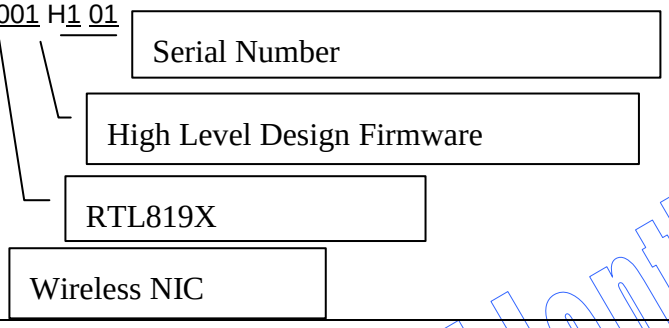
RealTek Mass Production DLL Specification

Document Number: 01-001-M4-0004

Naming Rule: Reference Internal Engineer Development Document Code

Example: 01 0001 H1 01



Document Code Management

			Code
RTL Document		Functional Spec	F1
		High Level Design	H1
		Low Level Design	L1
		Test Plan, Test Report	T1
		Guideline	G1
		Maintain	M1
		Decisions , Meeting Notes	D1

Realtek Semiconductor Corp.

Headquarters

1F, No. 2, Industry East Road IX, Science-based

Industrial Park, Hsinchu, 300, Taiwan, R.O.C.

Tel: 886-3-5780211 Fax: 886-3-5776047

http://www.realtek.com.tw/

Change History

Version	Editor	Date	Remarks
0.001	MH	2009/01/09	1. Merge from previoud DLL_Function_Specification.doc 2. Add new API for 92S series. 3. Add efuse operation API in chapter 4.
0.002	Cosa	2009/02/26	1. Add dll function for 92S
0.003	chiyoko	2011/08/19	1. For 8723A
0.004	chiyoko	2011/10/21	1. Revise BT Efuse API
0.005	chiyoko	2011/11/10	1. For 8188E
0.006	MarkFeng	2017/03/06	1. For MP Win7 ver. API

Table of Contents:

List of Figure

List of Table

1. MP DLL Introduction

1.1. Build Environment

We recommend Visual Studio® 2012 as build environment, because newer versions of Visual Studio® may have compatibility issues. Visual Studio® 2012 test build ok, project setting configuration is required.

Visual Studio® Project Setting (based on Visual Studio® 2012):

- Enable C++ Exceptions : No
- Security Check : No
- Character Set : Use Multi-Byte
- Enable Function Level Linking : No
- Enable Incremental Linking : No
- Additional Dependencies : msvcrt.lib setupapi.lib MP_DLL.lib

1.2. DLL Table

Table 1. DLL API Name Table

Direction	API Name	Remark
General		
	StartDUT	
	CloseDUT	
	StartTest	
	StopTest	
	SetModulation	
	SetAntennaBB	
	SetDataRate	
	SetChannelDirectCall	
	SetTxContinuousDirectCall	
	SetCarrierSuppressionTxContinuous	
	SetPreamble	
	SetTxPowerControl	
	ReadMACAddress	
	WriteMACAddress	
	QueryChipIDandVer	
	QuerySignalLocationType	
	SetSignalLocationType	
	ReadRFThermalMeter	
	DLL_SwitchTxPwrTrack	
	DLL_QueryTxPwrTrack	
	DLL_CheckAutloadStatus	
	DLL_ReadEFuseUtilization	
	LEDControl	
	ButtonPushed	
	QueryRfStatus	
	GetCalTxPwrIndex	

	SetRfPathSwitch	
	GetRfPathSwitch	
	SetRFOnOff	
	SetCrystalCap	
EFUSE		
	~~**ReadEFuse**~~	**Deprecated**
	ReadEFuseByte	
	~~**WriteEFuse**~~	**Deprecated**
	WriteEFuseByte	
	UpdateEFuse	
Packet TX/RX		
	ResetTxPacketSent	
	QueryTxPacketSent	
	ResetRxPacketReceived	
	QueryRxPacketReceived	
	QueryRxPacketCRC32Error	
	DLL_SendSpecificPacket	

2. General Functions

- **PVOID StartDUT(UINT32 ChipID,**
  UINT32 ChipVersion)

  This function opens the connection with specific DUT and get the handle of the DUT. It will return a NULL pointer if specifed ***ChipID*** and ***ChipVersion*** is not found, otherwise, it will allocate memory for a **ADAPTER** object and return a pointer to this object. Other function who needs the argumenent, LPADAPTER AdapterObject, shall use the pointer returned by **StartDUT()**.

  <Note>
  1. It should be invoked before other API needs the input argument, LPADAPTER AdapterObject.
  2. Open connection with specific DUT with specified ***ChipID*** and ***ChipVersion***

  <Example>

```
// For RTL8188EU
UINT32 ChipID = 0x8188
UINT32 ChipVersion = 0x2 //PCIE: 0x1, USB: 0x2, SDIO: 0x3
if(StartDUT(ChipID, ChipVersion) == NULL)
{
  MP_QUIT();
}
```

- **INT CloseDUT(LPADAPTER AdapterObject)**
  This function closes the connection with DUT and release the handle and memory block allocated in **OpenDUT()** for the DUT. It will return TRUE/FALSE to show operation success or fail.

  <Note>
  1. It MUST be invoked when we don't need to access the AdapterObject any more before terminating the calling program. Otherwise, a memory leak problem will be encounted.

  <Example>

```
//
// wchar wszAdapterName = L"\Device\mp8185s_{08E0D84D-1B31-4D32-93B2-0AE05C90192D}";
//
LPADAPTER pAdapter =OpenDUT(wszAdapterName);
StartTest(pAdapter);
StopTest(pAdapter);
CloseDUT(pAdapter);
```

- **INT StartTest(LPADAPTER AdaterObject)**
  This function initializes the DUT to the MP test mode. It will return TRUE/FALSE to show operation success or fail.

  <Note>
  1.We SHOULD invoked this function before other Mass Production related test item, for example, Packet Tx test, Packet Rx test, Continuous Tx test, Single Carrier Tx test, Carrier Suppression Tx test and so on .

- **INT StopTest(LPADAPTER AdapterObject)**
  This function change the DUT from the test mode into normal mode. It will return TRUE/FALSE to show operation success or fail.

  . **<Note>**
  1.This function MUST be invoked if StartTest() is had been called before.

- **INT SetModulation(LPADAPTER AdapterObject,**
  INT Modulation)

  This function sets the wireless bandwidth, such as 2.4G or 5G. It will return TRUE/FALSE to show operation success or fail..

  <Note>
  The modulation selection is :
  1. WIRELESS_MODE_N_24G,
  2. WIRELESS_MODE_N_5G,
  3. WIRELESS_MODE_AC_5G.

- **INT SetAntennaBB(LPADAPTER AdapterObject,**
  INT Ant)

  This function sets RF path with parameter **Ant**. The **Ant**'s higher 2-Bytes are indicated as TxPath value. The **Ant**'s lower 2-Bytes are indicated as RxPath value. It will return TRUE/FALSE to show operation success or fail..

  <Example>
  Ant = (**AntennaTx** <<16) | **AntennaRx** ;

- **INT SetDataRate(LPADAPTER AdapterObject, int Rate)**
  This function sets the transmit data rate of the DUT. It will return TRUE/FALSE to show operation success or fail.

  <Note>
  Available data rates are:
  1 : CCK 1M
  2 : CCK 2M
  3 : CCK 5.5M
  4 : CCK 11M
  5 : OFDM 6M
  6 : OFDM 9M
  7 : OFDM 12M
  8 : OFDM 18M
  9 : OFDM 24M
  10 : OFDM 36M
  11 : OFDM 48M
  12 : OFDM 54M
  13~28 : MCS0~MCS15

- **INT SetChannelDirectCall(LPADAPTER AdapterObject, int Channel)**
  This function sets the DUT to transmit and receive on a specific channel. It will return TRUE/FALSE to show operation success or fail..

- **INT SetTxContinuousDirectCall(LPADAPTER AdapterObject, int Mode)**
  This functions starts(Mode: 1) or stops(Mode: 0) the DUT to perform continuous transmission. It will return TRUE/FALSE to show operation success or fail.

- **INT SetCarrierSuppressionTxContinuous(LPADAPTER AdapterObject, int Mode)**
  This functions starts(Mode: 1) or stops(Mode: 0) the DUT to perform carrier suppression continuous transmission. It will return TRUE/FALSE to show operation success or fail.

- **INT SetPreamble(LPADAPTER AdapterObject, int Mode)**
  This function sets the DUT to use long preamble(Mode: 1) or short preamble(Mode: 2), or Long GI(Mode: 3), or Short GI(Mode: 4). It will return TRUE/FALSE to show operation success or fail.

- **INT SetTxPowerControl(LPADAPTER AdapterObject, int Value)**
  This function sets the DUT's TX POWER CONTROL of the current channel to the desired value, that is, it changes current setting of hardware, and corresponding field in EEPROM is not changed in this function. It will return TRUE/FALSE to show operation success or fail.

- **INT ReadMACAddress(LPADAPTER AdapterObject, char* MACAddress)**
  This function reads the MAC address from the DUT MAC registers. It will return TRUE/FALSE to show operation success or fail.

  <Note>
  .This function would not actually R/W EFUSE. The R/W EFUSE functions with specific offset can be found in the Chapter 3 EFUSE Operation Functions.

- **INT WriteMACAddress(LPADAPTER AdapterObject, char* MACAddress)**
  This function reads the MAC address from the DUT MAC registers. It will return TRUE/FALSE to show operation success or fail.

  <Note>
  This function would not actually R/W EFUSE. The R/W EFUSE functions with specific offset can be found in the Chapter 3 EFUSE Operation Functions.

- **INT QueryChipIDandVer(LPADAPTER AdapterObject)**
  This function queries the Chip ID and Version from the driver. It will return TRUE/FALSE to show operation success or fail.

- **INT QuerySignalLocationType (LPADAPTER AdapterObject, int * SignalLocation)**
  This function queries the Signal Location, 20M(Mode 0), 40MHZ_DUPLICATED(Mode 1), 40MHZ_LOWER(Mode 2), 40MHZ_UPPER(Mode 3), 40MHZ_DSC(Mode 4). It will return TRUE/FALSE to show operation success or fail.

- **INT SetSignalLocationType (LPADAPTER AdapterObject, int SignalLocation)**
  This function sets the Signal Location, 20M(Mode 0), 40MHZ_DUPLICATED(Mode 1), 40MHZ_LOWER(Mode 2), 40MHZ_UPPER(Mode 3), 40MHZ_FULL(Mode 4). It will return TRUE/FALSE to show operation success or fail. **(NOTE: only 20M(Mode 0) and 40M(Mode 4) are supported now.**

- **INT ReadRFThermalMeter (LPADAPTER AdapterObject, unsigned int *ThermalValue)**
  This routine reads back the Thermal Meter value. Bit0~15 is the value of RF-A and bit16~31 is the value of RF-C.

- **INT DLL_SwitchTxPwrTrack(LPADAPTER AdapterObject, unsigned char TxPwrTrackState)**
  This routine switch Tx Power Track state. It will return TRUE/FALSE to show operation success or fail.

- **INT DLL_QueryTxPwrTrack (LPADAPTER AdapterObject, int *TxPwrTrackState)**
  This routine query Tx Power Track state. It will return TRUE/FALSE to show operation success or fail.

- **INT DLL_CheckAutloadStatus(LPADAPTER AdapterObject, unsigned char *AutoloadFail)**
  This routine check autoload status fail or success. 1=Autoload Fail, 0=Autoload Success.It will return TRUE/FALSE to show operation success or fail.

- **INT DLL_ReadEFuseUtilization(LPADAPTER AdapterObject, unsigned int *EFuseUtilize)**

This routine get the EFUSE utilization, bit0~15 for how many bytes used, bit 16~31 for percentage. It will return TRUE/FALSE to show operation success or fail.

- **BOOLEAN LEDControl(LPADAPTER AdapterObject, int led_num, int led_mode)**
  This routine sets the led_num to light as the led_mode. Led_num=0~1, LED OFF:set led_mode=0, LED ON: set led_mode=1. It will return TRUE/FALSE to show operation success or fail.

- **BOOLEAN ButtonPushed(LPADAPTER AdapterObject, int buttonType, int *pushed)**
  This routine returns the button pushed or not. WPS Button: buttonType=0. pushed=1 if the button is pushed, pushed=0 means the button is not pushed. It will return TRUE/FALSE to show operation success or fail.

- **BOOLEAN QueryRfStatus(LPADAPTER AdapterObject, int *rfstatus)**
  This routine returns the HW RF on/off status. rfstatus=2 if HW RF off, pushed=0 means HW RF on. It will return TRUE/FALSE to show operation success or fail.

- **BOOLEAN GetCalTxPwrIndex(LPADAPTER AdapterObject, int *powerIndex)**
  This routine returns the real calculated Tx power index by current channel and rate in driver config. It will return TRUE/FALSE to show operation success or fail.

- **INT SetRfPAthSwitch(LPADAPTER AdapterObject, BOOL bMain)**
  This routine sets the RF path to Main or Aux. bMain = TRUE means set RF path to Main, bMain = FALSE means set RF path to Aux. It will return TRUE/FALSE to show operation success or fail.

- **INT GetRfPAthSwitch(LPADAPTER AdapterObject, BOOL *bMain)**
  This routine gets the RF path setting. bMain = TRUE means RF path is set to Main, bMain = FALSE means RF path is set to Aux. It will return TRUE/FALSE to show operation success or fail.

- **BOOLEAN SetRFOnOff(LPADAPTER AdapterObject, int RfOnOff)**
  This routine set RF on/off (//0: on, 1: off). Set Rf off before start BT test and set RF on after BT test finished.

- **BOOLEAN SetCrystalCap(LPADAPTER AdapterObject, int CrystalCap)**
  This routine set Crystal Cap value (0x00~0x3F). It will return TRUE/FALSE to show operation success or fail.

<Recommend>

The following sequence is a suggestion that RealTek MP Utility used RealTek DLL API function.

1. **SetModulation**
2. **SetAntennaBB**
3. **SetSignalLocationType**
4. **SetChannelDirectCall**
5. **SetDataRate**
6. **SetTxPowerControl**

3. EFUSE Operation Functions

- **int ReadEFuse(LPADAPTER AdapterObject,**
  char* EFuse)

  (Deprecated) This function support to read EFUSE shadow memory from NIC side. You can not read real Efuse content. Driver will assist to protect you to execute incorrect operation on EFUSE area.

- **int ReadEFuseByte(LPADAPTER AdapterObject,**
  UCHAR Offset,
  UCHAR* Value)

  This function reads the value of the byte offset specified from EFUSE shadow memory. It will return TRUE/FALSE to show operation success or fail

  <Note>
  For Wifi/BT combo IC : ReadBTEFuseByte

- **int WriteEFuse(LPADAPTER AdapterObject,**
  UCHAR *EFuse)

  (Deprecated) This function reads allow user to send specific number of bytes Efuse table to driver. At the time, driver only save the content in shadow memory.

- **int WriteEFuseByte(LPADAPTER AdapterObject,**
  UCHAR offset,
  UCHAR Value)

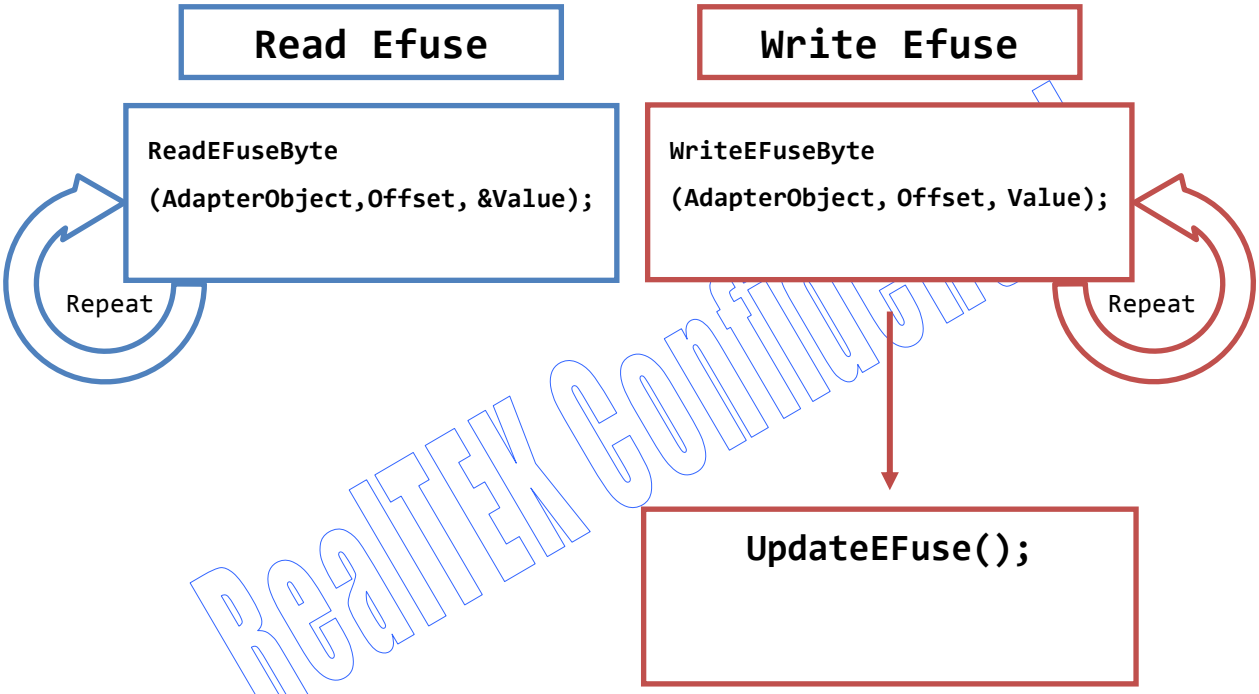
  This function writes 1-byte value into the byte offset of EFUSE shadow memory. It will return TRUE/FALSE to show operation success or fail.
  <Note>
  For Wifi/BT combo IC : WriteBTEFuseByte

- **int UpdateEFuse (LPADAPTER AdapterObject)**

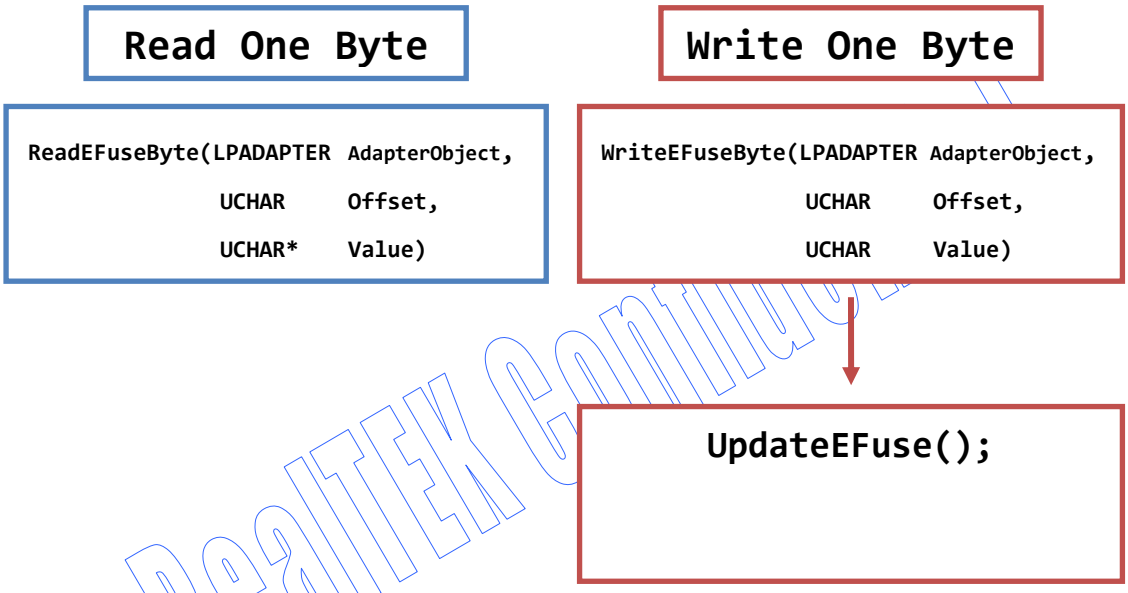
  This function allow user to update previous content in shadow memory to real EFUSE area. Driver will automatically compare shadow map and logical map. And we only write different area into efuse to prevent from wasting EFuse space.
  <Note>
  For Wifi/BT combo IC : UpdateBTEFuse

4. A Better EFUSE Read/Write Method

4.1. Read/Write Whole Efuse



4.2. Read/Write One Byte of Efuse



4.3. WriteEfuse/ReadEfuse are Deprecated

Because the arguments of WriteEfuse/ReadEfuse don't include the Efuse length, there is a vulnerability that the invalid memory might be accessed and cause UI crash.

Either for reading/writing a full map or one byte, we could implement the methods by repeatedly calling WriteEfuseByte/ReadEfuseByte APIs instead. We recommend using WriteEfuseByte/ReadEfuseByte directly for better flexibility.

5. Packet TX/RX Functions

- **INT ResetTxPacketSent (LPADAPTER AdapterObject)**
  This function resets the packet amount transmitted. It will return TRUE/FALSE to show operation success or fail.

- **INT QueryTxPacketSent (LPADAPTER AdapterObject,**
  INT * TxPacketCount)
  This function queries for the packet amount transmitted. It will return TRUE/FALSE to show operation success or fail.

- **INT ResetRxPacketReceived (LPADAPTER AdapterObject)**
  This function resets the number of packet received successfully and the number of packet received with CRC32 error. It will return TRUE/FALSE to show operation success or fail.

- **INT QueryRxPacketReceived (LPADAPTER AdapterObject,**
  INT * RxPacketCount)
  This function queries for the number of packet received successfully. It will return TRUE/FALSE to show operation success or fail.

- **INT QueryRxPacketCRC32Error (LPADAPTER AdapterObject,**
  INT * RxPacketCRC32Error)
  This function queries for the number of packet received with CRC32 error. It will return TRUE/FALSE to show operation success of fail.

- **BOOLEAN DLL_SendSpecificPacket(LPADAPTER AdapterObject,**
  PVOID PktBuffer,
  USHORT PktLength,
  PVOID Reserved)
  This function sends out a packet with **PktBuffer** and **PktLength**. It will return TRUE/FALSE to show operation success or fail.

<Recommend>

During the test of Tx packet, do not test Rx packet at the same time.