Device Settings

Overview

RDK Device Settings library is a cross-platform library for controlling the following hardware configurations:

- Audio Output Ports (Volume, Mute, etc.)
- Video Ouptut Ports (Resolutions, Aspect Ratio, etc.)
- Front Panel Indicators
- Zoom Settings
- Display (Aspect Ratio, EDID data etc.)
- General Host configuration (Power managements, event management etc.)

The library is split into three major components

- Application Level APIs. (Comcast component)
- SoC level APIs. (SoC component)
- IARM support. (Comcast Component)

Application Level API

This is the API that application should use to control hardware configurations in a platform independent way. It also hides single-app and multi-app difference of the implementation from the applications. This allows the application to switch among different SoC versions or between single or multi app mode freely.

> Example: API to get the current video resolution :
>
> const VideoResolution & VideoOutputPort::getResolution() const

SoC Level API

SoC Level APIs that that need to implement by SoC vendors. It provides primitive and hardware specific implementation for each controllable aspect of their device. This level API is considered single-app mode only, even though its SoC implementation may potentially support multiple-app mode.

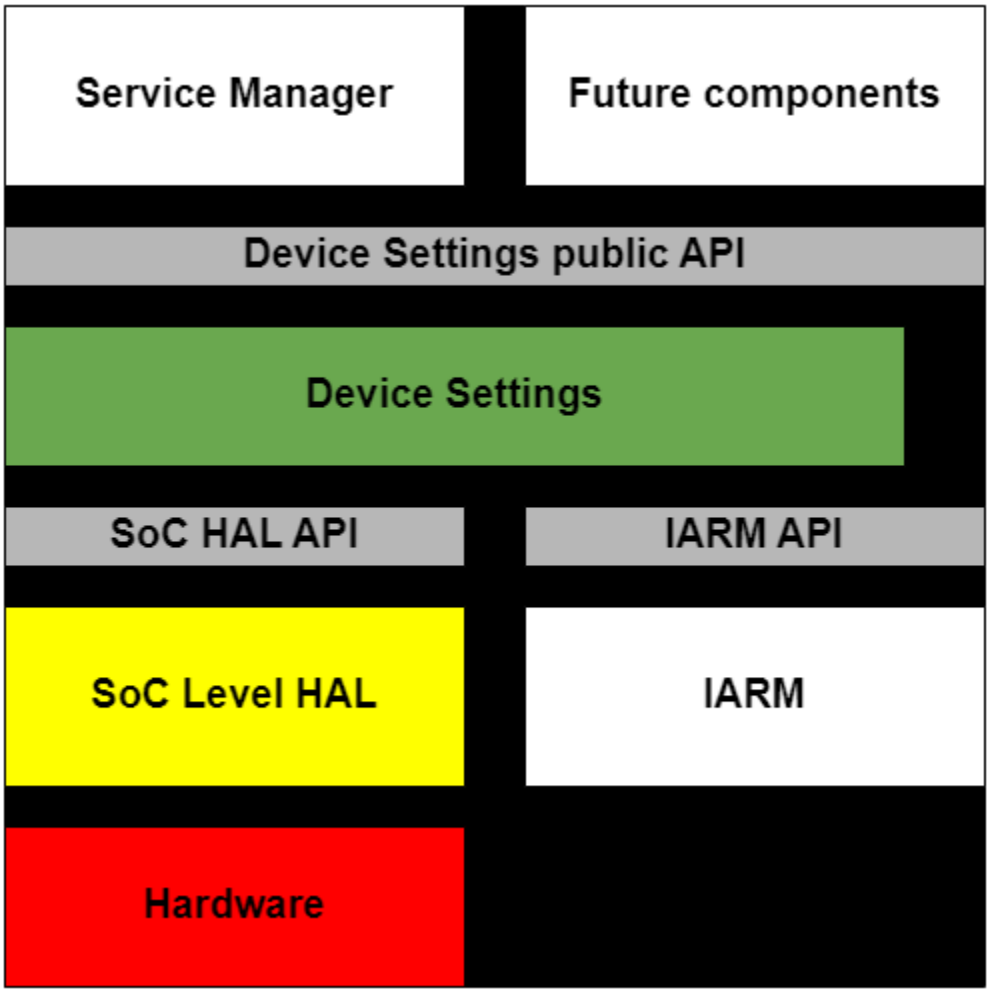
> Example: API to get the current video resolution :
>
> dsError_t dsGetResolution (int handle, dsVideoPortResolution_t *resolution)

IARM Support

If multiple applications need to control the device settings simultaneously, this component turns the single-app mode SoC level API into multi-app mode. Even though some SoC vendors implement the SoC level API to be multi-app capable, we still use Comcast's IARM support to achieve multiple-app mode. This allows the Application level API to remain truly platform neutral.

Architectural Overview

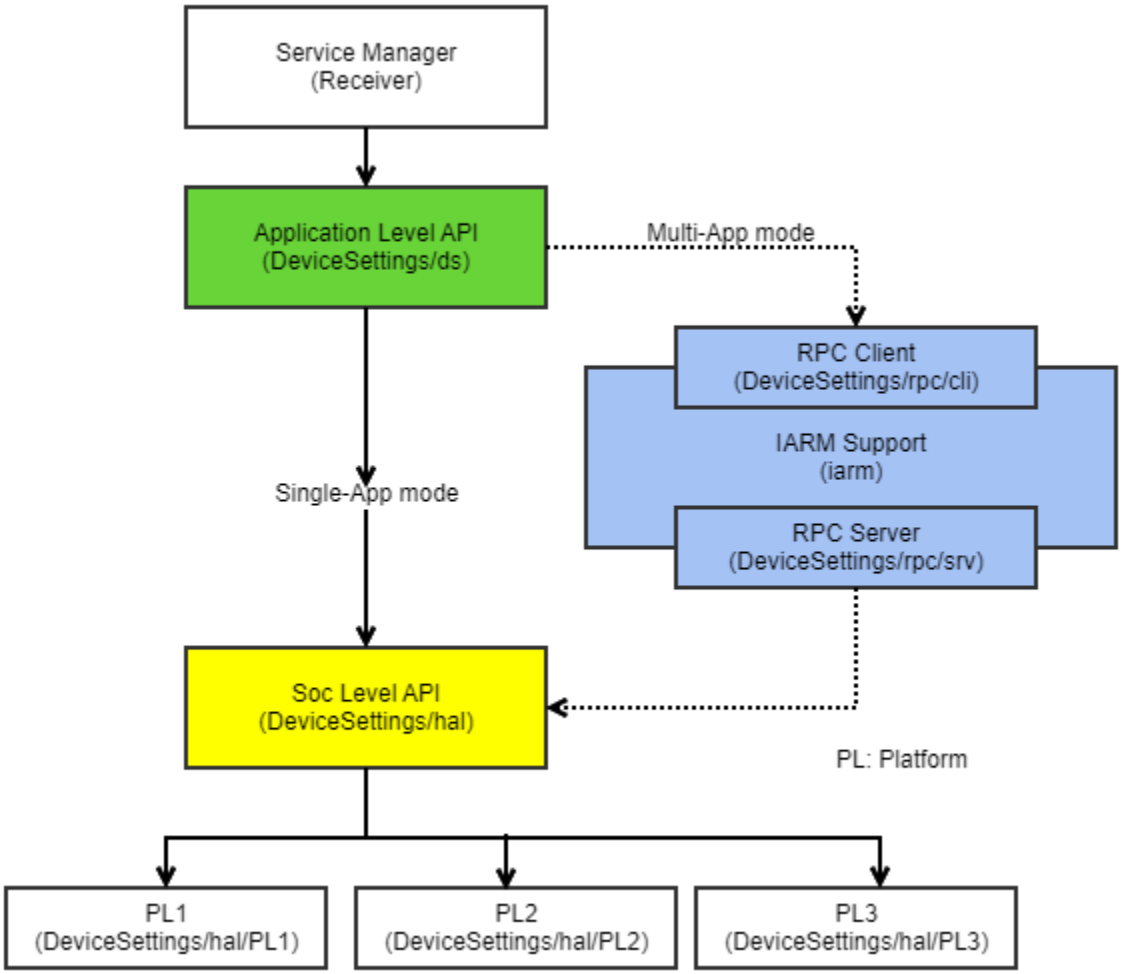
The Device Settings (DS) registers its services with the service manager. The Application uses/calls the DS Public API through service manager and DS Public API's intern calls the under lying SoC level API's to perform the required functionality.



Multi-App mode Support using IARM

Multi-app mode is used when multiple applications need to control the device settings simultaneously. Some SoC vendors implement the SoC level API to be multi-app capable, we still use Comcast's IARM support to achieve multiple-app mode. This allows the Application level API to remain truly platform neutral.

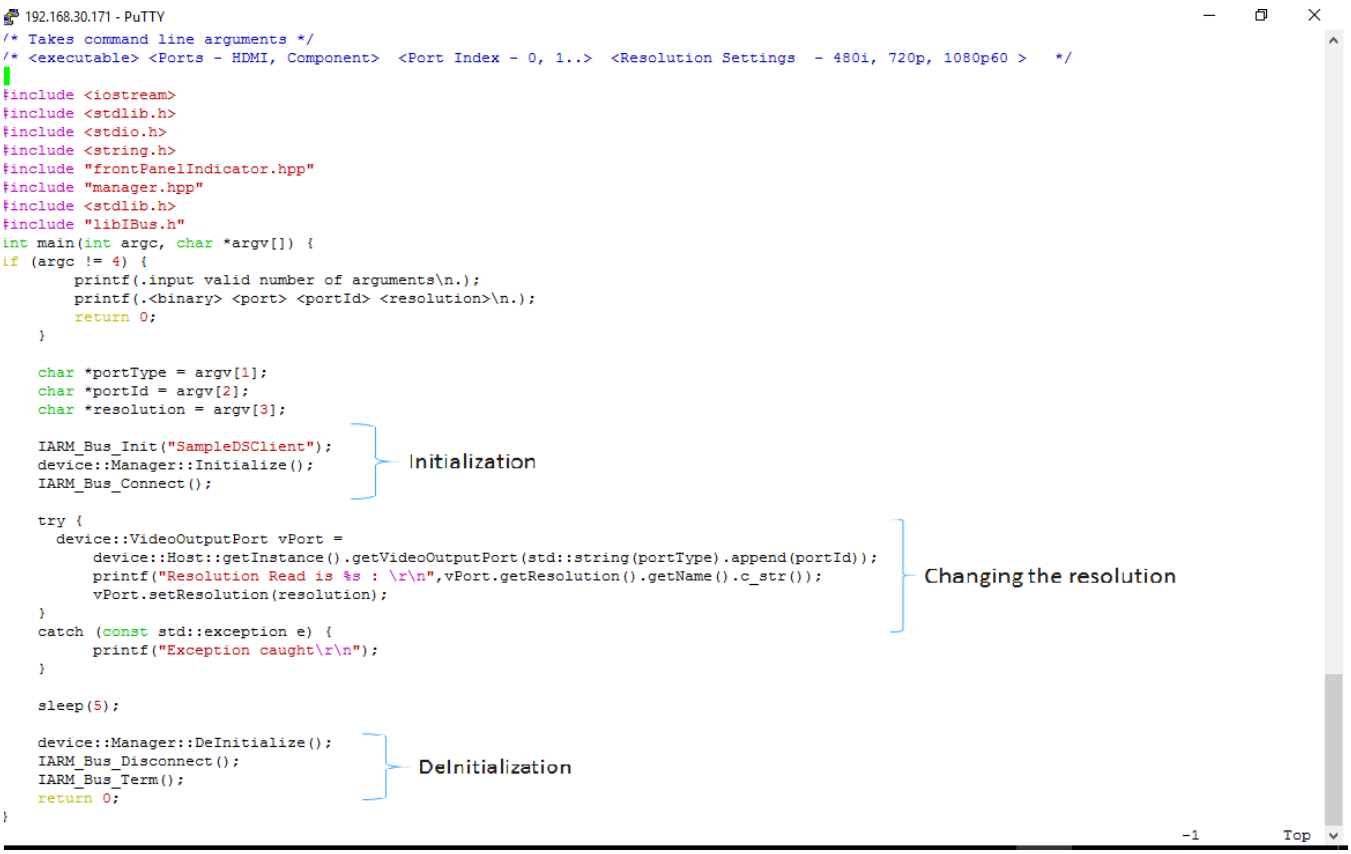
The below diagram depicts that how multi-app designed in RDK.



Device Settings Programming Guide

First we need to initialize the IARMbus and device manager, perform the required task making use of the API's and and finally need to de-initialize the device manager and disconnect the IARMbus.

Simple use case to set the resolution



API Documentation

To know more about SoC/Application level APIs details use in RDK, refer the link Device Settings API Documentation