Mesa3 Android RFID Source Code Explanation

Assumptions:

- You have already signed the RFID INDEMNIFICATION AGREEMENT
- You intend to write your own RFID application for the Android Mesa3

Your responsibilities:

- Regulatory compliance – This is the reason you signed the agreement, to control transmit power levels, operate in the correct frequency (region).
- Control power during sleep – Your application is responsible for putting the RFID expansion into a low power mode during sleep (i.e. kill power to it, and re-connect after resume).

RFID Configuration:

Each RFID Mesa3 is pre-configured with a configuration. This configuration is saved in a non-volatile area and special programming is required to retrieve it.

1) **The region**, such as NA2, EU3, or AU (for North America, Europe, or Australia). This corresponds to Jadak's regions that are set using the following function.

```
RfidReader.ParamSet("/reader/region/id", configuredRegion);
```

2) **The maximum average power level**. At the time of this writing, all planned regions are allowed to operate continuously at maximum power. But the RFID configuration that is saved in the device has provisions for regions where we might be restricted to a lower power level. You must follow the power level that is saved in the configuration data of the Mesa3. For example, the M6e Nano module is capable of running at 27 dBm, but due to regulatory reasons, you might be restricted to 20 dBm (2000 centi-dBm), and so if the RFID module is going to be operated continuously, 20 dBm is the highest power setting you can use. If, however, you are going to be scanning intermittently, you just need to ensure that the *average* power is kept below 20 dBm by preventing scans for a certain amount of "off time" following a scan (see "Scanning" section below).

Here are some example RFID Configurations:

1) NA2, 2000 – This is configured for North America (using NA2), has a maximum average power level of 2000 centi-dBm (20 dBm).
2) EU3, 2400 – This is configured for Europe (using EU3), has a maximum average power level of 2400 centi-dBm (24 dBm).

Software Components used:

The Mesa3 includes an expansion driver. This is how power is turned on or off to the RFID module; and cellular is disabled or re-enabled.

Source Code:

Application Startup

When this application starts up, the onCreate() function hooks up to the expansion driver (appCreate()), checks for a valid expansion ID, reads in the Region, sets up some USB intent filters, and then turns on power. Upon power being turned on, the RFID will now enumerate on USB, and initialization continues in the BroadcastReceiver. Once the app has permission from the user, the remaining initialization is handed off to a thread (InitThread()). In this thread, communication is established.

Handling Sleep/Resume

In onCreate(), we register for screen on/off broadcast notifications. Before the Mesa3 enters sleep, it is necessary to close the RfidReader and power down the RFID module. This means that after the Mesa3 resumes, that most of the startup code must be ran again.

Scanning and handling "Off Time"

See the TriggerScan() function. Here, when we are ready to scan, we check to see if we are allowed to scan yet or not. If the last scan was more powerful than the maximum average power level allowed, we might have to wait for so much "off time" before we can allow a scan again. The scan time is timed so we can calculate how much off time we must enforce for next time. See also the calculateOffTime() function.