MaRTE OS: Practical view

Daniel Sangorrín
Miguel Tellería

This talk is sponsored by the FRESCOR european FP6 project. More information available at http://www.frescor.org
Where can I get it?

MaRTE OS website: http://marte.unican.es

- news RSS2.0 feed
- documentation
- downloads
- finished projects
- on-going work
- forum
The documentation

MaRTE OS User's Guide
README, INSTALL
Tutorials
- Boot process
- Misc utilities
- QEMU
Papers and presentations
Other sources:
- FRESCOR deliverables
- Degree projects reports
- Forum
Installation, compilation, debugging

Requirements:

- **Host Computer**: GNU/Linux with compilation environment and AdaCore GNAT compiler

- **Target Computer (only x86)**:
  - Ethernet card for remote loading (optional)
  - Serial port for remote debugging (optional)

MaRTE OS scripts:

- mininstall, mkrtsmarteuc, mkmarte

Application scripts:

- mgnatmake, mgcc, mg++

Debugging (gdb, linux_arch, qemu, serial port)
Development environment
Example 1: hello world

mgnatmake -g hello_world.adb

with Text_IO;

procedure Hello_World is

begin

    Text_IO.New_Line;
    Text_IO.Put_Line ("Hello FOSDEM");
    Text_IO.Put ("I'm an Ada program running on MaRTE OS.");
    Text_IO.New_Line (2);

end Hello_World;
Example 2: concurrency

Basic test of the Ceiling Locking protocol in protected objects:

- A high priority task (T_High) displays periodically a message and suspends.

- A low priority task (T_Low) is active all the time and periodically executes a protected operation.

- T_High's messages stop while T_Low executes the protected operation since the PO's ceiling is above T_High's priority.
Example 3: Hardware management

Parallel port

- Handle Interrupt + Access to data register

MHI.Associate (PP_IRQ, PP_IRQ_Handler'Unrestricted_Access);

Outb_P (PP_CONTROL_REG, PP_IRQ_ENABLE);

MHI.Unlock (PP_IRQ);

loop

   PHI.Wait (Intr, Handler);

   Outb_P (PP_DATA_REG, 2#00_001_001#); delay (0.5);

   Outb_P (PP_DATA_REG, 2#00_010_010#); delay (0.5);

end loop;
Example 4: RT-EP

A Real Time Communication Protocol over Standard Ethernet

- Ada95 and C Interface
- Sporadic servers, Distributed Mutexes, Reliable Multicast
- Bandwidth reservation
- Polyorb CORBA and DSA (also Rt-CCM Thales)

Some features of Ada that are used in RT-EP implementation:

- Tasking, Protected Objects, Synchronous Task control
- Package hierarchy
- Exceptions
- Generics
- C Interfaces, FLORIST (Posix Ada bindings)
Example 4: RT-EP

a) Priority arbitration

b) Transmission
Example 4: RT-EP

[Diagram showing the states and transitions of the RT-EP algorithm]

- **Offline**
  - [NOT isTokenMaster]
  - Send Initial Token

- **Idle**
  - Message Reception / Recv Info; Send Initial Token
  - Transmirt Token Reception / Send Info
  - Token Reception [isTokenMaster AND NOT isAllowedToSend]
    - Send Info

- **Error Check**
  - ACK Reception
  - Token Reception [isTokenMaster AND isAllowedToSend]
    - Send Info
  - Token Reception [NOT isTokenMaster]
    - Send Token

- **Delay**
  - TimeOut [isMaxNumRetries AND NOT (isTokenMasterFailure AND NOT isTokenMasterAllowedToSend AND LastAction = Send Token)] / Reconfigure Ring; Send Initial Token

- **Reconfigure Ring**
  - Send Permission

- **Retry Last Action**
Example 4: RT-EP

Server.Tx_ Queues
S1  S2  S3
packet  packet  packet

Fixed Priorities
Tx  Queue
packet  packet  packet

Servers_Prio_Queue
S3  S1  S2

Rx  Queues
Chan 1  packet  +  packet  +  packet
Chan 2  packet  +
Chan 3  packet  +

/dev/eth0

RT-EP
MAIN TASK

Mutex  operations

Packet  Received

Highest  Priority
Example 4: RT-EP

<table>
<thead>
<tr>
<th>No.</th>
<th>Time</th>
<th>Source</th>
<th>Destination</th>
<th>Protocol</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>3597</td>
<td>9:469888</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_INIT_REQUEST Dest: 2</td>
<td></td>
</tr>
<tr>
<td>3598</td>
<td>9:469770</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_INIT_ACK Dest: 1</td>
<td></td>
</tr>
<tr>
<td>3599</td>
<td>9:469990</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 2 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3600</td>
<td>9:470031</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 1 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3601</td>
<td>9:470147</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 2 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3602</td>
<td>9:470252</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 1 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3603</td>
<td>9:470352</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 2 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3604</td>
<td>9:470728</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_INFO_PACKET Dest: 1</td>
<td></td>
</tr>
<tr>
<td>3605</td>
<td>9:470871</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 2 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3606</td>
<td>9:471080</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 1 No Multicast Message</td>
<td></td>
</tr>
<tr>
<td>3607</td>
<td>9:471088</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>rtep</td>
<td>RTCP_REGULAR_TOKEN Dest: 2 No Multicast Message</td>
<td></td>
</tr>
</tbody>
</table>

Real-Time Ethernet Protocol

Dest Station: 0x0000
Type: TX_GRANT_TOKEN (0x50)
Prio: 0x09
Packet Number: 0x0000
Master ID: 0x0000

0000 ff ff ff ff ff ff 00 00 00 00 00 00 00 84 04 ee aa a0 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00
0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

File: '/tmp/ether0000rQUI5k' 6223 KB 00:00:16
Packets: 47765 Displayed: 44159 Marked: 0 Dropped: 925
**Example 4: RT-EP**

**Send Packet time (DRTT/4) min size 100 us delay**

<table>
<thead>
<tr>
<th>microseconds</th>
<th>number of samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>587</td>
<td>5</td>
</tr>
<tr>
<td>588</td>
<td>5</td>
</tr>
<tr>
<td>589</td>
<td>5</td>
</tr>
<tr>
<td>590</td>
<td>1</td>
</tr>
<tr>
<td>591</td>
<td>1</td>
</tr>
<tr>
<td>592</td>
<td>0</td>
</tr>
<tr>
<td>593</td>
<td>0</td>
</tr>
<tr>
<td>594</td>
<td>0</td>
</tr>
</tbody>
</table>

**Send Packet time (DRTT/4) Max Size with 100us delay**

<table>
<thead>
<tr>
<th>microseconds</th>
<th>number of samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>854</td>
<td>5</td>
</tr>
<tr>
<td>855</td>
<td>13</td>
</tr>
<tr>
<td>856</td>
<td>9</td>
</tr>
<tr>
<td>857</td>
<td>1</td>
</tr>
<tr>
<td>858</td>
<td>1</td>
</tr>
<tr>
<td>859</td>
<td>7</td>
</tr>
<tr>
<td>860</td>
<td>2</td>
</tr>
<tr>
<td>861</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>DRTT RT-EP</strong></th>
<th><strong>Min Size Packets</strong></th>
<th><strong>Max Size packets</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>max</td>
<td>2360.48 us</td>
<td>3436.05 us</td>
</tr>
<tr>
<td>average</td>
<td>2354.85 us</td>
<td>3426.15 us</td>
</tr>
<tr>
<td>min</td>
<td>2348.88 us</td>
<td>3416.19 us</td>
</tr>
</tbody>
</table>
Communication protocols and middlewares

- PolyORB (DSA and CORBA) and GLADE ported to MaRTE OS x86 bare machine using RT-EP
- RT-WMP Real-Time Wireless protocol (UNIZAR)
- RT-EP protocol with a Bandwith reservation layer, distributed mutexes, broadcast services
- FRESCAN network protocol for CAN bus with bandwith reservation and sporadic servers
- DTM, the Distributed Transaction Manager of FRES COR
- MyCCM, implements the OMG CCM (THALES)
List of Drivers, Protocols, ...

Drivers

- CAN bus
- Wireless ralink rt61
- SVGA, BTTV, Soundblaster 16
- Mouse, Keyboard, Joystick
- Ethernet drivers (intel eepro100, rtl8139, SiS900)
- Serial port driver
- I2C protocol, compass CMPS03 driver
- IDE disk driver (CompactFlash and HD) and FAT 16 filesystem
- Advantech Data adquisition and Digital IO cards drivers (PCM3718 and others)
List of Drivers, Protocols, ...

- laser-sick-lms200
- GPS Novatel ProPak driver
- P2OS driver (compass, sonar, odometer, motors)
- Yenta Cardbus

Other features:

- MaRTE OS utilities: memory buffer driver, console switcher, logger, oscilloscope, time measurements
- MaRTE-spy
- C++ language support
- TLSF Dynamic Memory Allocator
Apps: Robot Controllers
Apps: CASEVA
Apps: AUSILGE
Apps: other projects

- Robocup robots
- Satellite antenna for high-speed train
- Platform video control
- Mobile robot
On-Going: FRESCOR
Thanks for your attention

protected body Questions is

   entry Ask_Question(Gift : out Gift_Type) is

   begin

      Gift := Live_USB_With_MaRTE_OS_And_FRESCOR;

      Put_Line ("Thanks :)");

   end Ask;

   end Questions;