Space Systems Finland and the Bound-T tool

Overview

• Who we are
• What we are doing, and why
• Future plans
Who we are

• *Space Systems Finland founded in 1988:*  
  – Space-related software development and system design  
  – Currently about 30 people  
  – ESA subcontracts (Astrium, Alcatel, Alenia, ...)  
  – Nationally funded space projects  

• *SSF staff involved in WCET work:*  
  – Niklas Holsti (PhD, computer science)  
  – Thomas Långbacka (PhD, computer science)  
  – Sami Saarinen (working on MSc thesis)  
  – and others part-time.
Why we started with WCET

• **On-board flight software**
  – hard real-time, concurrent, embedded
  – very constrained by computer power
  – under 1 MIPS processors in the 1990's
  – now flying 5 - 20 MIPS

• **Example: GOMOS Instrument Application SW**
  – 100's of timing test cases, 1000's of measured times
  – 50-page Excel table to compute final results

• **ESA interest**
  – Hard-Real-Time methodology, HRT-HOOD
  – WCET tools: York U., TLD Ada, Aonix Ada
The Bound-T project

- **ADSP/TSC21020 processor**
  - Radiation-resistant DSP developed for ESA by TEMIC
  - Also used for complex control software, therefore need for HRT analysis

- **SSF develops WCET tool: Bound-T**
  - Aim at commercial product
  - ESA funds tool for 21020 and ERC32 (SPARC)
  - Other funding for other targets (Intel 8051)

- **Status**
  - 21020 and 8051 versions ready for pilot applications
  - ERC32 in progress (ready by Oct 2001)
Bound-T features

- **Reads executable binaries (compiled, linked)**
- **Automatically bounds counted loops**
- **Uses assertions for other loops and to set scenarios**
Some implementation details

- **Control-flow analysis**
  - May include more program-sequencing state than the "program counter": instruction pipeline, zero-overhead loops, ...

- **Finding and bounding loop counters**
  - Flow graphs must be reducible
  - Modeling integer calculations with Presburger Arithmetic
  - Solving with the Omega tool (Kelly, Maslow, Pugh, ...)

- **Finding the worst-case path**
  - Using Integer Linear Programming (lp_solve, Berkelaar)

- **User interface**
  - Command-line input, textual and graphical output
  - Graphics with DOT, part of GraphViz (Bell Labs)
Future plans for Bound-T

- **Marketing to space and non-space users**
- **New target processors**
  - ERC32 (SPARC V7), ADI SHARC (2106x)
  - Medium-sized mass-market embedded processors
- **Technical improvements**
  - Better aliasing (points-to) analysis
  - Cache analysis
- **Problems foreseen**
  - Convincing customers that WCET is useful & reliable
  - Popular targets are hard to model precisely (e.g. PowerPC)
  - New processors may be poorly documented for WCET
Future of WCET analysis: challenges

• *Keeping up with the processors*
  – Convince processor builders to document WCET
  – Perhaps using a common "standard" set of models

• *Keeping up with the programs to be analysed*
  – Convince programmers to make programs analysable
  – Object orientation => increased run-time dynamism
  – Increasing size and complexity

• *Combining info from source code and object code*
  – Convince compiler writers to export info for WCET
  – Static analysis tools (ASIS for Ada; ? for C/C++)
  – Define standard formats (ELF extension? XML?)