

## CREATING AND USING A SHARED DATA OBJECT

The typical use case for a shared data object in VxWorks® is to share information among tasks in different memory spaces, such as kernel tasks or tasks in one or more RTPs. Typically a task in one memory space (the “Creator”) creates the shared data region; then another task in a different memory space (the “Joiner”) attaches to the region. Tasks in those spaces may then freely access the shared region with the permissions they specify in the sdOpen() call.

### TASK GUIDE

Step	Action	Example
1	Create a project for the Creator and the Joiner.	
2	For the Creator, the first task to execute, write code that does the following:  a) Creates the shared region and captures its ID b) Writes a value to the first location in the region	The code will look something like this: <pre>SD_ID SdId; int *pValue; SdId = sdOpen("/SharedData", 0, OM_CREATE, 1000, 0,               SD_ATTR_RW   SD_CACHE_OFF, (void **)               &amp;pValue); * pValue = 0x12345678; . .</pre>
3	For the Joiner, write code that does the following:  a) Attaches to (gains access to) the shared data region and captures the ID b) Accesses the region and prints the contents of the first location in the region	The code will look something like this: <pre>SD_ID SdId; int *pValue; SdId = sdOpen("/SharedData", 0, 0, 1000, 0,               SD_ATTR_RO   SD_CACHE_OFF, (void **)               &amp;pValue); printf ("Shared data content is 0x%x\n", *pValue);</pre>

### Key Points

- The sdOpen() call is used either to create the region initially or to gain access to it. The only difference is the OM\_CREATE option.
- Each space may request different access rights.
- This example may be extended to any number of memory spaces.

education.windriver.com – [training@windriver.com](mailto:training@windriver.com)