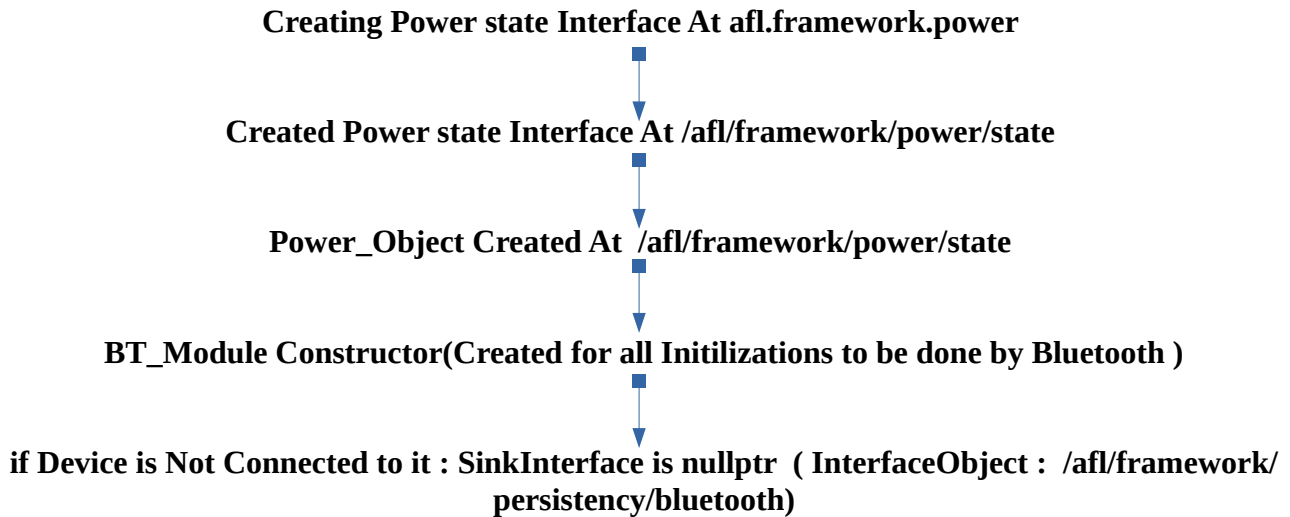


BLUETOOTH FLOW

Firstly When the Bluetooth is Powered ON,the below process runs when the bluetooth turns on the bluetooth adapter first registers with the system-bus through the middleware.

When We are Running Bluetooth Contrl command & also Running the Bluetooth at Middleware with



Before Creating the Bluetooth Module Constructor it has to register at session bus from there the interface is created at Middleware with some reference.

The Reference object will Added from Middleware to System bus and For Accessing the bluetooth we need to Register at Bluez.org.

Here Middleware acts As the intermidate path or connecting path between the Bluetooth Adapter and the D-bus.

The Below State Machine Cycle Explains about What is Happening While Turning ON .

It Firstly Checks Where Bluetooth is in ON or OFF state.It directly goes to the **BT_Off State**. If it is in OFF state then it checks **BT_Turned_Off** already it **Turned off,Then it checking for BT_Turning_On**.

If the Bluetooth is in **Turning_On state** then it goes to **BT_ON state** after that it enters into the **BT_Turned_ON state**. It is in **Turned on state** then it checks whether it goes to **BT_Turning_Off state**. Here in the above State Cycle it creates Entry nodes for each state in BT State Machine as it has entered into that state and After Turning off the Bluetooth the Nodes created for BT State Machine are Destroyed.

After the Bluetooth is in ON State then it creates path for Bluetooth Entry as

Create_Interface_Adapter(Adapter interface signal)
Custom_AdapterProperty_UUID(Unique id is created for Adapter)

In the below State Adapter inteface is Added at Middleware

BT::Ad_Interface_Added (Registering signals for Adapter interface in Middleware)

BLUETOOTH FLOW

BT_Resource::Create_Interface_Source_Adapter (Created source Adapter object in middleware)

Adapter resourceObject is not null in middleware(Here no Interface is connected)

After the device get Connected likewise a **Signal InterfacesAdded: interface name: afl.middleware.bluetooth.Adapter** and the new path is Created for Bluetooth
/afl/middleware/bluetooth.

If interface is already registered it directly get connected internally with giving a signal and Path name because it directly goes the path which was perviously created.

The Next Step in this Process is the BT_Scan State Machine.

The Default State is BT_Scan_Off State.

If it is in Scan_Off state, it checks for the BT_Scan_On State.

Here When the Scan state is ON the device nears it range are Created with interface name as(**afl.middleware.bluetooth.SearchDevices**)

After the BT_Scan is in ON state it checks for the devices near by its range.

The Devices which are present Near are already device is Paired in it is directly goes to the **BT_Paired State** and Creates node for the device **along with the device MAC addresses path(/org/bluez/hci0/dev_E4_B5_03_56_C6_52) and interface name: (afl.middleware.bluetooth.PairedDevices)**

BT_Device::Dev_Paired(AflStateNode Prev state: BT_Unpaired and new state is BT_Paired)

The devices gets Paired with enter into paired list and removed from the search device list

Signal InterfacesRemoved: interface name: afl.middleware.bluetooth.SearchDevices path: /org/bluez/hci0/dev_E4_B5_03_56_C6_52 **(Removed Signal from Search Devices)**

If the BT_Scan is ON it goes to the Next State Cycle for Device to Connect.

Default State entry into the **BT_Unpaired State**.If the device is in **unpaired state** it cheks for the device **BT_Paired State** for the device to connect to it.Next it goes to the **BT_Disconnected State** if it is in **disconnected state** it checks for the **BT_Connected State** if the device is connected to the bluetooth is gets the **interface signal node is created at Entry state (Node Entry: BT_Device).**

Here the Paired Device goes in to paired list BT_Device::Dev_Paired and the Node is Created Comparing Betweenprev state and new state(**Prev state: BT_Unpaired and new state is BT_Paired**).

BT_Paired::Entry (Enterd into paired state)

BT_Resource::Create_Interface_Source_PairedDevices (Interface created at Paired Device list)

BT_Connected::Entry (Entry into device Connected list)

BT_Resource::Create_Interface_Source_ConnectedDevices (Interface is created for Connected device)