Embedding a TANGO device into a digital BPM

- Global Orbit Feedback
- Digital BPM
- porting of TANGO to ARM processor
- results with the embedded device
Global Orbit Feedback

- goal: equip Elettra storage ring with a fast digital feedback system to improve orbit stability

- requirements:
  - 10 kHz sampling rate
  - sub-micron resolution
  - dump disturbances up to 150 Hz
  - suppress mains disturbances up to 300 Hz
Digital BPM

- analogue BPM electronics replaced by state-of-art digital one:

Instrumentation Technologies “Libera Electron”
Digital BPM

- meet resolution requirements
- digital processing done in FPGA
- beam position data available at 3 different rates:
  - turn-by-turn (single shot buffer reading)
  - 10 Hz
  - 10 kHz
Digital BPM

Libera is equipped with single board computer:

- Intel Xscale XA255 ARM processor
- Linux operating system
- Ethernet interface

- Device management
- Extraction of “slow” data (single-shot and 10 Hz)
TANGO device for Libera

- First developed at Soleil.
- Based on:
  - Instrumentation Technologies CSPI library
- Runs on external host
- CSPI library extracts data via TCP/IP socket and multicast over UDP
TANGO device for Libera

- network bandwidth is wasted
- “intermediate server” just for Tango device:
  deployment and troubleshooting more difficult
- full source code available for Tango
- full source code available for omniORB, CORBA library used by TANGO (C++ version)
  ➜ develop embedded version of the TANGO device
Embedding a TANGO device into a digital BPM

porting omniORB to ARM

main difficulties to overcome due to:

- building with a cross-compiler
  - the first stage of compilation builds the IDL compiler
  - the second stage uses the IDL compiler to generate stubs and skeleton classes
  - the third stage compiles core, stubs and skeleton classes
porting omniORB to ARM

IDL sources

CORBA services

arm-gcc

IDL

CORBA stubs

CORBA core

arm-gcc

omniORB

omniORB building process

Giulio Gaio, Sincrotrone Trieste S.C.p.A.
Embedding a TANGO device into a digital BPM

porting omniORB to ARM

IDL sources

CORBA services

i386 binary

ARM binary

C++ source

IDL source

omniORB building process

Giulio Gaio, Sincrotrone Trieste S.C.p.A.
Embedding a TANGO device into a digital BPM

porting omniORB to ARM

IDL sources

CORBA services

arm-gcc

IDL

CORBA stubs

CORBA core

i386 binary

ARM binary

C++ source

IDL source

copy IDL binaries from “native” build tree

omniORB building process

Giulio Gaio, Sincrotrone Trieste S.C.p.A.
Porting omniORB to ARM

Main difficulties to overcome due to:

- Mixed “endianess” of ARM
  - ARM is little-endian (like i386)
  - “Double” data on ARM is big-endian!
  - This “weird” case not handled by omniORB:
    result: mangled numbers when exchanging data between different platforms
Embedding a TANGO device into a digital BPM

porting omniORB to ARM

- endianess problem solved by patching omniORB core:
  
  Thanks to Duncan Grisby!

- last manual intervention: patch by hand (using ar) one of the omniORB library objects to overcome some deficiencies of the arm-ld linker
porting TANGO to ARM

- extra step: regenerate CORBA stubs and skeletons with patched omniORB for handling doubles correctly

- followed standard build procedure with minor Makefile adaptations

- TANGO executables must be linked as fully static executables (problem under investigation)
embedded TANGO Device

- Compilation of the Soleil designed TANGO Device for Libera for the ARM single board computer done in a very short time
- Uses the “embedded” version of the CSPI library and the newly compiled TANGO and omniORB libraries
- We had to adjust some thread management code to cope with differences of the environment and CSPI behaviour
embedded TANGO device

- no problems or bugs due to TANGO or omniORB
- deployed on tens of devices at Soleil and Elettra (96)
- easier management
- dramatic cut of network bandwidth
performances

- performances are limited only by the TCP/IP stack of the Libera single board computer

- 8 ms for reading a single BPM position

- less than 50 ms for reading the whole orbit on both planes (182 positions)
we can exploit all TANGO tools
TANGO on ARM...

As a by-product of the BPM project we now have TANGO for ARM:
http://www.elettra.trieste.it/~tango/downloads.html

Lots of gadgets are ready for learning to dance!