



## ST90158-EMU2

### ST9 HDS2V2 EMULATOR FOR ST90158

DOC-ST90158-EMU2/DS

#### EMULATOR DESCRIPTION

- Emulator composed of:
  - a mainboard included in a box powered by an external power supply
  - a ST90158 specific probe
  - a Windows based GNU debugger software running under Windows-3.1x / 9x / NT4.0
- Emulator connected to the user application through the probe:
  - ST90158-EMU adapts to QFP80 or LCC84 package
- Emulator connected to a host PC or compatible with a standard parallel cable
- 3V or 5V +/- 10% operating voltage
- Up to 16 MHz internal clock operation at 5V and 14 MHz at 3V

#### HARDWARE FEATURES

- Clock source selectable:
  - 4 MHz oscillator on probe
  - 8 MHz quartz on probe
  - TTL source from application
- Application power up detection
- 9 external input triggers:
  - 1 input trigger on subclck connector
  - 8 input triggers from analyser probe
- 2 output triggers (TTL levels)



Rev. 2.0

July 1998

1/4

This is preliminary information on a new product in development or undergoing evaluation. Details are subject to change without notice.

### DEBUGGING FEATURES

#### Mapping

- Accessible memory segments in the emulator: all ST9+ segments (21h is reserved).
  - ROMs are emulated with on-emulator memories
  - RAMs are emulated with on-emulator memories
  - external memories are either emulated with on-emulator memories (full space ST9+) or accessed by the emulator on the application board (size of one segment only)
- Granularity of 32 bytes
- Write or non existing protection mechanism

#### Instruction Breakpoints

- Unlimited on the whole memory space
- Program stop on specified instruction fetch at C or assembler level
- Can be associated with a counter or/and a condition (loss of real time operation)

#### Register Breakpoints

- 2 register events defined by register number, register page (if any), data value with bit mask
- Read, write, or read/write access conditions
- Can be used:
  - independently to generate up to 2 breakpoints
  - to define one range of register to generate 1 breakpoint
  - combined with AND or THEN condition to generate 1 breakpoint

#### Advanced Breakpoints

- 4-level sequencer that enables to perform actions when specified event or events sequence occur
- Each level is defined by:

```
IF <NOT> (N1*Event1) logic_operator1<NOT> (Event2)
THEN action_list1
ELSIF <NOT> (N2*Event3) logic_operator2 <NOT> (Event4)
THEN action_list2
```

Logic\_operator1 or logic\_operator2 are one of AND, NAND, OR, NOR, XOR, XNOR
- Only one level is active at the same time
- 2x16-bit counters available per level (one for IF, one for ELSIF condition)
- Events can be:
  - memory events defined with address, data

value with bit mask, read, write or read/write access, opcode fetch

- external events using the 9 input triggers
- trace full information
- any combination of the above listed inputs
- Actions can be:
  - to stop the program execution
  - to output a waveform on one or both of the 2 output triggers
  - to enable, disable the trace
  - to record a snapshot in the trace
  - to go to another level
  - a set of the above actions

#### Trace

- Trace contents disassembled at source level, optionally interleaved with machine instruction and buses activities
- Recordable trace in a text file
- 64K-word trace records composed of:
  - memory segment number, address, data and control buses activities
  - 8-bit analyser probe
  - 1 trigger input
  - 30-bit timestamp extendable by overflow counting. Timestamp clock is device clock source or fixed 20 MHz reference
  - advanced breakpoint sequencer information
- Selective trace defined by events in the sequencer:
  - trace on
  - trace off
  - snapshot
  - programmable trigger position in the trace record

#### Performance Analysis

- Real-time counting of time spent between 2 specified instructions
- 48-bit wide count based on timestamp clock
- Up to 32K executions recorded for the couple of instructions
- Time analysis giving average time, minimum time, maximum time, standard deviation
- Graphically displayed results

**Freeze Peripherals when program execution is stopped**

- Individual selection to enable/disable freeze function for:
  - Watchdog Timer when used as Timer
  - Standard Timer
  - Multi Function Timer0
  - Multi Function Timer1
  - Multi Function Timer3
- Watchdog always stopped when program is stopped
- ADC always stopped when program is stopped

**Code Debug**

- Full C source level and/or assembly level debugging capabilities
- Display of source code in concurrent windows
- Disassembling capability, optionally with interleaved source lines and ST9+ instructions, with symbolic and/or hexadecimal instruction operands
- Log file feature, allowing storage and subsequent redisplay or re-use of all displayed information
- Command files capability, which may be executed at user's request or automatically interpreted at start-up

**Display and Modification Capabilities**

- All variables, taking into account the exact structure declared in the source code
- C-like expressions (display only)
- The entire ST9+ memory structure
- All ST9+ registers
- The ST9+ system registers:
  - stacks
  - working registers
  - flags
- Stack showing current function calls for each stack level and their parameter values (display only)

**Running Features**

- Execution starting from reset entry point, or from the current position in the user program
- Single stepping capabilities:
  - at source code level
  - at ST9+ machine instruction level

– entering in interrupts or not

- Function call or a macro reference may optionally be considered as a single instruction depending on the level of detail required

**Debugger Customising:**

- Last working environment restored automatically for a given application:
  - number of windows opened
  - windows position
  - breakpoint lists
- Working environment storage in personalization files capability

**Help Files**

- Full on-line help following Windows standard help facilities
- Context-sensitive accesses from any window
- Keyword search facility

**Debugger based on GDB technology**

- Line mode accessible, with access to all GDB commands

### Notes:

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics.

©1998 STMicroelectronics - All Rights Reserved.

Purchase of I<sup>2</sup>C Components by STMicroelectronics conveys a license under the Philips I<sup>2</sup>C Patent. Rights to use these components in an I<sup>2</sup>C system is granted provided that the system conforms to the I<sup>2</sup>C Standard Specification as defined by Philips.

STMicroelectronics Group of Companies

Australia - Brazil - Canada - China - France - Germany - Italy - Japan - Korea - Malaysia - Malta - Mexico - Morocco - The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.