Introducing Wi-Fi and Bluetooth Application in AGL Charming Chinook

2nd June 2017
SRI MALDIA HARI ASTI
Introduction

- SRI MALDIA HARI ASTI (ALPS Electric Japan)
- Location: Furukawa – Osaki City – Miyagi Prefecture – Japan
- Wi-Fi Software Engineer for Automotive (2013 ~ Present)
- Email: hariasti.srimaldia-1@jp.alps.com
ALPS Electric has become a member since 2015 and decided to directly contribute into connectivity field of AGL. We have been developing application for Wi-Fi and Bluetooth for AGL, including middle layer (API) and UI, which works on AGL application framework. Our applications have been merged to Charming Chinook distribution, however not yet been presented in public. This talk will introduce about Wi-Fi and Bluetooth application in AGL Charming Chinook and listed APIs we have inside, as guidance to any developers who would like to apply Wi-Fi and Bluetooth connection into their application into AGL.
Outline

① Company Profile

② ALPS Activity in AGL

③ Bluetooth and Wi-Fi’s APIs in CC (Charming Chinook)

④ Live Demo and Result
  4-1 AGL Reference Environment
  4-2 AGL ft. ALPS product

⑤ Latest version in Master Branch

⑥ Conclusion
① Company Profile

② ALPS Activity in AGL

③ Bluetooth and Wi-Fi’s APIs in CC

④ Live Demo and Result
   4-1 AGL Reference Environment
   4-2 AGL ft. ALPS product

⑤ Latest version in Master Branch

⑥ Conclusion
Company Profile

★ To provide best solution to the customers ★
Company Profile

Keyless Entry
- Vehicle positioning detect
- Keyless entry
- Car sharing service

Autonomous Vehicle
- V2V / V2I
- High accuracy positioning
- 3D Map updating

Infotainment
- Hands free
- Audio streaming
- Phone book download
- Wireless MirrorLink
- Mostly support Linux OS

Network access
- Tethering
- Web browsing
- Emergency call
- Vehicle data for diagnostic
- Mostly support Linux OS

ALPS Connectivity for Vehicle
- BT BLE
- V2X
- Wi-Fi
- GNSS
- LTE/5G

ALPS eager to support all car connectivity platforms (Genivi, AGL, etc)
Outline

① Company Profile

② ALPS Activity in AGL

③ Bluetooth and Wi-Fi’s APIs in CC (Charming Chinook)

④ Live Demo and Result
   4-1 AGL Reference Environment
   4-2 AGL ft. ALPS product

⑤ Latest version in Master Branch

⑥ Conclusion
ALPS Activity in AGL

April, 2015
Joined AGL

May, 2016
Connectivity was not supported.
Participate in AGL event, especially Connectivity EG

Oct, 2016
Start contribution to AGL for Wi-Fi and Bluetooth

Jan, 2017
Wi-Fi Bluetooth applications are merged into CC distribution

Automotive Linux Summit June 2017
Outline

① Company Profile

② ALPS Activity in AGL

③ Bluetooth and Wi-Fi’s APIs in CC (Charming Chinook)

④ Live Demo and Result
   4-1 AGL Reference Environment
   4-2 AGL ft. ALPS product

⑤ Latest version in Master Branch

⑥ Conclusion
ALPS source code has been merged!

- Charming Chinook branch (support http protocol):
  
  $ git clone https://gerrit.automotivelinux.org/gerrit/apps/settings -b chinook

- Master branch (support websocket)
  
  $ git clone https://gerrit.automotivelinux.org/gerrit/apps/settings
Bluetooth and Wi-Fi’s APIs in CC - 3

**Bluetooth**
- Bluetooth UI (QML)
  - QML
  - Bluetooth API (Plugin)
    - C
    - dbus
  - BlueZ
  - Linux API
    - Bluetooth Driver
  - Bluetooth device (USB dongle)

**Wi-Fi**
- Wi-Fi UI -> +QT Company
  - C
  - dbus
  - AGL security framework
    - Connnman
    - dbus
    - wpa_supplicant
    - dbus
    - Wi-Fi Driver
  - Wi-Fi device (USB dongle)

**Linux**
- Bluetooth
  - BlueZ
  - Linux API
- Wi-Fi
  - wpa_supplicant
  - Linux API

**ALPS**
- Contribution
### List of APIs in CC

<table>
<thead>
<tr>
<th>Features</th>
<th>API (verb name)</th>
<th>Features</th>
<th>API (verb name)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BLUETOOTH</strong></td>
<td></td>
<td><strong>Wi-Fi</strong></td>
<td></td>
</tr>
<tr>
<td>Power ON/OFF</td>
<td>power</td>
<td>Power ON</td>
<td>activate</td>
</tr>
<tr>
<td>Start Scanning</td>
<td>start_discovery</td>
<td>Power OFF</td>
<td>deactivate</td>
</tr>
<tr>
<td>Stop Scanning</td>
<td>stop_discovery</td>
<td>Start scanning</td>
<td>scan</td>
</tr>
<tr>
<td>Display scan result</td>
<td>discovery_result</td>
<td>Display scan result</td>
<td>scan</td>
</tr>
<tr>
<td>Pairing</td>
<td>pair</td>
<td>Connection</td>
<td>connect</td>
</tr>
<tr>
<td>Remove pairing</td>
<td>remove_device</td>
<td>Disconnection</td>
<td>disconnect</td>
</tr>
<tr>
<td>Cancel pairing</td>
<td>cancel_pair</td>
<td>Check connection status</td>
<td>status</td>
</tr>
<tr>
<td>Connection</td>
<td>connect</td>
<td>Input password</td>
<td>security</td>
</tr>
<tr>
<td>Disconnection</td>
<td>disconnect</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set device property</td>
<td>set_property</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Bluetooth Application Example in CC

UI (User Interface) @ app/bluetooth/Bluetooth.qml

http://localhost/<port>/Bluetooth-manager/power?value=1

API @ binding-bluetooth/bluetooth-api.c

```c
/* VB'S NAME                  FUNCTION TO CALL */
.
.type = AFB_BINDING_VERSION_1,
.vl = {
  .info = "Application Framework Binder - Bluetooth Manager plugin",
  .prefix = "Bluetooth-Manager",
  .verbs = binding_verbs
}

/API: binding_verbs

.verbs = {
  .name = "power",
  .callback = bt_power,
  .name = "start_discovery",
  .callback = bt_start_discovery,
  .name = "stop_discovery",
  .callback = bt_stop_discovery,
  .name = "discovery_result",
  .callback = bt_discovery_result,
  .name = "remove_device",
  .callback = bt_remove_device,
  .name = "pair",
  .callback = bt_pair,
  .name = "cancel_pair",
  .callback = bt_cancel_pairing,
  .name = "connect",
  .callback = bt_connect,
  .name = "disconnect",
  .callback = bt_disconnect,
  .name = "set_property",
  .callback = bt_set_property,
};
```

```c
/* value parameter is "1" or "true" */
else if (atoi(value) == 1 || strcmpc(value, "true") ) {
  if (adapter_set_powered (TRUE)) {
    aFB_req_fail (request, "failed", "no more radio devices available");
    return;
  }
  json_object_object_add (resp, "power", json_object_new_string ("on"));
  pathH.music_status (ACTIVE);
}
/* value parameter is "0" or "false" */
else if (atoi(value) == 0 || strcmpc(value, "false") ) {
  if (adapter_set_powered (FALSE)) {
    aFB_req_fail (request, "failed", "Unable to release radio device");
    return;
  }
  json_object_object_add (resp, "power", json_object_new_string ("off"));
  pathH.music_status (INACTIVE);
}
```
Wi-Fi Application Example in CC

UI (User Interface) @ app/wifi/Wifi.qml

http://localhost/<port>/wifi-manager/activate

API @ binding-wifi/wifi-api.c

```c
static const struct atb_binding binding
    /* description conforms to VERSION 1 */
    .type = AFB BINDING VERSION 1, .v1 = {
        .prefix = "wifi-manager", /* the API name */
        .info = "wifi API", /* short description */
        .verbs = binding_verbs /* the array describes */
    }

/* VERB'S NAME -> FUNCTION TO CALL */
   .name = "activate", .callback = wifi_activate,
   .name = "deactivate", .callback = wifi_deactivate,
   .name = "scan", .callback = wifi_scan,
   .name = "scan_result", .callback = wifi_scanResult,
   .name = "connect", .callback = wifi_connect,
   .name = "status", .callback = wifi_status,
   .name = "disconnect", .callback = wifi_disconnect,
   .name = "reconnect", .callback = wifi_reconnect,
   .name = "security", .callback = wifi_insertPasskey,
```

```c
static void wifi_activate(struct atb_req request) /*AFB_SESSION_CHECK*/
{
    json_object *jresp;
    GError *error = NULL;

    if (ptr_my_callback == NULL) {
        printf("Registering callback
n");
        ptr_my_callback = ask_for_passkey;
        register_callback(ptr_my_callback);
    }

    jresp = json_object_new_object();
    json_object_object_add(jresp, "activation", json_object_new_string("on"));
    error = do_wifiActivate();
    if (error == NULL) {
       afb_req_success(request, jresp, "Wi-Fi - Activated");
    } else {
       afb_req_fail(request, "failed", error->message);
    }
}
```
Bluetooth and Wi-Fi’s APIs in CC - 2

Wi-Fi and Bluetooth Application in **CC (Charming Chinook)**

- Support Wi-Fi and Bluetooth basic features and connection
- Support AGL framework
- Compatible with BlueZ and connman
- Connect UI and API via http protocol

- Not yet support receive event
- Not yet support security on framework
- No yet support websocket
- Not yet support network management

Lot of things to do to make it better…
Outline

① Company Profile

② ALPS Activity in AGL

③ Bluetooth and Wi-Fi’s APIs in CC (Charming Chinook)

④ Live Demo and Result
   4-1 AGL Reference Environment
   4-2 AGL ft. ALPS Product

⑤ Latest version in Master Branch

⑥ Conclusion
Demo 1.
AGL Reference Environment
Demo 1. AGL Reference Environment

Upload Demo video on youtube:
https://www.youtube.com/watch?v=HLEcL7M_eY&feature=youtu.be
Demo 1. AGL Reference Environment

Bluetooth

LIVE DEMO

Input Password

Wi-Fi scan

Paired connected

Demo 4G GSM Connection...
Demo 2.
AGL ft. ALPS Product
AGL ft. ALPS Product

DEMO VIDEO
Outline

① Company Profile

② ALPS Activity in AGL

③ Bluetooth and Wi-Fi’s APIs in CC (Charming Chinook)

④ Live Demo and Result
   4-1 AGL Reference Environment
   4-2 AGL ft. ALPS product

⑤ Latest version in Master Branch

⑥ Conclusion
## List of APIs in CC

<table>
<thead>
<tr>
<th>Features</th>
<th>BLUETOOTH</th>
<th>Wi-Fi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power ON/OFF</td>
<td>power</td>
<td>activate</td>
</tr>
<tr>
<td>Start Scanning</td>
<td>start_discovery</td>
<td>deactivate</td>
</tr>
<tr>
<td>Stop Scanning</td>
<td>stop_discovery</td>
<td>scan</td>
</tr>
<tr>
<td>Display scan result</td>
<td>discovery_result</td>
<td>scan</td>
</tr>
<tr>
<td>Pairing</td>
<td>pair</td>
<td>connect</td>
</tr>
<tr>
<td>Remove pairing</td>
<td>remove_device</td>
<td>disconnect</td>
</tr>
<tr>
<td>Cancel pairing</td>
<td>cancel_pair</td>
<td>status</td>
</tr>
<tr>
<td>Connection</td>
<td>connect</td>
<td>security</td>
</tr>
<tr>
<td>Disconnection</td>
<td>disconnect</td>
<td>eventadd</td>
</tr>
<tr>
<td>Set device property</td>
<td>set_property</td>
<td>evensub</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eventpush</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eventunsub</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eventdel</td>
</tr>
</tbody>
</table>
Latest version in Master branch

Wi-Fi and Bluetooth Application in Master Branch

- Support Wi-Fi and Bluetooth basic features and connection
- Support AGL framework
- Compatible with BlueZ and connman
- Connect UI and API via http protocol
- Support receive event
- Support websocket

Not yet support security on framework
Not yet support network management

We need help to make it better!
① Company Profile
② ALPS Activity in AGL
③ Bluetooth and Wi-Fi’s APIs in CC
④ Live Demo and Result
  4-1 AGL Reference Environment
  4-2 AGL ft. ALPS product
⑤ Latest version in Master Branch
⑥ Conclusion
Conclusion

1. AGL finally supports Bluetooth and Wi-Fi connectivity pioneered by ALPS
2. ALPS Product (Hardware and Software) compatible with AGL platform
3. ALPS invites everybody to improve Bluetooth and Wi-Fi application as one of AGL OSS applications by submitting advice and/or patches into JIRA and Gerrit
4. Contact us for further ALPS hardware and software information
   ALPS Electric Co.,Ltd
   http://www.alps.com/e/index.html
   SRI MALDIA HARI ASTI
   hariasti.srimaldia-1@jp.alps.com
ALPS Connectivity Solutions for Automotive

Creating new value that satisfies stakeholders and is friendly to the Earth
Advancing society and caring for the environment through perfecting electronics