

Building MicroAPITester for PIE

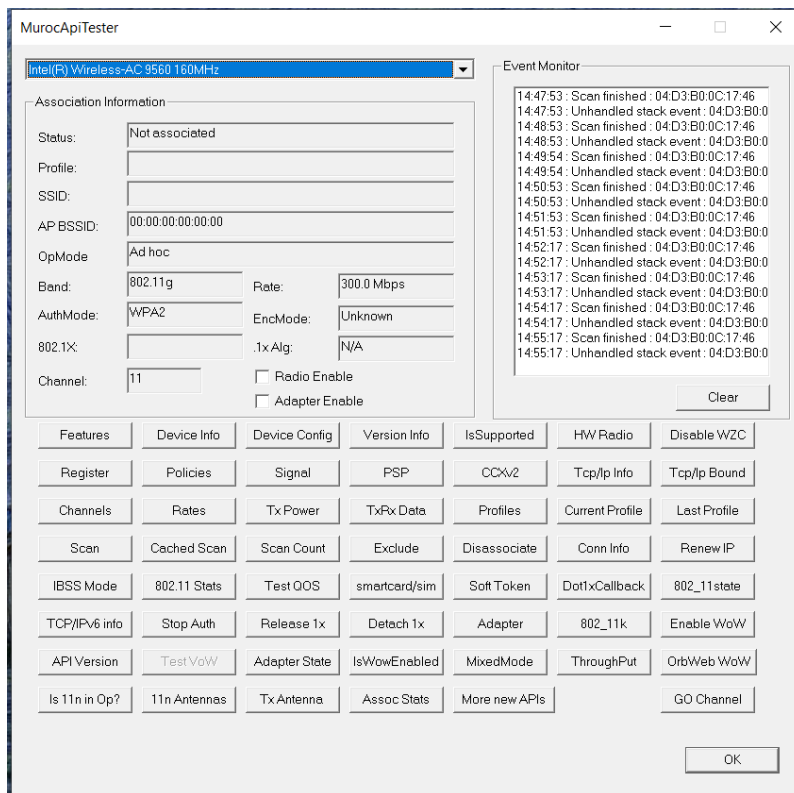
Thanks to Zhiqiang's PPT and explanation.

Background

More and more OEM will develop characteristic Applications on PC with Intel WiFi APIs

In PROSET era, there is "MurocAPITester.exe", but can not work with PIE

Build a new similar MurocAPITester for PIE



Some Popular APIs

Used for SoftAP/GO start and client fast connection to SoftAP/GO:

Query Connected AP info: GetConnectedRSSI

Dialog

Query Supported Channels

Query Supported Channel

MCC :

Channel Array Size

Channel Array :

Query Operating Channels

Query Operating Channel

GO State :

Band Type :

Channel :

Channel Width

Set Discovery Filter

Set Discovery Filter

Channel :

Filter Bitmask

MAC Address

SSID :

Set Preferred Channel

Set Preferred Channel

Reset Preferred Channel

Reset Discovery Filter

OK

Prepare for PDK Compile

Download PHWFW01690_21.110.1.1PDK from VIP

PC WLAN config: JFP above + 21.90 Driver above + PIE installation

Install Microsoft Visual Studio Professional 2019

PIE SDK Architecture

Programming Language: C

Load MurocApi.dll (Created after PIE installation):

1. Locate the Device Node
2. Read the registry which Include the full path into the driverStore
3. Load the Dll

Using SDK APIs

1. Get WiFi Adapter List via "WifiGetAdapterList"

2. RegisterIntelCallback
3. Call corresponding PIE APIs
4. Un-RegisterIntelCallback
5. Free Memory

Example of API “GetConnectedRSSI” execute

4.17 GetConnectedRSSI

This function retrieves the RSSI from the connect AP.

```
MUROC_RET APIENTRY GetConnectedRSSI(const HADAPTER hAdapter,
PINTEL_WIFI_HEADER pHeader, void* pInputData)
```

GetConnectedRSSI parameters

Term	Description
hAdapter	Handle to the adapter
pHeader	Intel header containing the version and size of INTEL_WIFI_CONNECTED_AP_RSSI
pInputData	Pointer to an INTEL_WIFI_CONNECTED_AP_RSSI structure

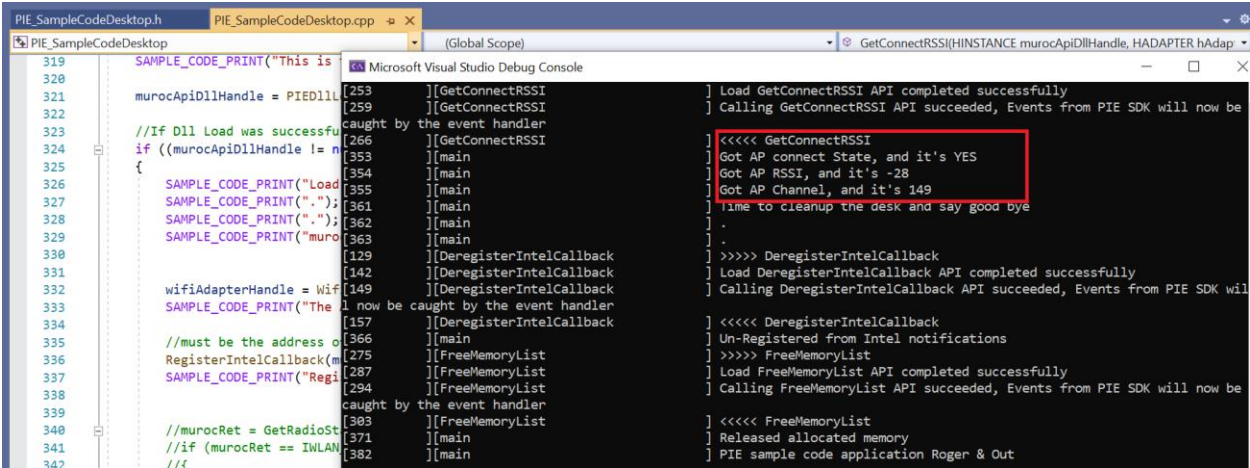
GetConnectedRSSI return value

Value	Meaning
IWLAN_E_SUCCESS	Successful
IWLAN_E_FAILURE	Failure
IWLAN_E_INVALID_PARAMETER	Invalid parameter

```
28 typedef MUROC_RET(APIENTRY *GETCONNECTEDRSSI) (HADAPTER hAdapter, INTEL_WIFI_HEADER* pWifiHeader, MurocDefs::INTEL_WIFI_CONNECTED_AP_RSS
```

Build and Compile

Currently PDK can be successfully executed via Console



For example, if I want to add `WifiGetDeviceInfo` into the project,

1. See the doc (like below) to check the parameters it wants and add this into `PIE_SampleCodeDesktop.h`

4.62 WifiGetDeviceInfo

This function retrieves the miscellaneous device information for the specified adapter.

```
MUROC_RET APIENTRY WifiGetDeviceInfo(
    HADAPTER hAdapter,
    PINTEL_HEADER pHeader,
    void* pDevInfo)
```

WifiGetDeviceInfo parameters

Term	Description
hAdapter	Handle to the adapter
pHeader	Intel header containing the version and size of INTEL_WIFI_DEVICE_INFO
pDevInfo	Pointer to an INTEL_DEVICE_INFO structure

```
typedef MUROC_RET(APIENTRY* GETDEVICEINFO) (HADAPTER hAdapter, MurocDefs::INTEL_DEVICE_INFO* pDevInfo);
```

*Ignore the pHeader because it's already written in the main function.

*If the parameter is defined in MurocDefs (ex: Pointer to an INTEL_..... structure), then remember to add "MurocDefs::" in front of the parameter.

2. In PIE_SampleCodeDesktop.cpp, add function like this

```
MUROC_RET GetDeviceInfo(HINSTANCE murocApiDllHandle, HADAPTER hAdapter, INTEL_DEVICE_INFO* pDevInfo)
{
    GETDEVICEINFO getDeviceInfoFunc = nullptr;
    DWORD dwError = ERROR_SUCCESS;
    MUROC_RET murocApiRetVal = IWLAN_E_FAILURE;

    SAMPLE_CODE_TRACE_IN;

    getDeviceInfoFunc = (GETDEVICEINFO)GetProcAddress(murocApiDllHandle, "GetDeviceInfo");

    if (getDeviceInfoFunc == nullptr)
    {
        dwError = GetLastError();
        SAMPLE_CODE_PRINT("GetProcAddress \"__FUNCTION__\" status %lu", dwError);
    }

    if ((getDeviceInfoFunc != nullptr))
    {
        SAMPLE_CODE_PRINT("Load \"__FUNCTION__\" API completed successfully");

        murocApiRetVal = getDeviceInfoFunc(hAdapter, pDevInfo);

        if (murocApiRetVal == IWLAN_E_SUCCESS)
        {
            SAMPLE_CODE_PRINT("Calling \"__FUNCTION__\" API succeeded, Events from PIE SDK will now be caught by the event handler");
        }
        else
        {
            SAMPLE_CODE_PRINT("Calling \"__FUNCTION__\" API failed with status %ld", murocApiRetVal);
        }
    }
    SAMPLE_CODE_TRACE_OUT;
    return murocApiRetVal;
}
```

3. In the main function, first declare the variable you want.

```
INTEL_DEVICE_INFO PDevInfo = { 0 }; //I add this.
```

Then call the function you just wrote and print the result by "SAMPLE_CODE_PRINT"

```
murocRet = GetDeviceInfo(murocApiDllHandle, wifiAdapterHandle, &PDevInfo);
if (murocRet == IWLAN_E_SUCCESS)
{
    SAMPLE_CODE_PRINT("Got Device Info MAC address, and it's %u ", PDevInfo.btMACAddress);
    SAMPLE_CODE_PRINT("Got Device Info UC Revision ID, and it's %u ", PDevInfo.ucRevisionID);
    SAMPLE_CODE_PRINT("Got Device Info US Revision ID, and it's %hu ", PDevInfo.usDeviceID);
    SAMPLE_CODE_PRINT("Got Device Info US Hardware Version ID, and it's %hu ", PDevInfo.usHardwareVersionId);
    SAMPLE_CODE_PRINT("Got Device Info US Subsystem Vendor ID, and it's %hu ", PDevInfo.usSubsystemVendorID);
    SAMPLE_CODE_PRINT("Got Device Info US System ID, and it's %hu ", PDevInfo.usSubsystemID);
    SAMPLE_CODE_PRINT("Got Device Info US Vendor ID, and it's %hu ", PDevInfo.usVendorID);
}
```

*If you don't know what is in the structure, you can see what VSCode hints you (like below)

```
SAMPLE_CODE_PRINT("Got Device Info US Vendor ID, and it's %hu ", PDevInfo.);  
    (field) public : UCHAR MurocDefs::INTEL_DEVICE_INFO::btMACAddress[6]  
pcRet = GetRadio  
SAMPLE_CODE_PRINT("File: PieDefinitions.h  
(murocRet == IWLAN_E_SUCCESS)  
  
SAMPLE_CODE_PRINT("At first, Got WiFi Radio State, and it's %s ", pbEnable  
pbManualOff = true;  
pbEnable = true;  
SAMPLE_CODE_PRINT("At first, Got WiFi Radio State, and it's %s ", pbEnable
```

4. See the result.

```
[240] ][GetDeviceInfo] ] >>>> GetDeviceInfo  
[253] ][GetDeviceInfo] ] Load GetDeviceInfo API completed successfully  
[259] ][GetDeviceInfo] ] Calling GetDeviceInfo API succeeded, Events from PIE SDK will now be caught by the event handler  
[266] ][GetDeviceInfo] ] <<<< GetDeviceInfo  
[445] ][main] ] Got Device Info MAC address, and it's 3591371416  
[446] ][main] ] Got Device Info UC Revision ID, and it's 48  
[447] ][main] ] Got Device Info US Revision ID, and it's 40432  
[448] ][main] ] Got Device Info US Hardware Version ID, and it's 0  
[449] ][main] ] Got Device Info US Subsystem Vendor ID, and it's 32902  
[450] ][main] ] Got Device Info US System ID, and it's 48  
[451] ][main] ] Got Device Info US Vendor ID, and it's 32902
```