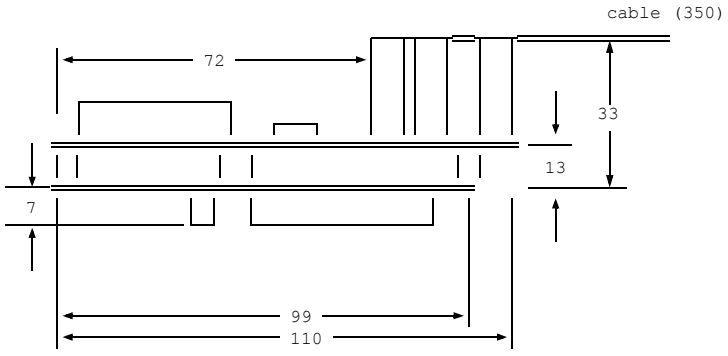
SIDE VIEW



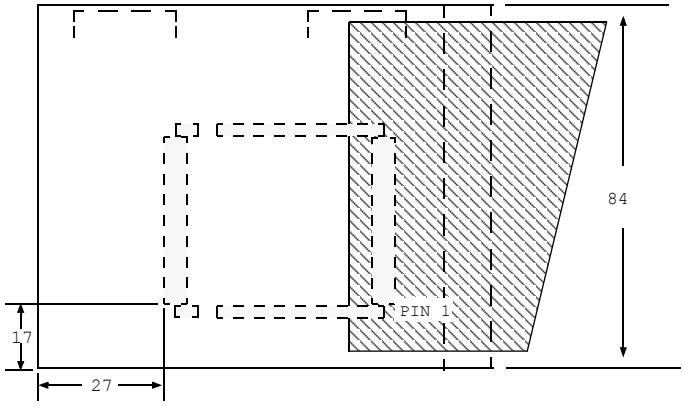
TOP VIEW (all dimensions in mm)

Dimension

LA-6720 M-MC68HC001



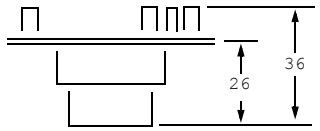
SIDE VIEW



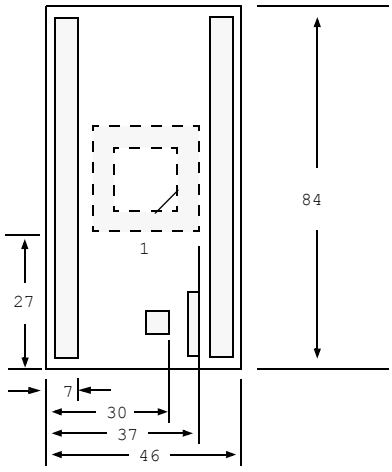
TOP VIEW (all dimensions in mm)

Dimension

LA-6721 A-MC68HC001-PLCC
LA-6729 A-MC68EC000-PLCC



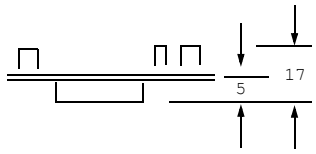
SIDE VIEW



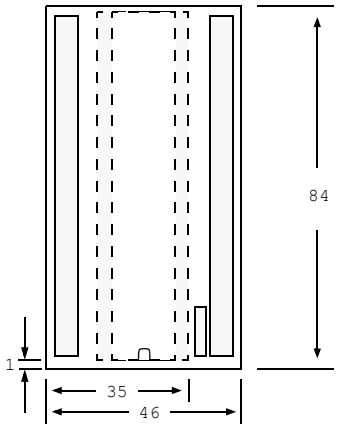
TOP VIEW (all dimensions in mm)

Dimension

LA-6722 A-MC68HC001-DIL64



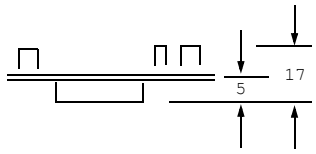
SIDE VIEW



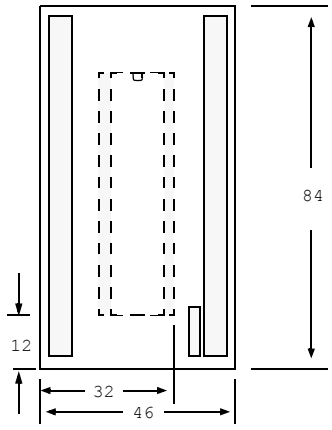
TOP VIEW (all dimensions in mm)

Dimension

LA-6723 A-MC68HC001-DIL48



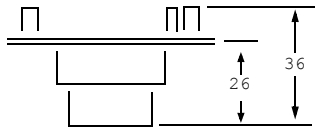
SIDE VIEW



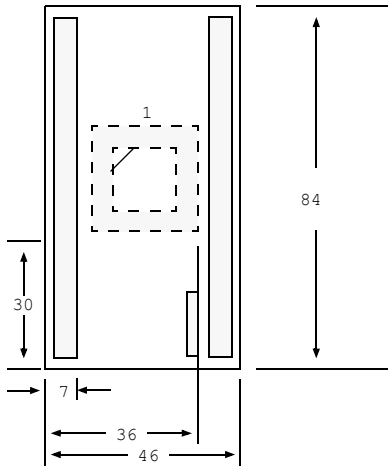
TOP VIEW (all dimensions in mm)

Dimension

LA-6724 A-MC68HC001-PLCC52



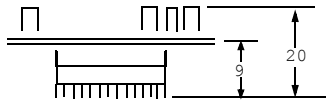
SIDE VIEW



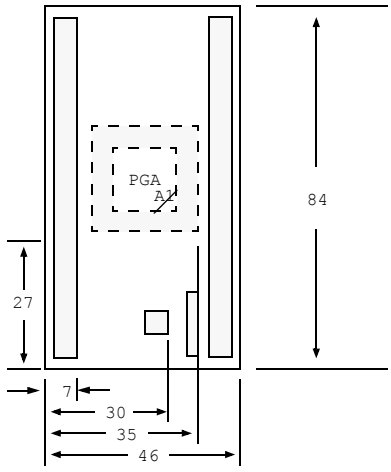
TOP VIEW (all dimensions in mm)

Dimension

LA-6725 A-MC68HC001-PGA



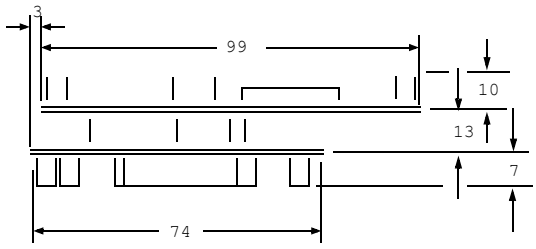
SIDE VIEW



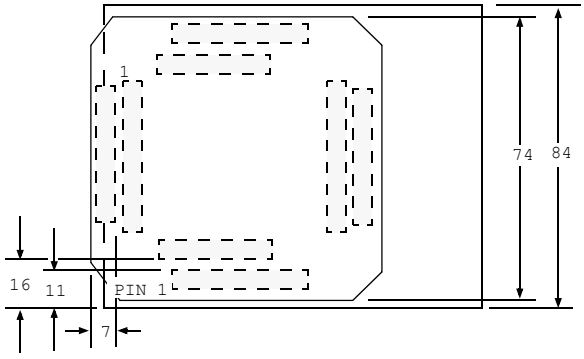
TOP VIEW (all dimensions in mm)

Dimension

LA-6726 A-68301



SIDE VIEW

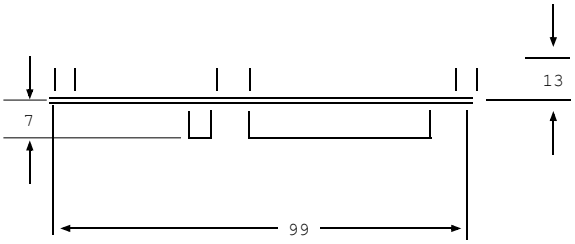


TOP VIEW (all dimensions in mm)

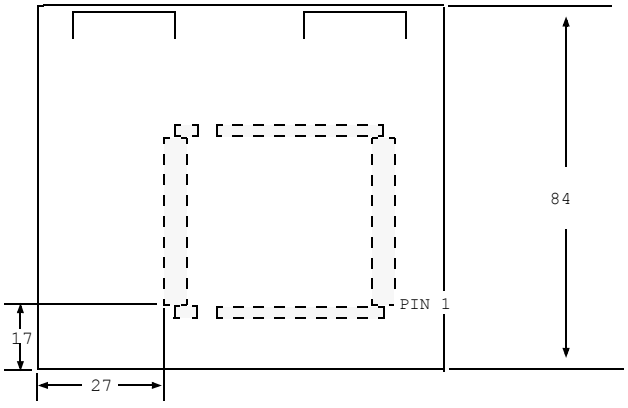
LA-6727 A-TMP68303

Dimension

LA-6716 A-MC68307



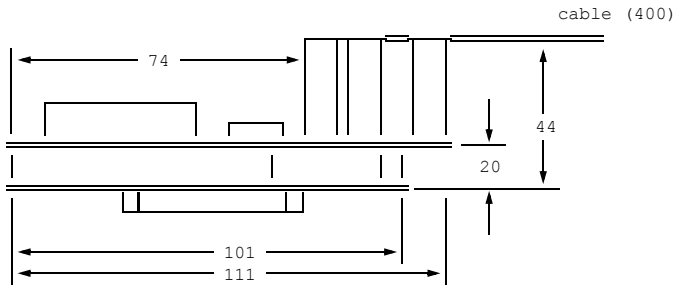
SIDE VIEW



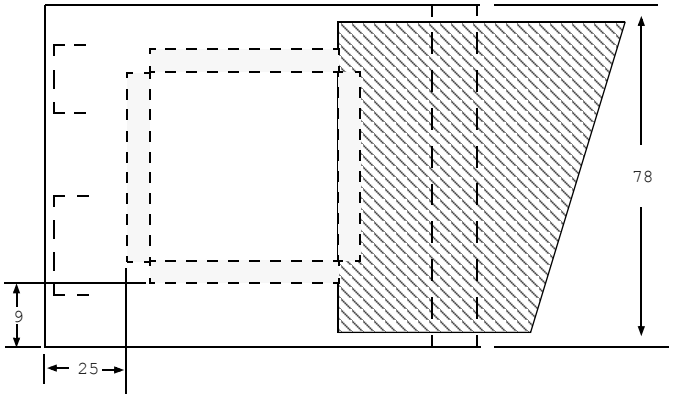
TOP VIEW (all dimensions in mm)

Dimension

LA-6718 M-MC68306



SIDE VIEW

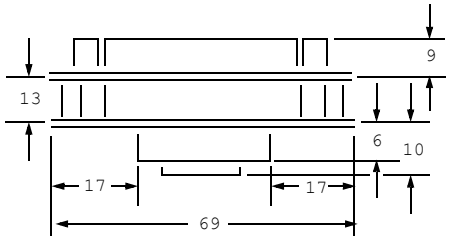


TOP VIEW (all dimensions in mm)

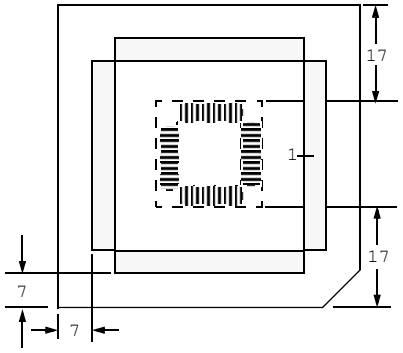
LA-6890 ET-68306-144-S

Dimension

LA-6891 68306-YA-144



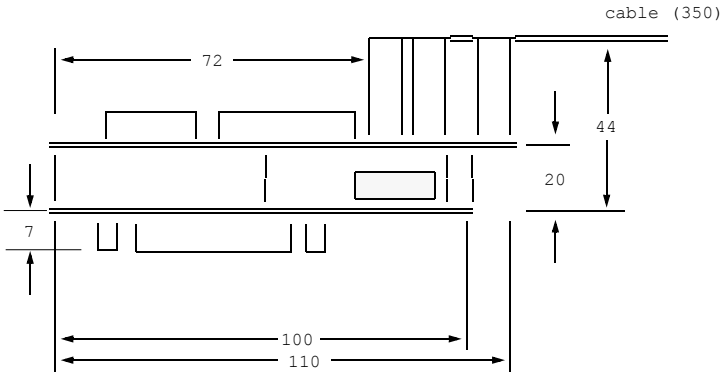
SIDE VIEW



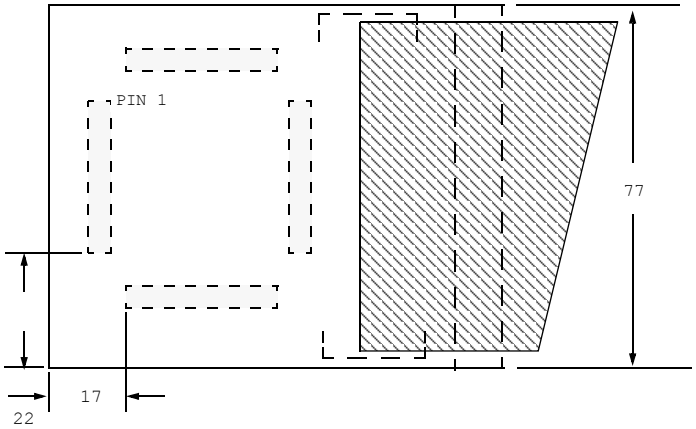
TOP VIEW

Dimension

LA-6719 M-MC68308



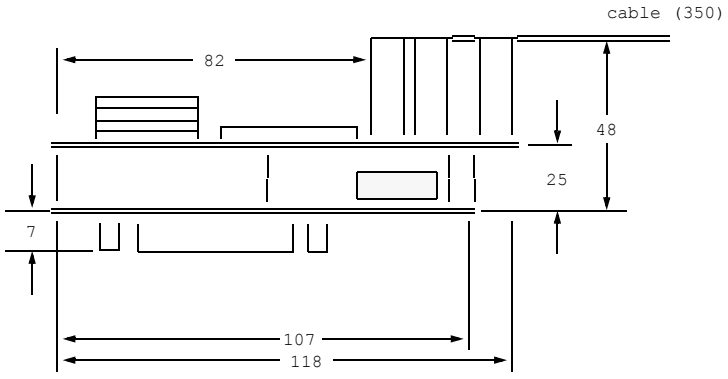
SIDE VIEW



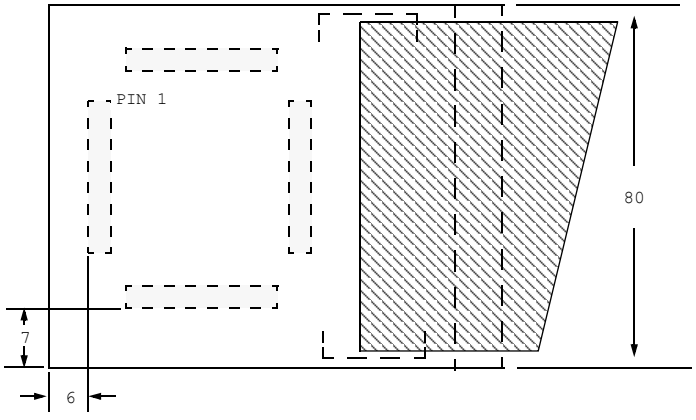
TOP VIEW (all dimensions in mm)

Dimension

LA-6717 M-KELVIN



SIDE VIEW



TOP VIEW (all dimensions in mm)