

MS-9A79
GPIO API
For Windows 10
64bit

Programming Guide
Ver. 0.1

10/17/2019
Prepared by MSI IPC

Document Revision History

Date	Version	Modifier	Remark
2019/10/17	0.1	Josh Wu	Initial version

Content

1.	Specification	1
2.	File List	1
3.	Function List	2
4.	GPIO API Control Example.....	3
4.1	<i>Sample Diagram of GPO application</i>	3
4.2	Pseudo Code in C Format of GPO application.....	3
4.3	Sample Diagram of GPI application	4
4.4	Pseudo Code in C Format of GPI application	4
5.	Function Details	5
5.1	IsMsiDriverOpen.....	5
5.2	GPIO_SetGPOValue	5
5.3	GPIO_GetGPOValue.....	5
5.4	GPIO_GetGPIValue	6

1. Specification

The GPIO API shall be implemented in the following specifications under Microsoft Windows operating system.

- (a). The GPIO API has no default value.
- (b). The “set output to High or Low” of GPIO outputs function supported
- (c). The “read value” of GPIO inputs function supported

Supported Operating System:

Microsoft Windows 10 64bit

2. File List

To use the GPIO API, The GPIO application shall include:

- (a). A header file (.h): **msiDrv.h** for msiBoardDriver.dll
- (b). A static library (.lib): **msiBoardDriver.lib**
- (c). A dynamic link library (.dll): **msiBoardDriver.dll**
- (d). A driver installation program: **InstallDriver.exe**

Note:

1. Please have administrator privileges on the first call DLL to automatically install the driver. Otherwise, use “InstallDriver.exe” to manually install the driver.

3. Function List

There are 4 functions in the GPIO API, listed as following. Please refer to chapter 5 for function details.

No.	Function Name	Return Data Type	Description
1	<i>IsMsiDriverOpen</i>	bool	Check driver availability
2	<i>GPIO_SetGPOValue</i>	bool	Set GPO output value of given port
3	<i>GPIO_GetGPOValue</i>	short	Get GPO output value of given port
4	<i>GPIO_GetGPISValue</i>	short	Get GPI input value of given port

3.1 Content of API Header file “msiDrv.h”

```
#pragma once

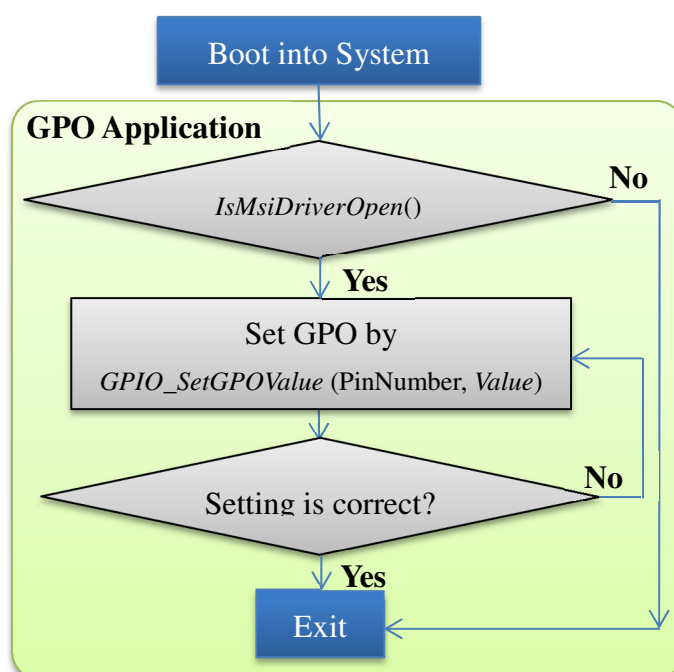
// Functions exported from DLL.
// Check driver availability, Returns true if the driver was opened successfully
bool _stdcall IsMsiDriverOpen();

// GPIO related function
// Set the value (HIGH (true)/LOW (false)) of GPO PinNumber. Returns false for invalid PinNumber.
bool _stdcall GPIO_SetGPOValue(short PinNumber, bool value);
// Get the value of GPI PinNumber. Return 1 for HIGH / 0 for LOW / -1 for invalid PinNumber
short _stdcall GPIO_GetGPISValue(short PinNumber);
// Get the value of GPO PinNumber. Return 1 for HIGH / 0 for LOW / -1 for invalid PinNumber
short _stdcall GPIO_GetGPOValue(short PinNumber);
```

4. GPIO API Control Example

Here is an example shows how to use GPIO API in the GPIO application. First, check the driver availability by function “*IsMsiDriverOpen*”. Then set the output value of GPO by function “*GPIO_SetGPOValue*”, you can use the return value to check if the setting have problem. To get the input value of GPI, please use the function “*GPIO_GetGPIValue*”.

4.1 Sample Diagram of GPO application



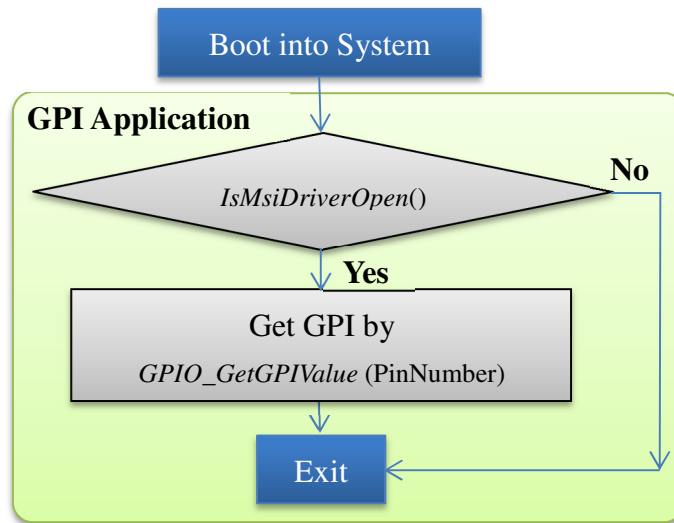
4.2 Pseudo Code in C Format of GPO application

```

#include "msiDrv.h" // include the API header file

bool set_GPO(short PinNumber, bool PinValue)
{
    if (IsMsiDriverOpen() == false) // Check the driver availability
        Return 0; // If driver not loaded, exit program
    if (GPIO_SetGPOValue (PinNumber, PinValue)== false )
        return false; // Setting is wrong, exit program
    return true; // Exit program.
}
  
```

4.3 Sample Diagram of GPI application



4.4 Pseudo Code in C Format of GPI application

```

#include "msiDrv.h" // include the API header file

void read_GPI ( short PinNumber )
{
    if (IsMsiDriverOpen () == false) // Check the driver availability
        Return 0; // If driver not loaded, exit program
    Switch (GPIO_GetGPIValue (PinNumber) )
    {
        Case -1: // PinNumber is wrong.
            printf("Wrong Ping Number!")
            break;
        Case 0: // Input value of PinNumber is low.
            printf("Input value of Pin GPI %d is Low", PinNumber);
            break;
        Case 1: // Input value of PinNumber is high.
            printf("Input value of Pin GPI%d is High", PinNumber);
            break;
    }
}
  
```

5. Function Details

Functions in GPIO API are described as follows.

5.1 IsMsiDriverOpen

Input: N/A

Return Data Type: bool

Return Value:

TRUE: Driver is successfully opened.

FALSE: Can't open driver.

Description:

Use this function to check the driver availability when GPIO application starts. If returns fail, please install the driver manually by running **InstallDriver.exe** with administrator privilege.

5.2 GPIO_SetGPOValue

Input Data Type:

Pin Number: short

Value: bool

Input Data Value and Range:

Pin Number: 0 ~ n (GPO0~ GPO n), n depends on hardware design.

Value: FALSE for output low, TRUE for output high.

Return Data Type: bool

Return Value:

TRUE: GPIO set successfully.

FALSE: GPIO can't set, Pin Number is not correct.

Description:

Set GPO output value of given port number.

5.3 GPIO_GetGPOValue

Input Data Type:

Pin Number: short

Input Data Value and Range:

Pin Number: 0 ~ n (GPO0~ GPO n), n depends on hardware design.

Return Data Type: short

Return Value:

- 1 : The value of GPO is high.
- 0 : The value of GPO is low.
- 1 : Pin Number of GPO is not correct.

Description:

Get GPO output value of given port number.

5.4 GPIO_GetGPIValue

Input Data Type:

Pin Number: short

Input Data Value and Range:

Pin Number: 0 ~ n (GPIO~ GPI n), n depends on hardware design.

Return Data Type: short

Return Value:

- 1 : The value of GPI is high.
- 0 : The value of GPI is low.
- 1 : Pin Number of GPI is not correct.

Description:

Get GPI input value of given port number.