Setting GearShift Points

1. Introduction

1.1. Purpose

The purpose of this document is to help understand how to set GearShift points using CodeWarrior Development Studio for StarCore 10.0.
2. Setting GearShift Points

2.1. Scope

The StarCore simulator allows you to shift between ISS and PACC modes at runtime on the same simulator device. In a debugger perspective, the difference between the two simulator types is the absence or presence of a cache. This can be helpful to profile only a section in an application.

The GearShift capability has the following new features:

- Two new targets in the launch configurations are used to indicate the starting mode: CCSSIM2 HS ISS and CCSSIM2 HS PACC
- A new toolbar for the debug perspective contains a button that can be used to switch between the two modes
- The following new debugger shell command can be used to facilitate the switch: cmdwin::sc::setGearShift ISS/PACC

2.2. GearShift Points Definition

The definition of the GearShift points is based on:

- The capability to define debugger shell actions attached to breakpoints,
- The existence of the setGearShift command.

To define a GearShift point, do the following:

1. Set a breakpoint
2. Select Properties>Actions tab>Available actions
3. Click New to create a new debugger shell action:
   a) Enter a name for the action
   b) Select Debugger Shell Action as its Action Type
   c) Select Commands and enter the following:
       cmdwin::sc::setGearShift PACC
       go
4. Click Attach to attach the newly created action to the breakpoint and save the breakpoint settings

When this breakpoint is reached, the mode changes to PACC, but the execution will not be suspended.
Figure 1 shows a case in which you want to profile only part of an application (func2). Two breakpoints are set. The first breakpoint is set on the func2() function call to attach a switch to PACC mode. The second breakpoint is set after the func2() function call, to attach a switch to ISS mode. The initial mode set in the launch configuration is CCSSIM2 HS ISS because you do not want to stop the application when these breakpoints are reached. Both actions include a go command.

Figure 1. GearShift Points