

# I7565DNM Linux Software Manual

---

## User Manual

### **Warranty**

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

### **Warning**

ICP DAS assume no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

### **Copyright**

Copyright 2012 by ICP DAS. All rights are reserved.

### **Trademark**

The names used for identification only may be registered trademarks of their respective companies.

# Tables of Contents

<b>1. i-7565-DNM Linux Driver Installation .....</b>	<b>4</b>
1.1 Linux Driver Installing Procedure .....	4
<b>2. i-7565-DNM Static Library Function Description.....</b>	<b>5</b>
2.1 Table of ErrorCode and ErrorString .....	5
2.2 Function Descriptions .....	7
2.3 Module Initial Functions .....	9
2.3.1 I7565DNM_ActiveModule .....	9
2.3.2 I7565DNM_CloseModule .....	10
2.3.3 I7565DNM__GetLibraryVersion .....	10
2.4 Firmware Functions .....	10
2.4.1 I7565DNM_ResetFirmware .....	10
2.4.2 I7565DNM_GetFirmwareVersion .....	11
2.5 Master Operation Functions.....	11
2.5.1 I7565DNM_GetMasterMACID.....	11
2.5.2 I7565DNM_SetMasterMACID.....	12
2.5.3 I7565DNM_GetBaudRate .....	12
2.5.4 I7565DNM_SetBaudRate.....	12
2.5.5 I7565DNM_GetMasterStatus .....	13
2.5.6 I7565DNM_GetSlaveStatus .....	13
2.5.7 I7565DNM_StartDevice.....	14
2.5.8 I7565DNM_StopDevice .....	14
2.5.9 I7565DNM_StartAllDevice .....	14
2.5.10 I7565DNM_StopAllDevice.....	15
2.5.11 I7565DNM_AddDevice .....	15
2.5.12 I7565DNM_RemoveDevice.....	15
2.5.13 I7565DNM_AddIOConnection .....	16
2.5.14 I7565DNM_RemoveIOConnection .....	17
2.5.15 I7565DNM_GetAttribute .....	17
2.5.16 I7565DNM_IsGetAttributeOK.....	18
2.5.17 I7565DNM_GetAttributeValue .....	18
2.5.18 I7565DNM_SetAttribute .....	19
2.5.19 I7565DNM_IsSetAttributeOK.....	19
2.5.20 I7565DNM_ClearAllConfig.....	20
2.5.21 I7565DNM_SearchAllDevices .....	20
2.5.22 I7565DNM_SearchSpecificDevice.....	20

2.5.23	<i>I7565DNM_IsSearchOK</i> .....	21
2.5.24	<i>I7565DNM_GetSearchedDevices</i> .....	21
2.5.25	<i>I7565DNM_GetDeviceInfoFromScanList</i> .....	22
2.5.26	<i>I7565DNM_GetScanList</i> .....	22
2.5.27	<i>I7565DNM_ImportEEPROM</i> .....	23
2.5.28	<i>I7565DNM_SendExplicitMSG</i> .....	24
2.5.29	<i>I7565DNM_IsExplicitMSGRespOK</i> .....	25
2.5.30	<i>I7565DNM_GetExplicitMSGRespValue</i> .....	25
2.6	I/O Configuration Functions .....	26
2.6.1	<i>I7565DNM_ReadInputData</i> .....	26
2.6.2	<i>I7565DNM_WriteOutputData</i> .....	26
2.6.3	<i>I7565DNM_ReadbackOutputData</i> .....	27
<b>3.</b>	<b>i-7565-DNM Demo Programs For Linux .....</b>	<b>28</b>
3.1	Demo code “i7565DNM” .....	28

---

## 1. i-7565-DNM Linux Driver Installation

The I-7565-DNM can be used in linux. For Linux O.S, the recommended installation and uninstall steps are given in Sec 1.1 ~ 1.2

---

### 1.1 Linux Driver Installing Procedure

#### (1) Linux PC(x86)

- Most Linux had supported the i-7565-DNM driver module “pl2303”. After user install i-7565-DNM hardware, Linux OS would install driver automatically. Then user can type command “dmesg” to check I-7565-DNM device file(please refer to Figure 1-1)

**#dmesg**

```
pl2303 ttyUSB0: pl2303 converter now disconnected from ttyUSB0
pl2303 3-1:1.0: device disconnected
usb 3-1: new full speed USB device using uhci_hcd and address 3
usb 3-1: New USB device found, idVendor=067b, idProduct=2303
usb 3-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
usb 3-1: Product: USB-Serial Controller
usb 3-1: Manufacturer: Prolific Technology Inc.
usb 3-1: SerialNumber: 00003A87
usb 3-1: configuration #1 chosen from 1 choice
pl2303 3-1:1.0: pl2303 converter detected
usb 3-1: pl2303 converter now attached to ttyUSB0
[root@localhost ~]#
```

Figure 1-1

---

## 2. i-7565-DNM Static Library Function Description

The static library is the collection of function calls of the i-7565-DNM for linux kernel 2.6.x system. The application structure is presented as below figure “Figure 2-1”. The user application program developed by C (C++) language can call library “libI7565DNM.a” for x86 Linux OS. And then static library will call the module command to access the hardware system.

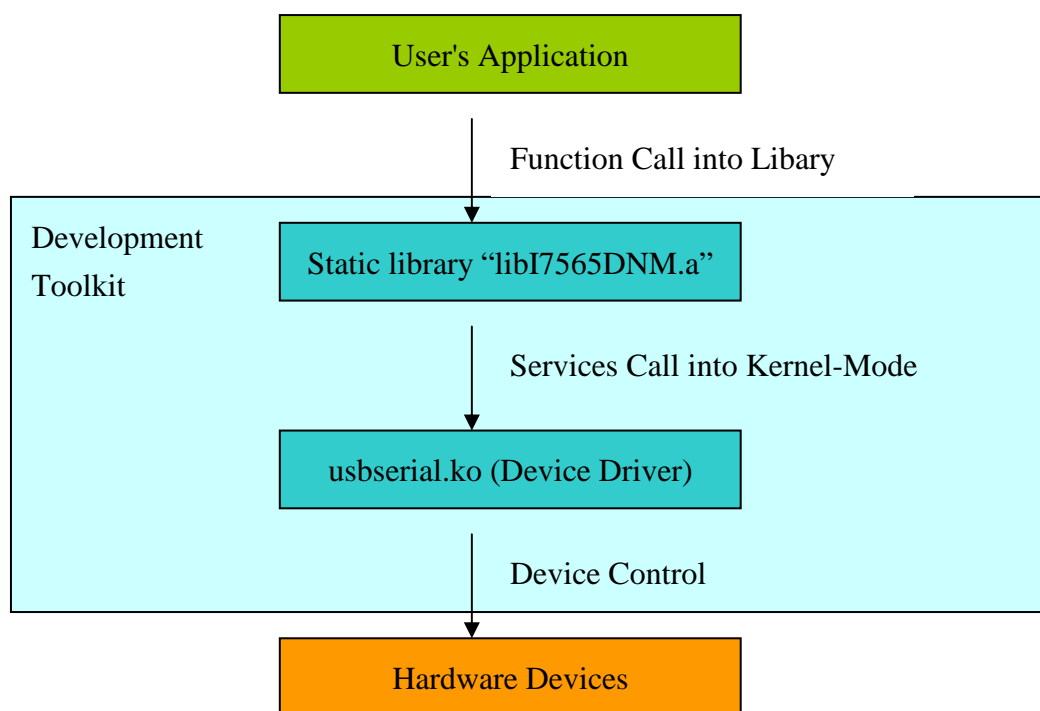


Figure 2-1

---

### 2.1 Table of ErrorCode and ErrorString

Table 2.1

Error Code	Error ID	Error String
0	I7565DNM_NoErr	OK ( No error !)
10001	I7565DNM_MasterNoResp	Master has no any response
10002	I7565DNM_SlaveNoResp	Slave has no any response
10003	I7565DNM_DevPortErr	The device file doesn't exist

Error Code	Error ID	Error String
10004	I7565DNM_MACIDErr	The MAC ID over range(0~63)
10005	I7565DNM_BaudRateErr	The baud rate over range(0~2)
10006	I7565DNM_ConnectionTypeErr	The DeviceNet connection's type over range(0~4)
10007	I7565DNM_ModNotExist	The module doesn't exist
10008	I7565DNM_PortNotOpen	The port doesn't open
10009	I7565DNM_PortCloseErr	The port doesn't close
10010	I7565DNM_ResetFirmwareErr	To reset firmware error
10011	I7565DNM_GetFirmwareVerErr	To get the firmware's version error
10012	I7565DNM_SetMasterMACIDErr	To set the master's MAC ID error
10013	I7565DNM_GetMasterMACIDErr	To get the master's MAC ID error
10014	I7565DNM_SetBaudRateErr	To set the master's baud error
10015	I7565DNM_GetBaudRateErr	To get the master's baud error
10016	I7565DNM_AddDeviceErr	The master add slave error
10017	I7565DNM_RemoveDeviceErr	The master remove slave error
10018	I7565DNM_AddIOConnectionErr	To add I/O Connection error
10019	I7565DNM_RemoveIOConnectionErr	To remove I/O Connection error
10020	I7565DNM_StartDeviceErr	To start device error
10021	I7565DNM_StopDeviceErr	To stop device error
10022	I7565DNM_StartAllDeviceErr	To start all devices error
10023	I7565DNM_StopAllDeviceErr	To stop all devices error
10024	I7565DNM_GetAttributeErr	The master get slave's attribute error
10025	I7565DNM_GetAttributeValueErr	To get attribute's value error
10026	I7565DNM_SetAttributeErr	To set attribute error
10027	I7565DNM_ReadInputDataErr	To read the input's data error
10028	I7565DNM_WriteOutputDataErr	To write the output's data error
10029	I7565DNM_GetSlaveStatusErr	To get slave's status error
10030	I7565DNM_SearchAllDevicesErr	To search all devices error
10031	I7565DNM_SearchSpecificDeviceErr	To search specific device error

Error Code	Error ID	Error String
10032	I7565DNM_IsSearchOKErr	To check the searching result failure
10033	I7565DNM_GetSearchedDevicesErr	To get the searching data error
10034	I7565DNM_GetMasterStatusErr	To get master's status error
10035	I7565DNM_GetDeviceInfoErr	To get device's information error
10036	I7565DNM_GetLastDeviceErr	To get last device's information error
10037	I7565DNM_GetScanListErr	To get the scanning list error
10038	I7565DNM_ImportEEPROMErr	To import data to EEPROM error
10039	I7565DNM_ClearAllConfigErr	To clear all configuration error
10040	I7565DNM_SendExplicitMSGErr	To send explicit message error
10041	I7565DNM_IsExplicitMSGRespOKErr	To check the result of sending explicit message failure
10042	I7565DNM_GetExplicitMSGRespValueErr	To get the return value of explicit message error
10043	I7565DNM_ReadbackOutputDataErr	To read the output data error

## 2.2 Function Descriptions

Table 2.2

NO	Module Initial Functions
1	DWORD I7565DNM_ActiveModule(BYTE DevPort)
2	DWORD I7565DNM_CloseModule(BYTE DevPort)
3	char * I7565DNM_GetLibraryVersion(void)
No	Firmware Functions
1	DWORD I7565DNM_ResetFirmware(BYTE DevPort)
2	DWORD I7565DNM_GetFirmwareVersion(BYTE DevPort)
NO	Master Operation Functions
1	DWORD I7565DNM_GetMasterMACID(BYTE DevPort)
2	DWORD I7565DNM_SetMasterMACID(BYTE DevPort, BYTE MasterMACID)

NO	Master Operation Functions
3	DWORD I7565DNM_GetBaudRate(BYTE DevPort)
4	DWORD I7565DNM_SetBaudRate(BYTE DevPort, BYTE BaudRate)
5	DWORD I7565DNM_GetMasterStatus(BYTE DevPort)
6	DWORD I7565DNM_GetSlaveStatus(BYTE DevPort, BYTE DesMACID)
7	DWORD I7565DNM_StartDevice(BYTE DevPort, BYTE DesMACID)
8	DWORD I7565DNM_StopDevice(BYTE DevPort, BYTE DesMACID)
9	DWORD I7565DNM_StartAllDevice(BYTE DevPort)
10	DWORD I7565DNM_StopAllDevice(BYTE DevPort)
11	DWORD I7565DNM_AddDevice(BYTE DevPort, BYTE DesMACID, WORD Explicit_EPR)
12	DWORD I7565DNM_RemoveDevice(BYTE DevPort, BYTE DesMACID)
13	DWORD I7565DNM_AddIOConnection(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD DeviceInputLen, WORD DeviceOutputLen, WORD EPR)
14	DWORD I7565DNM_RemoveIOConnection(BYTE DevPort, BYTE DesMACID, BYTE ConType)
15	DWORD I7565DNM_GetAttribute(BYTE DevPort, BYTE DesMACID, BYTE ClassID, BYTE InstanceID, BYTE AttributeID)
17	DWORD I7565DNM_GetAttributeValue(BYTE DevPort, BYTE DesMACID, WORD *DataLen, BYTE *DATA)
16	DWORD I7565DNM_IsGetAttributeOK(BYTE DevPort, BYTE DesMACID)
18	DWORD I7565DNM_SetAttribute(BYTE DevPort, BYTE DesMACID, BYTE ClassID, BYTE InstanceID, BYTE AttributeID, WORD DataLen, BYTE *DATA)
19	DWORD I7565DNM_IsSetAttributeOK(BYTE DevPort, BYTE DesMACID)
20	DWORD I7565DNM_ClearAllConfig(BYTE DevPort)
21	DWORD I7565DNM_SearchAllDevices(BYTE DevPort)
22	DWORD I7565DNM_SearchSpecificDevice(BYTE DevPort, WORD ListCount, BYTE *DesMACIDList)
23	DWORD I7565DNM_IsSearchOK(BYTE DevPort)



NO	Master Operation Functions
24	DWORD I7565DNM_GetSearchedDevices(BYTE DevPort, WORD *TotalDevices, BYTE *DesMACID, BYTE *Type, WORD *DeviceInputLen, WORD *DeviceOutputLen)
25	DWORD I7565DNM_GetDeviceInfoFromScanList(BYTE DevPort, BYTE DesMACID, WORD *ListCount, BYTE *ConnectionTypeList, WORD *InputDataLenList, WORD *OutputDataLenList, WORD *EPRList)
26	DWORD I7565DNM_GetScanList(BYTE DevPort, WORD *TotalDevices, BYTE *DesMACIDList, BYTE *ConnectionTypeList, WORD *InputDataLenList, WORD *OutputDataLenList, WORD *EPR_List)
27	DWORD I7565DNM_ImportEEPROM(BYTE DevPort, WORD ListCount, BYTE *DesMACIDList, BYTE *ConnectionTypeList, WORD *InputDataLenList, WORD *OutputDataLenList, WORD *EPR_List)
28	DWORD I7565DNM_SendExplicitMSG(BYTE DevPort, BYTE DesMACID, BYTE ServiceID, BYTE ClassID, BYTE InstanceID, WORD DataLen, BYTE *DATA)
29	DWORD I7565DNM_IsExplicitMSGRespOK(BYTE DevPort, BYTE DesMACID);
30	DWORD I7565DNM_GetExplicitMSGRespValue(BYTE DevPort, BYTE DesMACID, WORD *DataLen, BYTE *DATA)
NO	I/O Configuration Functions
1	DWORD I7565DNM_ReadInputData(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD *IOLen, BYTE *IODATA)
2	DWORD I7565DNM_WriteOutputData(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD IOLen, BYTE *IODATA)
3	DWORD I7565DNM_ReadbackOutputData(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD *IOLen, BYTE *IODATA)

---

## 2.3 Module Initial Functions

---

### 2.3.1 I7565DNM\_ActiveModule

- **Description:**

The function is used to activate the I-7565-DNM module. It must be called once before using the other functions of I-7565-DNM APIs.

- **Syntax:**  
DWORD I7565DNM\_ActiveModule(BYTE DevPort);
  - **Parameter:**  
DevPort : [input] The USB port number.
  - **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").
- 

### 2.3.2 I7565DNM\_CloseModule

- **Description:**  
The function is used to stop and close the USB driver. This method must be called once before exiting the user's application program.
  - **Syntax:**  
DWORD I7565DNM\_CloseModule(BYTE DevPort);
  - **Parameter:**  
DevPort : [input] The USB port number..
  - **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").
- 

### 2.3.3 I7565DNM\_\_GetLibraryVersion

- **Description:**  
To get the version of VCI\_CAN library.
  - **Syntax:**  
char \* I7565DNM\_\_GetLibraryVersion(void);
  - **Parameter:**  
None
  - **Return:**  
Return the I-7565-DNM library version.
- 

## 2.4 Firmware Functions

---

### 2.4.1 I7565DNM\_ResetFirmware

- **Description:**  
If uses changed the baud 、 MAC ID , user must call the function to

make the change enable. After calling the function , user must wait for 1 or 2 seconds to make the firmware reboot completely.

- **Syntax:**  
DWORD I7565DNM\_ResetFirmware(BYTE DevPort);
  - **Parameter:**  
DevPort : [input] The USB port number.
  - **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").
- 

#### 2.4.2 I7565DNM\_GetFirmwareVersion

- **Description :**  
The function can get the firmware's version.
  - **Syntax :**  
DWORD I7565DNM\_GetFirmwareVersion(BYTE DevPort);
  - **Parameter :**  
DevPort : [input] The USB port number.
  - **Return:**  
The firmware version information. For example: If 100(hex) is return, it means firmware version is 1.00. Others means failure (Please refer to "Section 2.1 Error Code").
- 

### 2.5 Master Operation Functions

---

#### 2.5.1 I7565DNM\_GetMasterMACID

- **Description :**  
The function can get the master's MAC ID.
- **Syntax :**  
DWORD I7565DNM\_GetMasterMACID(BYTE DevPort);
- **Parameter :**  
DevPort : [input] The USB port number.
- **Return:**  
The master's MAC ID, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.2 I7565DNM\_SetMasterMACID

- **Description :**  
The function can set the master's MAC ID. After calling this function, the users must call I7565DNM\_ResetFirmware to make the change enabled. It will save the information in the I-7565-DNM's EEPROM.
- **Syntax :**  
DWORD I7565DNM\_SetMasterMACID(BYTE DevPort, BYTE MasterMACID)
- **Parameter :**  
DevPort : [input] The USB port number.  
MasterMACID: [input] The new MAC ID of the master. (0 ~ 63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.3 I7565DNM\_GetBaudRate

- **Description :**  
The function can get the master's baud rate.
- **Syntax :**  
DWORD I7565DNM\_GetBaudRate(BYTE DevPort);
- **Parameter :**  
DevPort : [input] The USB port number.
- **Return:**  
The CAN bus baud rate information in the I-7565-DNM.  
If the value is 0, the baud rate is 125Kbps.  
If the value is 1, the baud rate is 250Kbps.  
If the value is 2, the baud rate is 500Kbps.  
Others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.4 I7565DNM\_SetBaudRate

- **Description :**  
This function can set the DeviceNet baud rate of the I-7565-DNM. After calling this function, user must call I7565DNM\_ResetFirmware to reset the firmware to make change enabled.
- **Syntax :**  
DWORD I7565DNM\_SetBaudRate(BYTE DevPort, BYTE BaudRate);

- **Parameter :**  
 DevPort : [input] The USB port number.  
 BaudRate: [input] The new baud rate value.  
 0 : 125K bps  
 1 : 250K bps  
 2 : 500K bps
  - **Return:**  
 Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").
- 

### 2.5.5 I7565DNM\_GetMasterStatus

- **Description :**  
 The function is used to obtain the firmware status inside the I-7565-DNM. The users can call this function to make sure that the DeviceNet master is online successfully.
  - **Syntax :**  
 DWORD I7565DNM\_GetMasterStatus(BYTE DevPort);
  - **Parameter :**  
 DevPort : [input] The USB port number.
  - **Return:**  
 Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").
- 

### 2.5.6 I7565DNM\_GetSlaveStatus

- **Description :**  
 This function is to get the remote slave device's communication status.
- **Syntax :**  
 DWORD I7565DNM\_GetSlaveStatus(BYTE DevPort, BYTE DesMACID);
- **Parameter :**  
 DevPort : [input] The USB port number.  
 DesMACID: [input] The remote slave's MAC ID. (0~63)
- **Return:**  
 Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.7 I7565DNM\_StartDevice

- **Description :**  
This function is used to start to communicate with the specific device that the users applying to.
- **Syntax :**  
DWORD I7565DNM\_StartDevice(BYTE DevPort, BYTE DesMACID);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.8 I7565DNM\_StopDevice

- **Description :**  
This function is used to stop to communicate with the destination device that the users appointed to.
- **Syntax :**  
DWORD I7565DNM\_StopDevice(BYTE DevPort, BYTE DesMACID);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.9 I7565DNM\_StartAllDevice

- **Description :**  
This function is used to start to communicate with all slave devices in the list.
- **Syntax :**  
DWORD I7565DNM\_StartAllDevice(BYTE DevPort);
- **Parameter :**  
DevPort : [input] The USB port number.
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section

2.1 Error Code").

---

### 2.5.10 I7565DNM\_StopAllDevice

- **Description :**

This function is used to stop to communicate with the all slave devices in the list.

- **Syntax :**

DWORD I7565DNM\_StopAllDevice(BYTE DevPort);

- **Parameter :**

DevPort : [input] The USB port number.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.11 I7565DNM\_AddDevice

- **Description :**

This function can add the slave devices into the ScanList of the I-7565-DNM and save the information into the EEPROM. Before communicating with any slave devices, the users should call this function to add these devices.

- **Syntax :**

DWORD I7565DNM\_AddDevice(BYTE DevPort, BYTE DesMACID, WORD Explicit\_EPR);

- **Parameter :**

DevPort : [input] The USB port number.

DesMACID: [input] The remote slave's MAC ID. (0~63)

Explicit\_EPR: [input] The Expected Packet Rate. (Usually is 2500).

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.12 I7565DNM\_RemoveDevice

- **Description :**

This function is used for removing the specified slave device from the ScanList in the I-7565-DNM. And the information of the device in

EEPROM is erased at the same time.

- **Syntax :**  
DWORD I7565DNM\_RemoveDevice(BYTE DevPort, BYTE DesMACID);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.13 I7565DNM\_AddIOConnection

- **Description :**  
This method is used to configure the I/O connection of the specific MAC ID device. This configuration data will be saved into EEPROM within the I-7565-DNM.
- **Syntax :**  
DWORD I7565DNM\_AddIOConnection(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD DeviceInputLen, WORD DeviceOutputLen, WORD EPR);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)  
ConType: [input] The remote slave device's I/O connection type  
    0 : Explicit connection type  
    1 : Poll connection type  
    2 : Bit-Strobe connection type  
    3 : COS connection type  
    4 : Cyclic connection type  
DeviceInputLen: [input] The remote slave device's input length. (Byte)  
DeviceOutputLen: [input] The remote slave device's output length. (Byte)  
EPR: [input] The expected packet rate. (mSec)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").



---

### 2.5.14 I7565DNM\_RemoveIOConnection

- **Description :**  
This method is used to remove the I/O connection of the specific MAC ID device.
- **Syntax :**  
DWORD I7565DNM\_RemoveIOConnection(BYTE DevPort, BYTE DesMACID, BYTE ConType);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)  
ConType: [input] The remote slave device's I/O connection type
  - 0 : Explicit connection type
  - 1 : Poll connection type
  - 2 : Bit-Strobe connection type
  - 3 : COS connection type
  - 4 : Cyclic connection type
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.15 I7565DNM\_GetAttribute

- **Description :**  
This function is used to send the request command to retrieve the attribute value of the specific device's instance. Before calling this function, you must start the device. After calling this function, you should execute the "I7565DNM\_GetAttributeValue" to get the response message returned from remote slave device.
- **Syntax :**  
DWORD I7565DNM\_GetAttribute(BYTE DevPort, BYTE DesMACID, BYTE ClassID, BYTE InstanceID, BYTE AttributeID);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)  
ClassID: [input] The remote slave device's ClassID  
InstanceID: [input] The remote slave device's InstanceID  
AttributeID: [input] The remote slave device's AttributeID
- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.16 I7565DNM\_IsGetAttributeOK

- **Description :**  
This function is used to check whether the I-7565-DNM has received the response message or not. After checking the response message, you should execute the "I7565DNM\_GetAttributeValue" to get the response message returned from remote slave device.
- **Syntax :**  
DWORD I7565DNM\_GetAttribute(BYTE DevPort, BYTE DesMACID);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.17 I7565DNM\_GetAttributeValue

- **Description :**  
This function is used to get the attribute value of the specific device's instance from the remote slave device. Before calling this function, the users should call I7565DNM\_GetAttribute to send request command first.
- **Syntax :**  
DWORD I7565DNM\_GetAttributeValue(BYTE DevPort, BYTE DesMACID, WORD \*DataLen, BYTE \*DATA);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)  
DataLen: [output] The length of the attribute value (in byte).  
DATA: [output] The attribute value that returned from the slave device.
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.18 I7565DNM\_SetAttribute

- **Description :**  
The method is used to set the attribute of the specific device's instance. Before calling this function, you must start the device. After calling this function, you should execute the "I7565DNM\_IsSetAttributeOK" to check the response message returned from the remote slave device.
- **Syntax :**  
DWORD I7565DNM\_SetAttribute(BYTE DevPort, BYTE DesMACID, BYTE ClassID, BYTE InstanceID, BYTE AttributeID, WORD DataLen, BYTE \*DATA);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)  
ClassID: [input] The remote slave device's ClassID  
InstanceID: [input] The remote slave device's InstanceID  
AttributeID: [input] The remote slave device's AttributeID  
DataLen: [input] The length of the attribute value (in byte).  
DATA: [input] The attribute value that the users want to send.
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.19 I7565DNM\_IsSetAttributeOK

- **Description :**  
This function is used to get the response value after executing the "I7565DNM\_SetAttribute" function.
- **Syntax :**  
DWORD I7565DNM\_IsSetAttribute(BYTE DevPort, BYTE DesMACID);
- **Parameter :**  
DevPort : [input] The USB port number.  
DesMACID: [input] The remote slave's MAC ID. (0~63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.20 I7565DNM\_ClearAllConfig

- **Description :**  
This function will clear all configurations in the EEPROM of the I-7565-DNM.
- **Syntax :**  
DWORD I7565DNM\_ClearAllConfig(BYTE DevPort);
- **Parameter :**  
DevPort : [input] The USB port number.
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.21 I7565DNM\_SearchAllDevices

- **Description :**  
This function is used to retrieve all devices in DeviceNet network. Attention! This function will terminate all communications with remote devices. This function is usually used for developing or debugging applications.
- **Syntax :**  
DWORD I7565DNM\_SearchAllDevices(BYTE DevPort);
- **Parameter :**  
DevPort : [input] The USB port number.
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.22 I7565DNM\_SearchSpecificDevice

- **Description :**  
This function is used to retrieve some devices which specified by the users. Attention! This function will terminate all communications with remote devices. This function is usually used for developing or debugging applications.
- **Syntax :**  
DWORD I7565DNM\_SearchSpecificDevice(BYTE DevPort, WORD ListCount, BYTE \*DesMACIDList);
- **Parameter :**

DevPort : [input] The USB port number.

ListCount: [input] The amount of the slave's ID.

DestMACIDList: [input] The list of all slave's MAC ID.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.23 I7565DNM\_IsSearchOK

- **Description :**

This function will check whether the searching process has finished or not.

- **Syntax :**

DWORD I7565DNM\_IsSearchOK(BYTE DevPort);

- **Parameter :**

DevPort : [input] The USB port number.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.24 I7565DNM\_GetSearchedDevices

- **Description :**

This function will get the device which have been searched in the network.

- **Syntax :**

DWORD I7565DNM\_GetSearchedDevices(BYTE DevPort, WORD \*TotalDevices, BYTE \*DesMACID, BYTE \*Type, WORD \*DeviceInputLen, WORD \*DeviceOutputLen);

- **Parameter :**

DevPort : [input] The USB port number.

TotalDevices: [output] The amount of all slave device which are found.

DesMACID: [output] The list of slave's MAC ID which are found.

Type: [output] The list of slave's connection type which are found.

0 : Explicit connection type

1 : Poll connection type

2 : Bit-Strobe connection type

3 : COS connection type

4 : Cyclic connection type

DeviceInputLen: [output] The list of slave's input length which are found.

DeviceOutputLen: [output] The list of slave's output length which are found.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.25 I7565DNM\_GetDeviceInfoFromScanList

- **Description :**

This function will get the ScanList data of certain device in the I-7565-DNM.

- **Syntax :**

DWORD I7565DNM\_GetDeviceInfoFromScanList(BYTE DevPort, BYTE DesMACID, WORD \*ListCount, BYTE \*ConnectionTypeList, WORD \*InputDataLenList, WORD \*OutputDataLenList, WORD \*EPRList);

- **Parameter :**

DevPort : [input] The USB port number.

DesMACID: [input] The MAC ID number.

ListCount: [output] The amount of all information items.

ConnectionTypeList: [output] The list of slave's connection type.

0 : Explicit connection type

1 : Poll connection type

2 : Bit-Strobe connection type

3 : COS connection type

4 : Cyclic connection type

InputDataLenList: [output] The list of slave's input length.

OutputDataLenList: [output] The list of slave's output length.

EPRList: [output] The list of slave's expected packet rate.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.26 I7565DNM\_GetScanList

- **Description :**

This function will get all the ScanList data in the I-7565-DNM.

- **Syntax :**

DWORD I7565DNM\_GetScanList(BYTE DevPort, WORD \*TotalDevices, BYTE \*DesMACIDList, BYTE \*ConnectionTypeList, WORD \*InputDataLenList, WORD \*OutputDataLenList, WORD \*EPR\_List);

- **Parameter :**

DevPort : [input] The USB port number.

TotalDevices: [output] The data count of all the information.

DestMACIDList: [output] The MAC ID of all the slave devices in the ScanList.

ConnectionTypeList: [output] The connection type of all the slave devices in the ScanList.

0 : Explicit connection type

1 : Poll connection type

2 : Bit-Strobe connection type

3 : COS connection type

4 : Cyclic connection type

InputDataLenList: [output] The input data length of all the slave devices in the ScanList.

OutputDataLenList: [output] The output data length of all the slave devices in the ScanList.

EPR\_List: [output] The EPR value of all the slave devices in the ScanList.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

## 2.5.27 I7565DNM\_ImportEEPROM

- **Description :**

This function will write all specific devices' information in the ScanList to the EEPROM.

- **Syntax :**

DWORD I7565DNM\_ImportEEPROM(BYTE DevPort, WORD ListCount, BYTE \*DesMACIDList, BYTE \*ConnectionTypeList, WORD \*InputDataLenList, WORD \*OutputDataLenList, WORD \*EPR\_List);

- **Parameter :**

DevPort : [input] The USB port number.

ListCount: [input] The data count of all the information.

DestMACIDList: [output] The MAC ID of all the slave devices in the

ScanList.

ConnectionTypeList: [input] The connection type of all slave devices.

0 : Explicit connection type

1 : Poll connection type

2 : Bit-Strobe connection type

3 : COS connection type

4 : Cyclic connection type

InputDataLenList: [input] The input data length of all slave devices.

OutputDataLenList: [input] The output data length of all slave devices.

EPR\_List: [input] The EPR value of all slave devices.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

## 2.5.28 I7565DNM\_SendExplicitMSG

- **Description :**

This function is used to send the explicit request command to retrieve or configure the attribute value of the specific device's instance. Before calling this function, you must start the device. After calling this function, you should execute the "I7565DNM\_GetExplicitMSGRespValue" to get the response message returned from remote slave device.

- **Syntax :**

```
DWORD I7565DNM_SendExplicitMSG(BYTE DevPort, BYTE  
DesMACID, BYTE ServiceID, BYTE ClassID, BYTE InstanceID, WORD  
DataLen, BYTE *DATA);
```

- **Parameter :**

DevPort : [input] The USB port number.

DestMACID: [input] The remote slave device's MAC ID (0~63)

ServiceID: [input] The remote slave device's ServiceID.

ClassID: [input] The remote slave device's ClassID.

InstanceID: [input] The remote slave device's InstanceID.

DataLen: [input] The length of the attribute value (in byte).

DATA: [input] The attribute value that the users want to send.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").



---

### 2.5.29 I7565DNM\_IsExplicitMSGRespOK

- **Description :**  
This function is used to check whether the I-7565-DNM has received the response message or not. After checking the response message, you should execute the "I7565DNM\_GetExplicitMSGRespValue" to get the response message returned from remote slave device.
- **Syntax :**  
DWORD I7565DNM\_IsExplicitMSGRespOK(BYTE DevPort, BYTE DestMACID);;
- **Parameter :**  
DevPort : [input] The USB port number.  
DestMACID: [input] The remote slave device's MAC ID (0~63)
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.5.30 I7565DNM\_GetExplicitMSGRespValue

- **Description :**  
This function is used to get the attribute value of the specific device's instance from the remote slave device. Before calling this function, the users should call I7565DNM\_SendExplicitMSG to send request command first.
- **Syntax :**  
DWORD I7565DNM\_GetExplicitMSGRespValue(BYTE DevPort, BYTE DestMACID, WORD \*DataLen, BYTE \*DATA);
- **Parameter :**  
DevPort : [input] The USB port number.  
DestMACID: [input] The remote slave device's MAC ID (0~63)  
DataLen: [output] The length of the attribute value (in byte).  
DATA: [output] The attribute value that returned from the slave device.
- **Return:**  
Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

## 2.6 I/O Configuration Functions

---

### 2.6.1 I7565DNM\_ReadInputData

- **Description :**

This function is to get the data according with the produced connection path of the specific MAC ID device via the I/O connection.

- **Syntax :**

DWORD I7565DNM\_ReadInputData(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD \*IOLen, BYTE \*IODATA);

- **Parameter :**

DevPort : [input] The USB port number.

DestMACID: [input] The remote slave device's MAC ID (0~63)

ConType: [input] The connection type of the remote slave.

0 : Explicit connection type

1 : Poll connection type

2 : Bit-Strobe connection type

3 : COS connection type

4 : Cyclic connection type

IOLen: [output] The length of the I/O data (In byte).

IODATA: [output] The remote I/O data.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 2.6.2 I7565DNM\_WriteOutputData

- **Description :**

The function will set the data according with the consumed connection path of the specific MAC ID device via the I/O connection.

- **Syntax :**

DWORD I7565DNM\_WriteOutputData(BYTE DevPort, BYTE DesMACID, BYTE ConType, WORD IOLen, BYTE \*IODATA);

- **Parameter :**

DevPort : [input] The USB port number.

DestMACID: [input] The remote slave device's MAC ID (0~63)

ConType: [input] The connection type of the remote slave.

0 : Explicit connection type

- 1 : Poll connection type
- 2 : Bit-Strobe connection type
- 3 : COS connection type
- 4 : Cyclic connection type

IOLen: [Input] The length of the I/O data (In byte).

IODATA: [Input] The remote I/O data.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

### 2.6.3 I7565DNM\_ReadbackOutputData

- **Description :**

The function will read the data according with the consumed connection path of the specific MAC ID device via the I/O connection.

- **Syntax :**

DWORD I7565DNM\_ReadbackOutputData(BYTE DevPort, BYTE DestMACID, BYTE ConType, WORD \*IOLen, BYTE \*IODATA);

- **Parameter :**

DevPort : [input] The USB port number.

DestMACID: [input] The remote slave device's MAC ID (0~63)

ConType: [input] The connection type of the remote slave.

- 0 : Explicit connection type
- 1 : Poll connection type
- 2 : Bit-Strobe connection type
- 3 : COS connection type
- 4 : Cyclic connection type

IOLen: [Output] The length of the I/O data (In byte).

IODATA: [Output] The remote I/O data.

- **Return:**

Return 0 means success, others means failure (Please refer to "Section 2.1 Error Code").

---

### 3. i-7565-DNM Demo Programs For Linux

Users can develop their DeviceNet application by applying the API functions. We provide the demos for users to show how to implement the system. Users can refer to the table 3.1 to understand the software package's structure.

Table 3.1

Directory Path	File Name	Description
Include	i7565DNM.h	The header of i-7565-DNM library.
lib	libI7565DNM.a	The i-7565-DNM library for x86 Linux PC.
doc	i-7565-DNM-Linux-Manual.pdf	The Linux manual for i-7565-DNM.
examples	I7565DNM.c	The source code of i-7565-DNM's demo.

---

#### 3.1 Demo code “i7565DNM”

The quick start demo is a good reference for users to get into the DeviceNet application by I-7565-DNM. Please follow the process step by step.

1. User must prepare for the equipments below (PC , I-7565-DNM, CAN-8124, I-8077). Plug I-7565-DNM into the PC's USB port and plug I-8077 into CAN-8124's slot. I-7565-DNM 's CAN\_H to CAN-8124's CAN\_H and I-7565-DNM 's CAN\_L to CAN-8124's CAN\_L as the following figure 3-1 and 3-2.



Figure 3-1

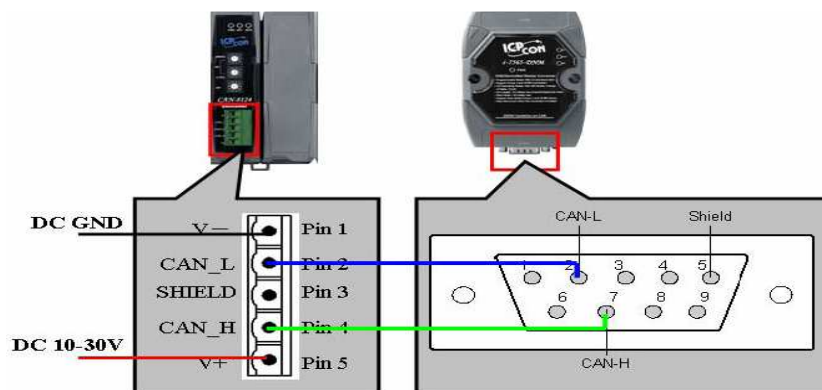


Figure 3-2

2. User must set the baud rate and MAC ID of CAN-8124. In this example, please refer to the figure 3-3 to set MSD = 1, LSD = 2, DR = 0 as follows. (MAC ID = 12, Rate = 125) ,

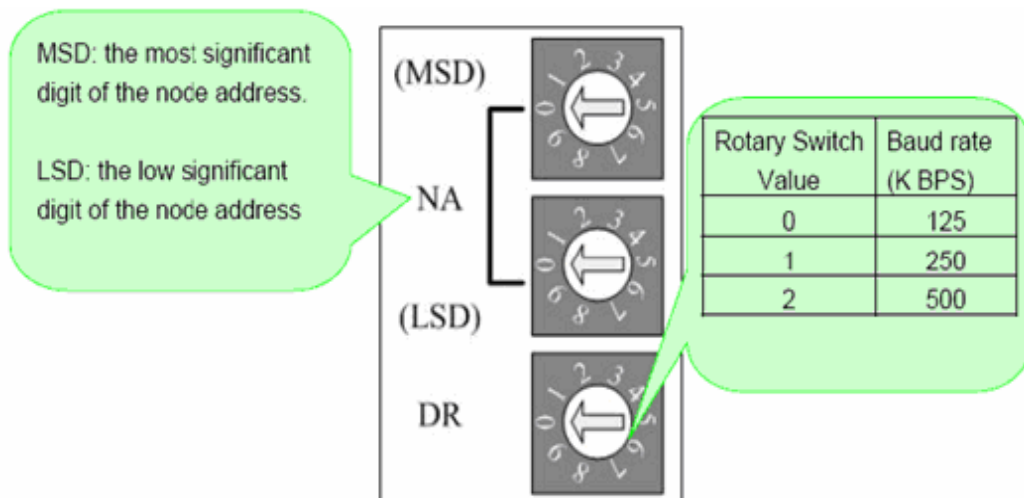


Figure 3-3

3. User can extract i-7565-DNM Linux software package into current directory and use command “**make**” to build quick start demo. Please refer to figure 3-4.

```
[root@localhost I7565DNM]# tar xzf I7565DNM_Linux_Library.tar.gz
[root@localhost I7565DNM]# cd I7565DNM_Linux_Library
[root@localhost I7565DNM_Linux_Library]# make
cd src && make
make[1]: Entering directory `/root/goldentask/I7565DNM/I7565DNM_Linux_Library/src'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/root/goldentask/I7565DNM/I7565DNM_Linux_Library/src'
```

Figure 3-4

4. User can execute the quick start demo “i7565DNM”, then choosing the i-7565-DNM device name “**ttyUSB\***” in user’s Linux OS. Please refer to figure 3-5.

```
[root@localhost examples]# ls
i7565DNM i7565DNM.c Makefile
[root@localhost examples]# ./i7565DNM

1 : ttyUSB0
2 : ttyUSB1
3 : ttyUSB2
4 : ttyUSB3
USB-Serial Device Name(1~4) :1
a. Get I-7565DNM Firmware Version:
b. Reset I-7565DNM Firmware:
c. Get I-7565DNM CAN Bus Baud Rate:
d. Set I-7565DNM CAN Bus Baud Rate:
e. Get I-7565DNM MAC ID:
f. Set I-7565DNM MAC ID:
g. Clear EEPROM Configuration:
h. Add Slave(MACID 12) Device:
i. Remove Slave(MACID 12) Device:
j. Add Slave(MACID 12) Poll I/O Connection:
k. Remove Slave(MACID 12) PollI/O Connection:
l. Start Slave(MACID 12) Device:
m. Stop Slave(MACID 12) Device:
n. Read Slave(MACID 12) Input Data:
o. Write Slave(MACID 12) Output Data:
y. Show All I-7565DNM's Demo Options:
z. Shutdown and exit:
```

Figure 3-5

5. In the step , user could follow the figure 3-6 to control the i-7565-DNM.

```
g. Clear EEPROM Configuration:
h. Add Slave(MACID 12) Device:
i. Remove Slave(MACID 12) Device:
j. Add Slave(MACID 12) Poll I/O Connection:
k. Remove Slave(MACID 12) Poll I/O Connection:
l. Start Slave(MACID 12) Device:
m. Stop Slave(MACID 12) Device:
n. Read Slave(MACID 12) Input Data:
o. Write Slave(MACID 12) Output Data:
y. Show All I-7565DNM's Demo Options:
z. Shutdown and exit:
g Using option "g" to clean all configuration
Clear I7565DNM EEPROM ... OK
h Using option "h" to add device(MAC ID 12)
Add Slave(MACID 12) Device... OK
j Using option "j" to add Poll connection
Add Slave(MACID 12) Device Poll I/O Connection Type(1) ... OK
l Using option "l" to start device
Start Slave(MACID 12) Device... OK
n Using option "n" to read i-8077 DI data
Read Slave(MACID 12) Device Input Data : 0x3
o Using option "o" to enable i-8077 DO data
Write Output Data(0x00 ~ 0xff)
0xff Type "0xff" to enable i-8077 DO
Write Slave(MACID 12) Device Output Data : 0xff
```

Figure 3-6