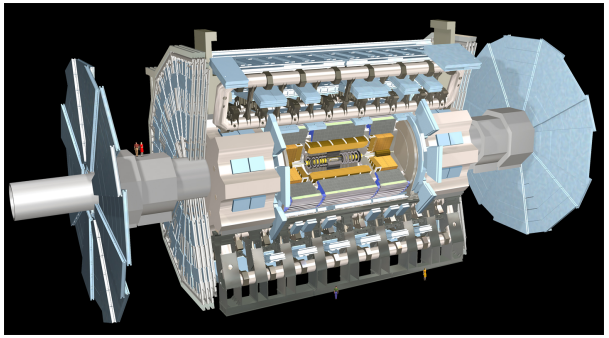
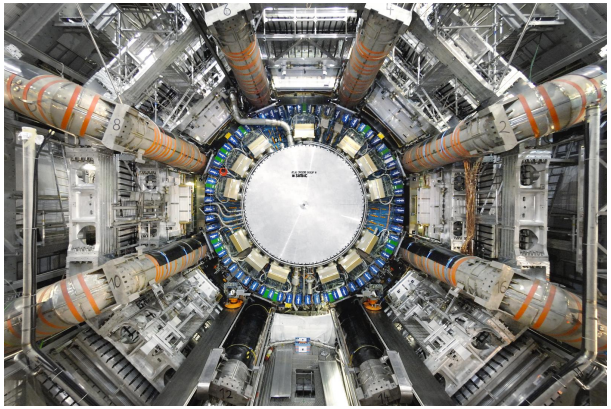




Tutorial on Programmer's Development Cycle

RCE Training Workshop

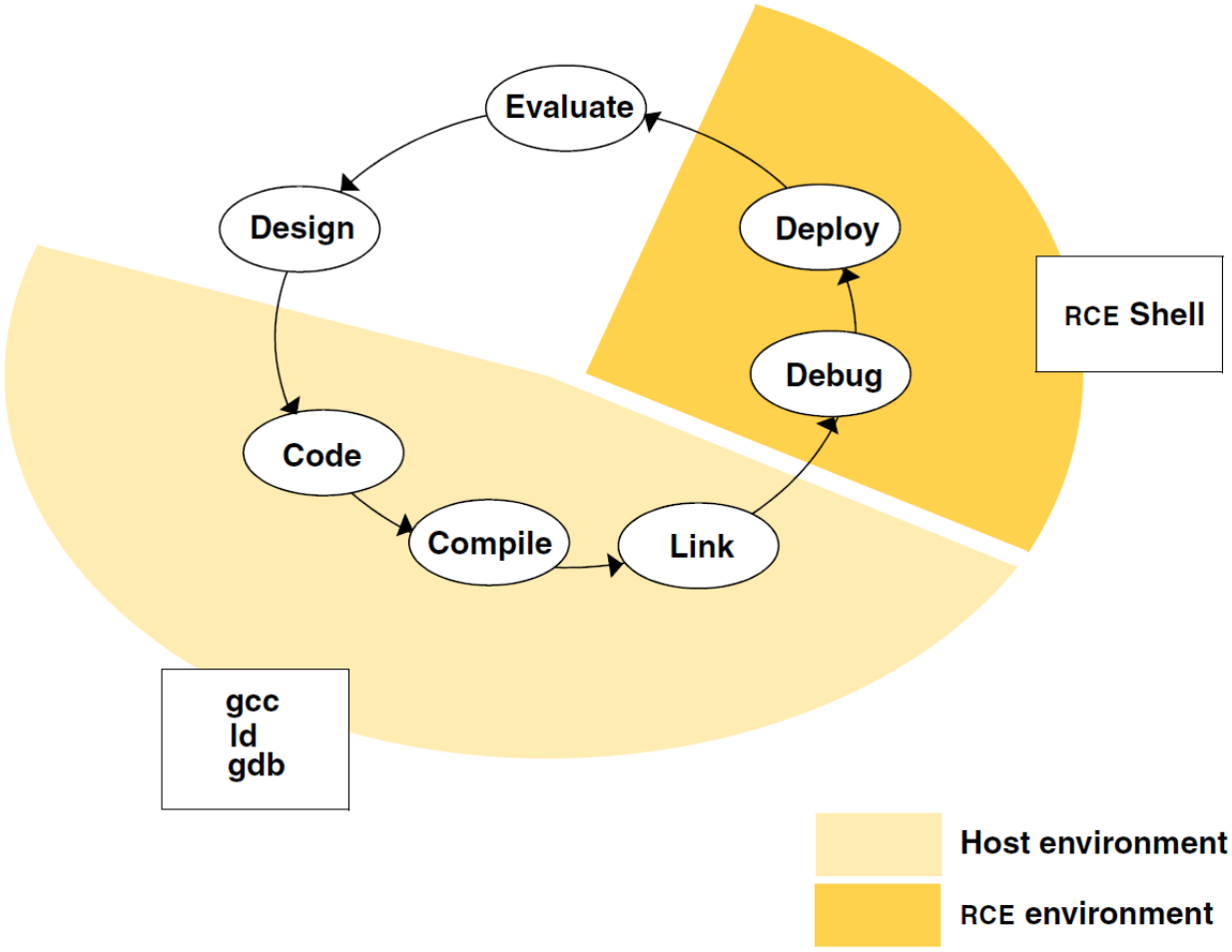
Jim Panetta, panetta@slac.stanford.edu

16 June, 2009

Overview

- Demonstrate the development cycle for a simple application that:
  - Demonstrates the basic application structure
  - Executes as an RTEMS Task
  - Illustrates the use of RTEMS directives
- Design a 'Hello World' application that:
  - Writes a series of values to the front panel of the RCE
  - Pauses between each write

RCE Development Cycle



The familiar cycle, but inherently *cross-platform*:

- Resource-intensive compilation and linking are performed under Linux
- The resulting code is run in the RCE (target) environment

Coding Hello World

- Headers for C++ and RTEMs runtime libraries [1,2]
- Local function [6]
- Application main [9]
  - Form of an RTEMS task entry point
  - Reserved name
  - "C" linkage
- Use of RTEMS directives [15]

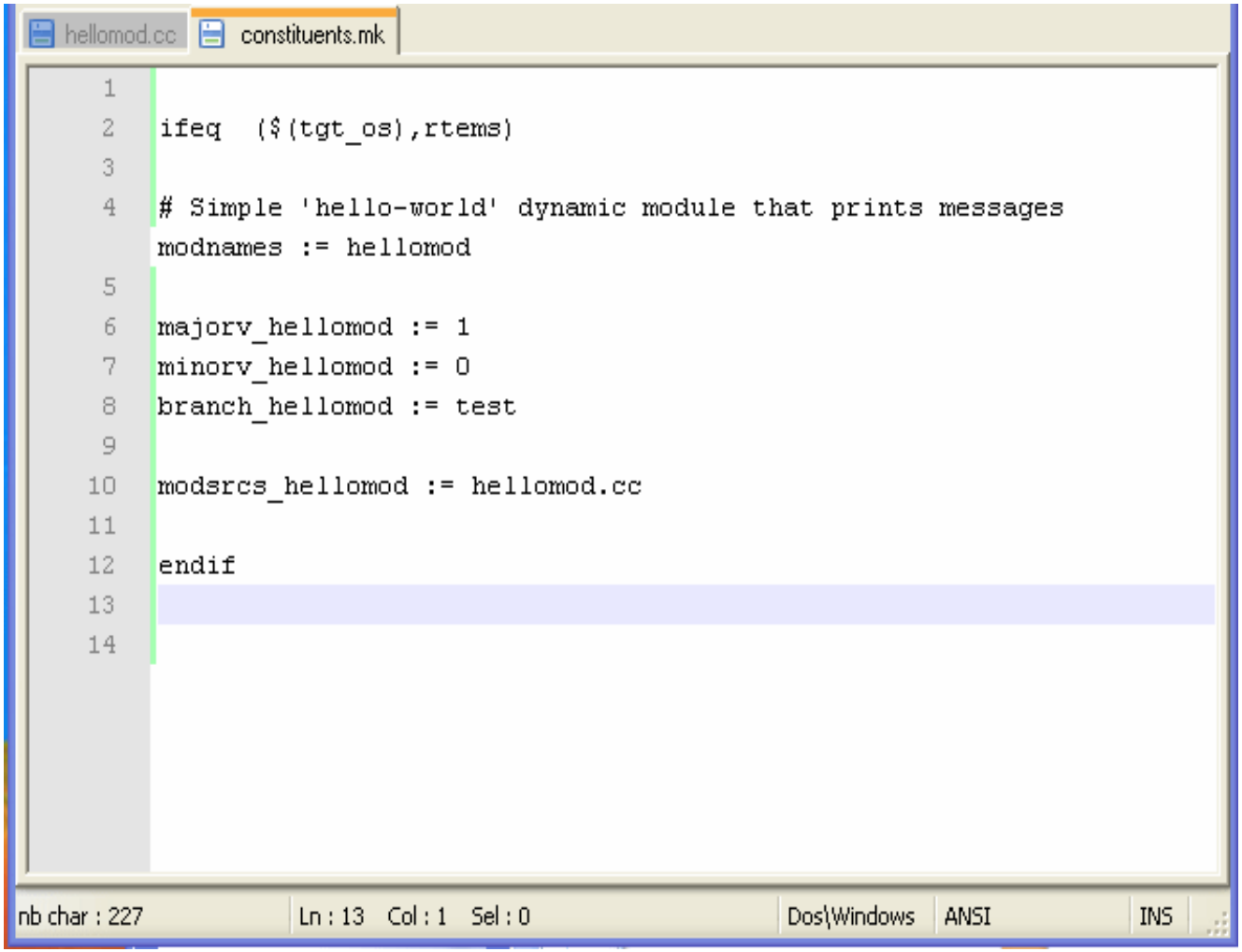
hellomod.cc

```
1  #include <cstdio>
2  #include <rtems.h>
3
4  #define TICKSPERSEC 100
5
6  static inline void write_display(unsigned v) {
7     asm volatile("mtdcr %0,%1" : : "i"(0x2f7), "r"(v)); }
8
9  extern "C" {
10    rtems_task rce_appmain(rtems_task_argument arg) {
11      int i = 10000;
12      uint32_t dispval[] = { 0xDEADBEEF, 0xDECAFBAD, 0x12345678 };
13      while (i--) {
14        write_display( dispval[i%3] );
15        rtems_task_wake_after(TICKSPERSEC);
16      }
17    }
18 }
19
```

nb char : 501 | Ln : 1 Col : 2 Sel : 0 | UNIX | ANSI | INS

Makefile and Build Sequence for Hello World

- Declare one or more module names [5]
- Give required version information for each module [6-8]
- List the source files making up the module [10]



Directory layout

- release/
  - Makefile
  - make/
    - sw/
      - flags.mk *(compiler/linker flags)*
      - project.mk *(most make code)*
  - rce/...
  - myproject/
    - Makefile
    - packages.mk
    - pkg1/
      - Makefile
      - constituents.mk
      - src1.cc ...

Deployment: Deployment Sequence

- Step 1: Run app from NFS
- Step 2: Burn app to configuration memory
- Step 3: Run app from configuration memory
- Step 4: Confirm app bootable from dipswitch

Deployment: RCE Shell Commands (1)

- mount (builtin RTEMS command)
  - `mount -t <type> <host>:<export> <local>`
  - Mounting a remote file system is required before RCE application modules can be transferred
  - File system types should be either "`nfs`" or "`tftp`"
- freeBlocks
  - Show the block usage on the RCE.
- reboot
  - `reboot [-S <system>] [-I <image>]`
  - Executes a warm reboot of the RCE, optionally choosing the system/image pair
  - Note: This sets the value of the front-panel rotary switch. To clear this setting, the user must perform a hard reset.
- syslog
  - `syslog [-l <n>]|[-f <n>]|[-c]`
  - Show or clear the system log.
  - `-l <n>` Show only the last n messages
  - `-f <n>` Show only the first n messages
  - `-c` Clear the system log
  - Arguments `-l`, `-n` and `-c` are mutually exclusive

Deployment: RCE Shell Commands (2)

- runTask
  - `runTask -N <taskName> [-P priority] [-S stacksize] [-M mode] [-A attribute] <filePath>`
  - `runTask -I <imageNumber>`
  - Executes the code in file pointed to by `<filePath>` with RTEMS task name as `<taskName>` (4 characters)
  - Optional arguments:
    - `-P <priority>`: RTEMS priority. Default = `100`
    - `-S <stacksize>`: RTEMS stack size. Default = `RTEMS_MINIMUM_STACK_SIZE`
    - `-M <mode>`: RTEMS Mode bits. Default = `RTEMS_DEFAULT_MODES`
    - `-A <attribute>`: RTEMS Attribute bits. Default = `RTEMS_DEFAULT_ATTRIBUTES`
    - `-I <imageNumber>`: Image number to run

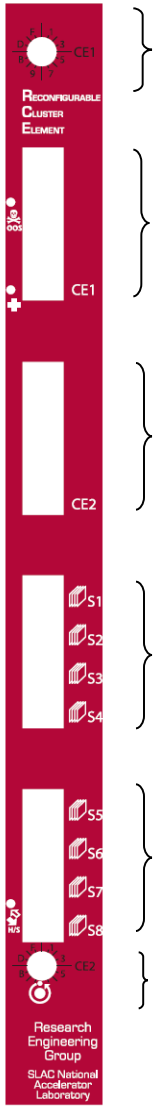
Deployment: RCE Shell Commands (3)

- burnTask
  - `burnTask -I <n> -N <taskName> [-P <priority>] [-S <stacksize>] [-M <mode>] [-A <attribute>] <filePath>`
  - Burn the file pointed to by `<filePath>` to Task slot number `<n>` and set the RTEMS task name to `<taskName>`
  - Arguments:
    - `-I <n>`: Task number
    - `-N <taskName>`: RTEMS task name, 4 characters
  - Optional arguments:
    - `-P <priority>`: RTEMS priority. Default = 1
    - `-S <stacksize>`: RTEMS stack size. Default = `RTEMS_MINIMUM_STACK_SIZE`
    - `-M <mode>`: RTEMS Mode bits. Default = `RTEMS_DEFAULT_MODES`
    - `-A <attribute>`: RTEMS Attribute bits. Default = `RTEMS_DEFAULT_ATTRIBUTES`

Deployment: RCE Shell Commands (4)

- lsTask
  - Show information about configured tasks such as task name, stack size, load address, RTEMS mode and attribute, and priority.
- lsSystem
  - Show information about System images such as load address, transfer address, and image size.
- stopTask
  - `stopTask [-I <id>] [-N <name>]`
  - Stop a task by name or by id.
- rmTask
  - `rmTask -I <n>`
  - Remove the Task image `<n>` from the RCE.
- rmContainer
  - `rmContainer -T <type> -I <n>`
  - Remove the Container `<n>` of type `<type>` from the RCE.
  - Required Arguments: `-I <n> -T <type>`
    - `<type> {DATA, TASK, CONFIGURATION, SYSTEM}`

Deployment: RCE Front-Panel Features



Rotary switch - boot selector CE1

LED status display CE1

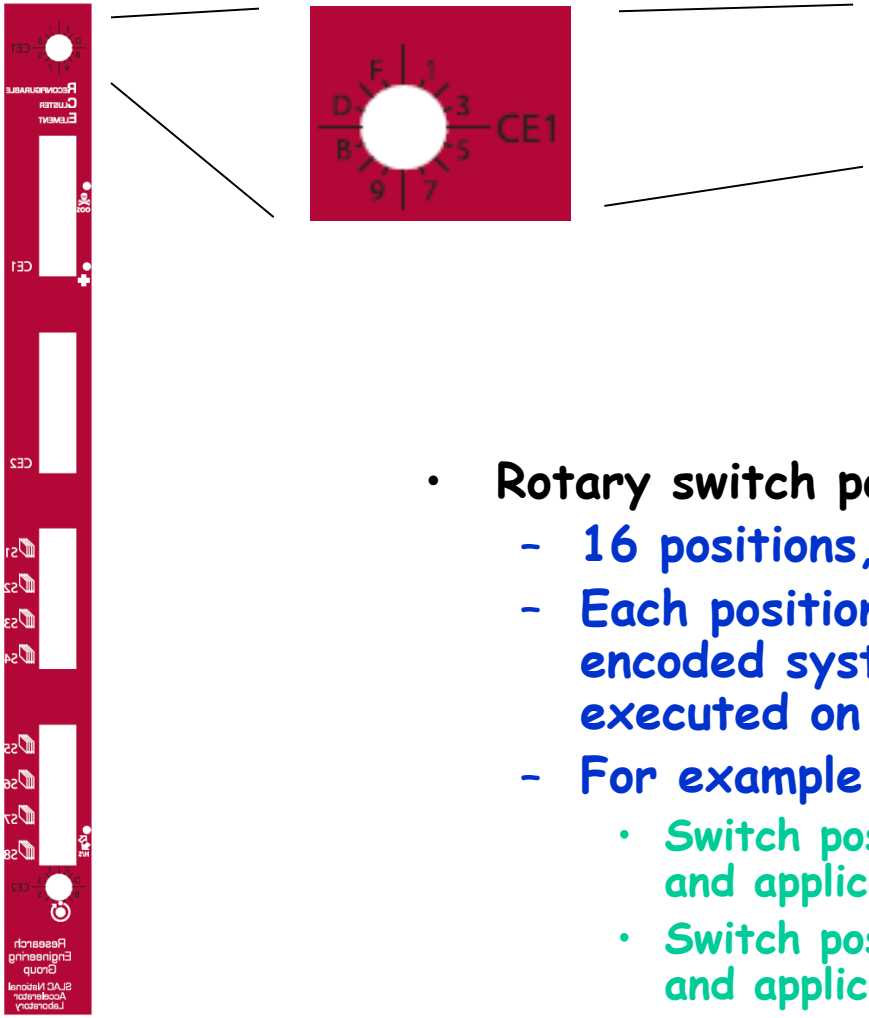
LED status display CE2

Networking status display bank 1

Networking status display bank 2

Rotary switch - boot selector CE2

Deployment: Booting the RCE



Switch position bitfield encoding

- Rotary switch position mapping
  - 16 positions, 0 - F
  - Each position represents a bitfield-encoded system/application pair to be executed on startup
  - For example:
    - Switch position 5 will boot system 0 and application 5
    - Switch position C will boot system 1 and application 4