ST7 MICROCONTROLLER TRAINING

1. INTRODUCTION
2. CORE
3. ADDRESSING MODES
4. ASSEMBLY TOOLCHAIN
5. STVD7 DEBUGGER
6. HARDWARE TOOLS
7. PERIPHERALS
8. ST-REALIZER II
9. C TOOLCHAINS
ST7 HARDWARE TOOLS

- ST7 EMULATOR
- ST7 DEVELOPMENT KIT
- PARALLEL CONNECTION
- APPLICATION BOARD
- ENGINEERING PROGRAMMING BOARD / STICK
- LCD Control
HARDWARE TOOLS OVERVIEW

• Complete support of ST7 families with:
  ➢ Dedicated emulators
  ➢ Development kit which contains in one single board:
    ✓ Low cost emulator
    ✓ EPROM Programmer (DIL packages only) + ISP/ICP connector
  ➢ Engineering Programming Boards (all packages)
  ➢ STICK: generic programming board used with daughters boards
EMU2B EMULATOR MAIN FEATURES

- Complete real time emulator up to 16MHz
- Dedicated to a specific MCUs family:
  - MDT1 for SDIP32 or SO28 package
  - MDT2 for SDIP 42/56 or QFP 44/64 package
  - MDT3 for SDIP 42/56 or QFP 44/64 package and CAN cell
  - MDT4 for USB or DAC with PWM cell
  - MDT5 for Motor Control cell
  - MDT6 for Op Amp cell
  - MDT7 for Card Reader
- Powerful logic analyser
- External logic probe
- Extended range of breakpoints
- Trigger outputs
- External clock input
EMU3 EMULATOR MAIN FEATURES

- CPU frequency: 32 KHz up to 24 MHz
- Application power supply follower: 0V to 5.5V
- EPLD or emuchip based
- Dedicated to a specific MCUs family:
  - MDT10 for SDIP32 or SO28 package
  - MDT20J/M for SDIP 42/56 or QFP 44/64/80
- BEM (Bus Event Machine)
- Performance Analysis
- Extended range of breakpoints
- Trigger input/outputs
- Trace 256 K cycles with records of 112 bits
- External clock input
- Timestamp in Trace window
- R/W on the fly
DEBUGGING WITH THE EMULATOR

Debug level

Bus Event Machine
- Advanced breakpoints
- Trace filtering

Real Time Trace analyser
- Record complex chained internal / external events
- Break Capability

Hardware Events/Data breakpoints
- Trigger output
- Filter trace
- Stop on R/W access on a variable

Software Breakpoints
- Break on a source code line execution
- Always stop execution

ST
BREAKPOINTS OVERVIEW

- **Software Breakpoints:**
  - Stop on a code line (on fetch), in the source or the disassembly window

- **Data Breakpoints:**
  - Stop at R/W accesses of a variable

- **Hardware Events:**
  - Record when the HW condition is met upon:
    - Event ON (@ event begins)
    - Event OFF (@ event ends)
    - Event HIT (1 particular @ accessed)
    - Enables to generate an output trigger signal
REAL TIME TRACE RECORDER

• Display for each CPU cycle address and data bus value, memory access type,…

• **Trace** feature provides a record of application execution with filtering, display controls, timestamp, save to file

• Trace buffer depth :256k with records of 112 bits (EMU3)
**EMU3 Read/write on the fly**

**Change a value**
- Double click in **Value** column
- Type the new value (eg. 1)
- Press **Enter**

*In this case, entering 1 for **enableBug** sets off a countdown.*

“5, 4, 3, 2…”

When **bugCountDown** reaches 0, **nbOfTransitions** is overwritten with a false value.

**Read/write on the fly** allows you to watch the evolution and enter new values for variables during the execution of your application:
- Window background is yellow
- Values appear in **bold** as they change
EMU3 Advanced breakpoints

• What are they for?
  ➢ They permit intricate debugging by allowing user to set complex breakpoints and to control trace and triggers on predefined events.

• Main features:
  ➢ The “Advanced breakpoints” feature is a configurable 4-level state machine which is executed at micro controller speed.
  ➢ Each level can specify up to 4 events, hence a user can choose to follow a series of up to 16 logical events.
  ➢ Actions may be performed upon the occurrence of the defined event or sequence of events.
  ➢ Any user configuration can be saved as a *.bem file and be re-used for any other workspace.
EMU3 Advanced breakpoints: Definition

(1/3)

• Events can be defined using any of the following parameters:
  ➢ a specific address or range of addresses,
  ➢ a specific data value with bit mask,
  ➢ a read, write or read/write access,
  ➢ an opcode fetch,
  ➢ external event(s) monitored using one or all of the nine input triggers, trace full information,
  ➢ a DMA memory access,
  ➢ an interruption,
  ➢ a stack operation access.
EMU3 Advanced breakpoints
Definition (2/3)

- Actions can be defined as:
  - a break in the execution of the program (i.e. a breakpoint),
  - the enabling, or disabling of trace recording,
  - the recording of a snapshot in the trace,
  - continuing to another level of conditions, defined by another event or series of events,
  - the outputting of a waveform to one or both of the two output triggers,
  - a set of the above actions.
EMU3 Advanced breakpoints

Definition (3/3)

• Levels can be defined as:
  ➢ The level defines the relationship between these events (i.e. specific, user-defined conditions) and the actions to be carried out by the debugger. A counter may also be set.
  ➢ The definition of each level has the following structure:

    IF [NOT] <[counter] event1> [<combination> [NOT] <event2>]
    THEN actions

    ELSE IF [NOT] <[counter] event3>
    [<combination> [NOT] <event4>]
    THEN actions

Combination operators: AND, NAND, OR, XOR, NOR, XNOR
Advanced Breakpoints feature allows you to trigger actions based on complex events that you define:

- Four levels
- Events (on data, address, signal, trigger…)
- Actions (break, output trigger, filter trace…)
- Synoptic view for graphic representation of breakpoint logic
- and Save configuration to file
EMU3 Advanced breakpoints

Example

• Break on memory access
  ➢ To set a breakpoint on memory access (either a read or a write) within a specific range of addresses. This sort of breakpoint is very useful if you wish to ensure that certain memory zones are never read or written.
  ➢ To set a breakpoint on the reading or writing of a particular data value to a particular memory address or range of memory addresses.
  ➢ This Break on <DATA> are useful after function call

<table>
<thead>
<tr>
<th>START with TRACE</th>
<th>ON or OFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEVEL 1:</td>
<td>IF (opcode fetch at main.c*57) THEN Goto Level 2;</td>
</tr>
<tr>
<td>LEVEL 2:</td>
<td>IF (access at memory location [0x40] with data=0x05) THEN break;</td>
</tr>
<tr>
<td>LEVEL 3-4:</td>
<td>(empty)</td>
</tr>
</tbody>
</table>
**EMU3 Advanced breakpoints**

**Example**

- Synchronize an external device
  - Sometimes it is useful to send a pulsed signal from one of the output triggers to synchronize an external device upon the occurrence of a particular event. For example, say that you wish to synchronize an oscilloscope upon the occurrence of an entry into an interrupt loop (or a specific task in a function).

<table>
<thead>
<tr>
<th>START with TRACE</th>
<th>ON or OFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEVEL 1:</td>
<td>\textbf{IF} \ (opcode fetch in task,,IT,,) \textbf{THEN} \ (Trigger 1 Pulse);</td>
</tr>
<tr>
<td>LEVEL 3-4:</td>
<td>(empty)</td>
</tr>
</tbody>
</table>
EMU3 Advanced breakpoints

Example

- Synchronize trace with an external events

  Sometimes it is useful to observe data or code after an external event (i.e.: a signal change on the application board). For example, you wish to measure the latency time between a port (input) change and the function treatment.

<table>
<thead>
<tr>
<th>START with TRACE</th>
<th>ON or OFF</th>
</tr>
</thead>
</table>
| LEVEL 1:         | IF (Trigger in =1)  
                  THEN (TRACE <Snapshot>  
                  Goto LEVEL2); |
| LEVEL 2:         | IF (access at memory location [0x40] with data=0x05)  
                  THEN TRACE <Snapshot>  break; |
| LEVEL 3-4:       | (empty) |
EMU3 Advanced breakpoints

• Saving and loading advanced breakpoints
  ➢ You can store the definition for your advanced breakpoint to a file (*.bem), which can be reloaded by any STVD7 project, so long as you are using the EMU3 emulator to debug the application. This will allow you to save time if you have to switch between advanced breakpoints while debugging your application.

• Help for advanced breakpoints on STVD7
  ➢ Open the Search window by selecting, from the main menu, Help > Search, or by clicking on (Search in Help, or Help Home Page icons) in the Emulator toolbar, and then type “bem” in the keyword field of the search folder.
EMU3 Advanced breakpoints

• Timing limitations
  ➢ Advanced breakpoints take time to be evaluated and triggered.
  ➢ Trace timestamp counter runs at 20Mhz internal clock or user-specified external clock and hence has a time base which differs from “Advanced Breakpoints”.
  ➢ Trace timestamp counter is a 30-bit counter (107 374 182 350 ns) whose reset means a change in reference point for timing calculations.
Performance analysis

**Performance Analysis** feature shows you how much time the ST7 spends processing a section of your application code that you specify.

The graphic interface provides a report of statistics and a graphical representation of the statistics.
Emulator update

- Emulators contain firmware that may be updated
- When the EMU3 is initialized
  - STVD7 checks the firmware versions
  - If needed, the user is prompted for automatic update
Emulator/Emulator discrepancies menu displays the differences between the emulation system and the target MCU as they are different for each target and emulator type (emuchip or ELPD).

MDT10 emulator is EPLD based so parametrics are different for instance (ADC conversion time, RC calibration not feasible as emulated by a fixed frequency from an oscillator, analog comparator not emulated, …)
**EPB: Engineering Programming Board OVERVIEW**

- **Hardware features:**
  - Device Read, Verify, Program for:
    - A given area (FLASH, EPROM, EEPROM, OPTION BYTE)
    - A range in a specific area
    - All areas (FLASH, EPROM, EEPROM, OPTION BYTE) at a time
  - EPROM Blank Check
  - ISP or ICP connector on board (if feature supported by the device)

- **Software features:** STVP7
  - Epromer configuration for hardware and device
  - Load/Save binary files in FLASH, EPROM, EEPROM or OPTION area
  - Dump and modify the FLASH, EPROM, EEPROM memory image:
    - Read from the device
    - Loaded from the binary file
  - Project management allows the user to configure the epromer and load a file in each area at a time
STICK OVERVIEW

• Hardware features:
  - Replace the EPB: generic board to be used with dedicated daughter boards depending on the chosen device
  - Parallel interface with the PC
  - Daughter board can be plugged on the STICK (no need then to supply it) -> called Socket Board
  - STICK can be also used with any board having a ICP connector
  - 12V power supply

• Software features: STVP7
  - Epromer configuration for hardware and device
  - Load/Save binary files in FLASH, EPROM, EEPROM or OPTION area
  - Dump and modify the FLASH, EPROM, EEPROM memory image:
    - Read from the device
    - Loaded from the binary file
  - Project management allows the user to configure the epromer and load a file in each area at a time
ICP & Option Bytes Management

• 2 different ICP modes (for option bytes clock):
  1) « ICP » or « ICP OPT Enable » Programming mode:
     ➢ Uses Option Bytes already programmed in the ST7
     ➢ Selected clock source must be provided to the application
  2) « ICP OPT Disable » Programming mode:
     ➢ Ignore Option Bytes programmed on ST7 device
     ➢ Always External Clock source in this mode: Need OSC pin to be connected from the tool HE10 connector (pin 9) to the ST7 Clock input
STVP7 CONFIGURATION
SCREEN DISPLAY

- Title bar
- Main menu bar
- Toolbar
- Device status area
- Display area
- Flash sector area
- Tab menu
- Output window
- Status bar
SCREEN DISPLAY

Option Byte Configuration

Information Area
PROJECT FEATURES

• A project is a file which may be created, loaded, edited and executed (Program all)

• Loading a project allows the user to:
  - Set the eepromer configuration (Hardware, Device and Port)
  - Automatically load a file in each area (Prog mem, EEPROM...)
  - Program all the device areas at a time

• Define programming options:
  - Blank Check
  - Verify after programming

• Implement serial numbering

• Protect sectors from edition
DEVELOPMENT KIT OVERVIEW (DVP3)

- MDT10 supports Lite family + ST72F26x
- MDT20M/J supports ST72F521/32x family
- MDT25 support ST72F561 family
- Parallel or USB connection to the PC
- Programming features:
  - Under STVP7 control
  - ICP connector on board
- Emulation features:
  - Under STVD7 control (hardware test available)
  - Full real time emulation (data breakpoints,...)
  - 16 MHz on-board clock or external clock possibility
  - Extended breakpoints range
  - Output/Input Trigger, 3 analyzer probe input pins
  - Timer frozen capability
  - 512x32 bits of Trace with filtering capabilities
  - Stack overflow/underflow detection
DVP3 TRIGGER MODES

- Hardware Events mode
  - Trigger output signal (external resource)
  - Trace filtering
- Trigin mode
  - Break on an external input signal (rising edge)
- Trace overflow break mode
  - Break when trace full
- Trigger output:
  - output level changes depending on events (EVT-ON, EVT-OFF, EVT-HIT)
DEVELOPMENT KIT LIMITATIONS

- Application supply is 3.3V or 5V
  - No application power supply follower
  - ADC referenced to Emulator VDD (not Appli VDD)
- No Advance break, no Time stamp, no Perf analysis
- Clock frequency: 2MHz ≥ Fcpu ≥ 8MHz
- Discrepancies window available in STVD7 menu as limitations can be different depending on the DVP3 and the devices:
<table>
<thead>
<tr>
<th>PRODUCT FAMILY</th>
<th>EMULATOR</th>
<th>DVP</th>
<th>EPB</th>
<th>STICK</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST72260G1</td>
<td>ST7MDT10-EMU3</td>
<td>ST7MDT10-DVP</td>
<td>ST7MDT10-EPB</td>
<td>ST7-STICK</td>
</tr>
<tr>
<td>ST72262G1/2</td>
<td>ST7MDT10-EMU3</td>
<td>ST7MDT10-DVP</td>
<td>ST7MDT10-EPB</td>
<td>ST7SB10-SU0</td>
</tr>
<tr>
<td>ST72264G1/2</td>
<td>ST7MDT10-EMU3</td>
<td>ST7MDT10-DVP</td>
<td>ST7MDT10-EPB</td>
<td>ST7SB10-26x</td>
</tr>
<tr>
<td>ST7Litex</td>
<td>ST7MDT10-EMU3</td>
<td>ST7MDT10-DVP</td>
<td>ST7MDT10-EPB</td>
<td>ST7SB10-123</td>
</tr>
<tr>
<td>ST7232xMx/Rx/Jx</td>
<td>ST7MDT20M/J-EMU3</td>
<td>ST7MDT20M/J-DVP</td>
<td>ST7MDT20-EPB</td>
<td>ST7-STICK</td>
</tr>
<tr>
<td>ST72521Mx/Rx</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-EPB</td>
<td>ST7SB20J/M</td>
</tr>
<tr>
<td>ST72312R4</td>
<td>ST7MDT20M/J-EMU3</td>
<td>ST7MDT20M/J-DVP</td>
<td>ST7MDT20-EPB</td>
<td>ST7MDT2-EPB</td>
</tr>
<tr>
<td>ST723532R4</td>
<td>ST7MDT20M/J-EMU3</td>
<td>ST7MDT20M/J-DVP</td>
<td>ST7MDT20-EPB</td>
<td>OR</td>
</tr>
<tr>
<td>ST72311R6/7/9</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-EPB</td>
<td>ST7MDT2-EPB</td>
</tr>
<tr>
<td>ST72331R6/7/9</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-EPB</td>
<td>ST7MDT2-EPB</td>
</tr>
<tr>
<td>ST72272K2</td>
<td>ST7MDT4-EMU2</td>
<td>ST7MDT4-DVP</td>
<td>ST7MDT4-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72372J4</td>
<td>ST7MDT4-EMU2</td>
<td>ST7MDT4-DVP</td>
<td>ST7MDT4-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72371N4</td>
<td>ST7MDT5-EMU2B</td>
<td>ST7MDT5-DVP</td>
<td>ST7MDT5-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72141K2</td>
<td>ST7MDT5-EMU2B</td>
<td>ST7MDT5-DVP</td>
<td>ST7MDT5-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72104G1/2</td>
<td>ST7MDT1-EMU2B</td>
<td>ST7MDT1-DVP</td>
<td>ST7MDT1-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72216G1</td>
<td>ST7MDT1-EMU2B</td>
<td>ST7MDT1-DVP</td>
<td>ST7MDT1-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72215G2</td>
<td>ST7MDT1-EMU2B</td>
<td>ST7MDT1-DVP</td>
<td>ST7MDT1-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72254G1/2</td>
<td>ST7MDT1-EMU2B</td>
<td>ST7MDT1-DVP</td>
<td>ST7MDT1-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72124J2/4</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-DVP</td>
<td>ST7MDT2-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72313J2/4</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-DVP</td>
<td>ST7MDT2-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72334J2/4</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-DVP</td>
<td>ST7MDT2-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72314N2/4</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-DVP</td>
<td>ST7MDT2-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72334N2/4</td>
<td>ST7MDT2-EMU2B</td>
<td>ST7MDT2-DVP</td>
<td>ST7MDT2-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72411R1</td>
<td>ST7MDT7-EMU2B</td>
<td>ST7MDT7-EPB</td>
<td>ST7MDT7-EPB</td>
<td></td>
</tr>
<tr>
<td>ST72171K2</td>
<td>ST7MDT6-EMU2B</td>
<td>ST7MDT6-EPB</td>
<td>ST7MDT6-EPB</td>
<td></td>
</tr>
</tbody>
</table>

Tools salestypes added at the end of the datasheets!
A Complete Set of Tools
IN HOUSE AND THIRD PARTIES

Emulator
Full featured emulator

DVP series
Entry-level emulator

In-Circuit Debugging
Low-cost real-time emulation

Starter kit
Start a new design quickly

Evaluation Board
Demonstration and evaluation of ST7 capabilities

Tools keys benefits:
- Productivity improvement
- Time to market

Graphics-oriented tool
Software to create an application without writing assembly code

http://mcu.st.com
## 8-bit & 16-bit 3rd Party Partners

<table>
<thead>
<tr>
<th>3SOFT</th>
<th>DataMan</th>
<th>8BIT SYSTEMS</th>
<th>Microsoft Windows Embedded Partner</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTUM Solutions</td>
<td>EE Tools</td>
<td>ELMEC</td>
<td>mapp Embedded System</td>
</tr>
<tr>
<td>ADVANTECH</td>
<td>dmWare</td>
<td>LÄUTERBACH</td>
<td>NOHAI</td>
</tr>
<tr>
<td>A</td>
<td>KEIL SOFTWARE</td>
<td>LEIA ELECTRONIC CO., LTD.</td>
<td>PHYTEC</td>
</tr>
<tr>
<td>BP Microsystems</td>
<td>HI-LO SYSTEMS</td>
<td>Kanda;shop</td>
<td>Phyton</td>
</tr>
<tr>
<td>CodeScape</td>
<td>hitex</td>
<td>Linux</td>
<td>pls</td>
</tr>
<tr>
<td>CONTEC DATENSYSTEME</td>
<td>hitex SYSTEMS</td>
<td>LINEO</td>
<td>raks</td>
</tr>
<tr>
<td>C</td>
<td>GENERAL SOFTWARE</td>
<td>LinePro</td>
<td>Raisonance</td>
</tr>
<tr>
<td>S</td>
<td>hse ELECTRONIC</td>
<td>metrowerk ver Software</td>
<td>RK-SYSTEM</td>
</tr>
<tr>
<td>COSMIC software</td>
<td>I-Logix</td>
<td>INFORM</td>
<td>SEGGER</td>
</tr>
<tr>
<td>Data</td>
<td></td>
<td></td>
<td>XELTEK</td>
</tr>
</tbody>
</table>

---

- Actum Solutions
- Advanced Micro Tools
- Advantech
- A
- Advanced Micro Tools
- BP Microsystems
- CodeScape
- CONTEC DATENSYSTEME
- COSMIC software
- Data
- Advantech
- Advanced Micro Tools
- A
- Advanced Micro Tools
- BP Microsystems
- CodeScape
- CONTEC DATENSYSTEME
- COSMIC software
- Data
Large range of high-end Emulators

ST EMU3 for all ST7 Flash

ST EMU2 for all ST7 Cxxx and Txxx

iSystem: Large range of High-end emulator iCx000

Hitex: Large range of high-end emulator AX-ST7, MX-ST7
ST7 3rd parties Emulators

**MODULAR Emulator**
Once the first emulator invested, additional target is from $1500 to $2000.

**GENERIC Emulators**
for 8 bits microcontrollers (ST and ST competition)

**iSystem**
- **iC1000** MONOLITHIC EMULATOR
- **iC2000** MODULAR EMULATOR

**Hitex**
- Tree versions with different prices & levels of functionalities:
  - **MX-ST7-L** debug of simple applications
  - **MX-ST7**: full features to detect complex bugs
  - **AX-ST7**: universal high-end in-circuit emulator

**Hitex** from $4000

**iSystem** from $5000
High performance and maximum flexibility:
Big emulation RAM, Big trace buffer, Complex breakpoints, Real-time Watch...

Powerful features:
bus width, high speed communication...

Powerful IDE:
iSystem IDE: 30 days free trial available on ST web site and in ST CD.

Non-intrusive

Advanced programmable logic technology
ST7 In Circuit Debugging
RAISONANCE

✓ **Rlink-ST**: USB/JTAG dongle for ST7, STR7 and μPSD
  - Low cost tools for In Circuit Debugging
  - **Modular demo boards** system, featuring In Circuit Debugging and In Circuit Programming
  - 1 **Generic** demo board, with led, potentiometer….(REVA board)
  - **Daughter** boards for selected MCU (card).

✓ 3 **Mass Market** starter kits (ST7):
  - **ST7LITE-SK/RAIS** (ST7FLITE3x, ST7FLITE0x, ST72F264x) @ $149
  - **ST732X-SK/RAIS** (ST72F321xBJ6) @ $149
Flasher: Manufacturing In-Circuit programmer

Flasher

- In-circuit programming tool
  Serial programming supported
- Engineering or Manufacturing env.:
  can work in stand-alone mode without the need of a PC.
- Can be connected to an Automatic Tester.
- Read, Erase, Program and Verify, Read back.
- High speed programming:
  app. 12 sec for 60 kB in serial mode with 16 MHz target for programming and verification

Available for ST72Fxxx General Purpose devices

$477

www.segger.com
ST7 SOFTWARE TOOLS

ST designed tools & wide 3rd party support

- TINY + Real Time OS
- REALIZER Graphical design
- C Compilers
- C Compilers
- ACTUM designed
- Genuine ST tools based
- 3rd party designed including debugger
- Several vendors: RAISONANCE, COSMIC
- emWARE
- Isystem: IDE for ST emulators
- Kanda: IDE with Starter-kit + Application Builder
- CMX
- IDE’s
- Assembler Linker Debugger Simulator
- Complete ST designed chain
- Compatible with ST real time emulators
- STVD7 (IDE)
TOOLS and SUPPORT - ST7 LIBRARY

OVERVIEW

• Benefits
  - Hides the Hardware Layer
  - Standardizes the Source code
  - Compact
  - Help getting started
  - Documentation makes it user friendly.
  - Free of charge (www.st.com/mcu)

• Support
  - 14 standard peripherals
  - Device
    - ST72F561
    - ST72F521
    - ST72F321
    - ST7LITE0
  - Compiler
    - Cosmic
    - Raisonance (soon)
    - Metrowerks
**Real time Operating System**

- Truly pre-emptive, multi-tasking operating system.
- Powerful functions: management for Task, Timer, Memory, Resource, Event, System, Message, Queue, Uart.

**OS Segger**

- Easy to use, Easily portable
- One of the smallest and fastest kernels
- No royalties, affordable

**OS CMX**

- Real time Operating System
- Truly pre-emptive, multi-tasking operating system.
- Powerful functions: management for Task, Timer, Memory, Resource, Event, System, Message, Queue, Uart.

**μC/OS-II Micrium**

- Real-Time Kernel
- Highly portable, ROMable, very scalable, preemptive real-time, multitasking kernel (RTOS)
- Can manage up to 63 application tasks.
Other ST7 Software
3rd Party Tools 2/2

Real-time Architecture configuration tool
Real-time Timing Analysis tool

- SSX5 Real-Time Operating System
- Architecture configuration tool
- Timing Analysis tool
- Smallest, fastest OSEK OS
- RTA supports all variants of the ST7
- MISRA C Compliance of all target code

Vector, IXXAT
- CAN
- LIN
- OSEK
3rd party C COMPILERS
Full interoperability with ST tools

ST IDE: STVD7

Cosmic / Raisonance IDE

Cosmic / Raisonance Compilers

ST Debugger: STVD7

Cosmic / Raisonance Debugger

Simulator

In-Circuit Debugging

Entry-level Emulator DVP3

High-end Emulator EMU3
C Tool Chain

*To decrease development cost and time to market!*
*To increase Design-reuse!*

**FREE version**

- Compiler
- Limitation (with registration on Cosmic site):
  - 16K bytes of C code
  - 16K bytes of Assembly code

**Time limited Full version**

- 30 days of FREE evaluation to be ordered directly to the 3rd party.
Cosmic C Tool Chain

IDEA
- Complete & Easy to use
- Intuitive GUI based on the popular Codewright Windows editor/project manager.

C Cross
- Optimized ANSI-C Compiler
- Reentrant stack model and static models
- Smart Linker optimizes and overlays RAM space
- In-Line Assembly
- IEEE floating point libraries

For more information: www.cosmicsoftware.com

3 licenses:
- Full featured
- Full limited 1 month
- 16Kb limited (FREE)

Full featured source-level debugger
- ZAM SIM: Debugging environment using ST7 core simulation
- ZAP ICE: Deb. env. for ST emulators (EMU2 & EMU3)
- ZAP Monitor: In Circuit Debugging with ST7FLITE2 starter kit

For more information: www.cosmicsoftware.com
RAISONANCE C Toolchain

- Soon available (Q2 2006)
- Same range of price and performances as COSMIC
- ST7 / STM75 C Compiler
- www.raisonance.com
Graphics-oriented tool
Simply draw a diagram of the application using a library of symbols that represent functions.

ST Realizer II

- Project Manager under IDE
  - Enhanced libraries
  - Symbols for accessing all peripheral control registers
  - HW interrupts
  - Real time management

- Automatic generation of code
  - source assembly,
  - simulation files
  - machine code (S19, HEX)

- Simulation with graphic i/f
  - Pin level simulation

Free demo on ST CDROM

Now supports:
- C254 / C334
- LITE0x
- F264

Simply draw a diagram of the application using a library of symbols that represent functions.

STREALIZER-II

$150
## Graphics-oriented tool

<table>
<thead>
<tr>
<th></th>
<th>ST REALIZER II</th>
<th>REALIZER Bronze</th>
<th>REALIZER Silver</th>
<th>REALIZER Gold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available from</td>
<td>ST</td>
<td>Actum</td>
<td>Actum</td>
<td>Actum</td>
</tr>
<tr>
<td>Support</td>
<td>No</td>
<td>No</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Features</td>
<td>ST6 + ST7 equivalent to Silver version</td>
<td>Demo version</td>
<td><a href="http://www.actum.com/actum.htm">http://www.actum.com/actum.htm</a></td>
<td>Silver version + Optimization features</td>
</tr>
<tr>
<td>Prices</td>
<td>$150</td>
<td>Free demo on <a href="http://www.actum.com">www.actum.com</a></td>
<td>Approx. $800 (with ACTUM support)</td>
<td>Approx. $1500</td>
</tr>
</tbody>
</table>