

EMA7308/ EMA7308A
EMA7308D/ EMA7308DA
Ethernet Analog I/O module

Software Manual (V1.1)

健昇科技股份有限公司

JS AUTOMATION CORP.

台北縣汐止市中興路 100 號 6 樓

6F, No.100, Chungshin Rd.

Shitsu, Taipei, Taiwan, R.O.C.

TEL : 886-2-2647-6936

FAX : 886-2-2647-6940

<http://www.automation.com.tw>

E-mail : control.cards@automation.com.tw

Correction record

Version	Record
1.0	EMA7308.dll v1.0
1.1	delete the unused parameter “channel” of EMA7308_AD_read
	modify EMA7308_DA_set parameter description

Contents

1.	How to install the software of EMA7308.....	3
1.1	Install the EMA driver	3
2.	Where to find the file you need	4
3.	About the EMA7308 software.....	5
3.1	What you need to get started.....	5
3.2	Software programming choices	5
4.	EMA7308 Language support.....	6
4.1	Building applications with the EMA7308 software library.....	6
5.	Software overview	7
5.1	Initialization and close.....	7
5.2	Analog Input/Output function.....	7
5.3	Configuration function.....	7
5.4	Software key function.....	7
5.5	Error codes and address	8
5.6	Variable data types	9
5.7	Programming language considerations	10
5.8	EMA7308 Functions.....	12
	Initialization and close.....	12
	EMA7308_Initial.....	12
	EMA7308_close	13
	Analog input/output function.....	14
	EMA7308_AD_read.....	14
	EMA7308_parameters_set.....	15
	EMA7308_parameters_read	16
	EMA7308_DA_set	17
	EMA7308_DA_read.....	18
	Miscellaneous function.....	19
	EMA7308_socket_port_change.....	19
	EMA7308_IP_change.....	19
	EMA7308_reboot	20
	Software key function.....	21
	EMA7308_security_unlock.....	21
	EMA7308_security_status_read.....	21
	EMA7308_password_change	22
	EMA7308_password_set_default.....	22
5.9	DLL list.....	23
6.	EMA7308 Error codes summary	24
6.1	EMA7308 Error codes table	24

1. How to install the software of EMA7308

Please register as user's club member to download the "Step by step installation of EMA7308" document from <http://automation.com.tw>

1.1 Install the EMA driver

The ether net module can not found by OS as PCI cards. You can just install the driver without the module installed. Execute the file ..\install\EMA7308_Install.exe to install the driver, Api and demo program automatically.

For a more detail descriptions, please refer "Step by step installation of EMA7308".

2. **Where to find the file you need**

Windows2000, XP and up

In Windows 2000,XP,Vista system, the demo program can be setup by

If you use the default setting, a new directory ..\JS Automation\EMA7308 will generate to put the associate files.

../ JS Automation /EMA7308/API (header files and VB,VC lib files)

../ JS Automation /EMA7308/Driver (copy of driver code)

../ JS Automation /EMA7308/exe (demo program and source code)

The dll is located at ..\system.

3. About the EMA7308 software

EMA7308 software includes a set of dynamic link library (DLL) based on socket that you can utilize to control the interface functions.

Your EMA7308 software package includes setup driver, test program that help you how to setup and run appropriately, as well as an executable file which you can use to test each of the EMA7308 functions within Windows' operation system environment.

3.1 What you need to get started

To set up and use your EMA7308 software, you need the following:

- EMA7308 software
- EMA7308 hardware

3.2 Software programming choices

You have several options to choose from when you are programming EMA7308 software. You can use Borland C/C++, Microsoft Visual C/C++, Microsoft Visual Basic, or any other Windows-based compiler that can call into Windows dynamic link libraries (DLLs) for use with the EMA7308 software.

4. **EMA7308 Language support**

The EMA7308 software library is a DLL used with Windows 2000/XP/Vista. You can use these DLL with any Windows integrating development environment that can call Windows DLLs.

4.1 Building applications with the EMA7308 software library

The EMA7308 function reference section contains general information about building EMA7308 applications, describes the nature of the EMA7308 functions used in building EMA7308 applications, and explains the basics of making applications using the following tools:

Applications tools

- Borland C/C++
- Microsoft Visual C/C++
- Microsoft Visual Basic

If you are not using one of the tools listed, consult your development tool reference manual for details on creating applications that call DLLs.

EMA7308 Windows Libraries

The EMA7308 for Windows function library is a DLL called **EMA7308.dll**. Since a DLL is used, EMA7308 functions are not linked into the executable files of applications. Only the information about the EMA7308 functions in the EMA7308 import libraries is stored in the executable files.

Import libraries contain information about their DLL-exported functions. They indicate the presence and location of the DLL routines. Depending on the development tools you are using, you can make your compiler and linker aware of the DLL functions through import libraries or through function declarations.

Refer to **Table 1** to determine to which files you need to link and which to include in your development to use the EMA7308 functions in EMA7308 .dll.

Header Files and Import Libraries for Different Development Environments		
Development Environment	Header File	Import Library
Microsoft C/C++	EMA7308.h	EMA7308VC.lib
Borland C/C++	EMA7308.h	EMA7308BC.lib
Microsoft Visual Basic	EMA7308.bas	

Table 1

5. Software overview

5.1 Initialization and close

You need to initialize system resource and port and IP each time you run your application,

[EMA7308 Initial\(\)](#) will do.

Once you want to close your application, call

[EMA7308 close\(\)](#) to release all the resource.

5.2 Analog Input/Output function

To read analog input use

[EMA7308 AD read\(\)](#)

To setup the analog input parameters

[EMA7308 parameters set\(\)](#) and read back status by

[EMA7308 parameters read\(\)](#)

To output analog voltage, use

[EMA7308 DA set\(\)](#) and read back the output register by

[EMA7308 DA read\(\)](#)

5.3 Configuration function

To change the socket port by

[EMA7308 socket port change\(\)](#) and change IP by

[EMA7308 IP change\(\)](#)

Sometimes you need to reset the system (hot reset), you can commend by

[EMA7308 reboot\(\)](#)

5.4 Software key function

To prevent un-authorized person to change the settings and outputs, software key is an essential protection. If you want to commend to change settings or output, you must unlock first by

[EMA7308 security unlock\(\)](#) and read back the status of security by

[EMA7308 security status read\(\)](#)

If you want to change password, use

[EMA7308 password change\(\)](#) will do.

If you forget the password and you want to reset password to factory default value remotely,

[EMA7308 password set default\(\)](#)^{*1} will do.

**1 Command concerning the system rebooting, please wait for about 10s to proceed the next communication.*

5.5 Error codes and address

Every EMA7308 function is consist of the following format:

Status = function_name (parameter 1, parameter 2, ... parameter n)

Each function returns a value in the **Status** global variable that indicates the success or failure of the function. A returned **Status** equal to zero that indicates the function executed successfully. A non-zero status indicates failure that the function did not execute successfully because of an error, or executed with an error.

Note : **Status** is a 32-bit unsigned integer.

The first parameter to almost every EMA7308 function is the parameter **CardID** which is set by *EMA7308_IP_mapping* . You can utilize multiple devices with different card ID within one application; to do so, simply pass the appropriate **CardID** to each function.

5.6 Variable data types

Every function description has a parameter table that lists the data types for each parameter. The following sections describe the notation used in those parameter tables and throughout the manual for variable data types.

Primary Type Names					
Name	Description	Range	C/C++	Visual BASIC	Pascal (Borland Delphi)
u8	8-bit ASCII character	0 to 255	char	Not supported by BASIC. For functions that require character arrays, use string types instead.	Byte
i16	16-bit signed integer	-32,768 to 32,767	short	Integer (for example: deviceNum%)	SmallInt
u16	16-bit unsigned integer	0 to 65,535	unsigned short for 32-bit compilers	Not supported by BASIC. For functions that require unsigned integers, use the signed integer type instead. See the i16 description.	Word
i32	32-bit signed integer	-2,147,483,648 to 2,147,483,647	long	Long (for example: count&)	LongInt
u32	32-bit unsigned integer	0 to 4,294,967,295	unsigned long	Not supported by BASIC. For functions that require unsigned long integers, use the signed long integer type instead. See the i32 description.	Cardinal (in 32-bit operating systems). Refer to the i32 description.
f32	32-bit single-precision floating-point value	-3.402823E+38 to 3.402823E+38	float	Single (for example: num!)	Single
f64	64-bit double-precision floating-point value	-1.797685123862315E+308 to 1.797685123862315E+308	double	Double (for example: voltage Number)	Double

Table 2

5.7 Programming language considerations

Apart from the data type differences, there are a few language-dependent considerations you need to be aware of when you use the EMA7308 API. Read the following sections that apply to your programming language.

Note: Be sure to include the declaration functions of EMA7308 prototypes by including the appropriate EMA7308 header file in your source code. Refer to Chapter 4. EMA7308 Language Support for the header file appropriate to your compiler.

5.7.1 C/C++

For C or C++ programmers, parameters listed as Input/Output parameters or Output parameters are pass-by-reference parameters, which means a pointer points to the destination variable should be passed into the function. For example, the read port function has the following format:

```
Status = EMA7308_read_port (u8 CardID, u8 port, u8 *data);
```

where **CardID** and **port** are input parameters, and **data** is an output parameter.

To use the function in C language, consider the following example:

```
u8 CardID=0, port=0 ; //assume CardID is 0 and port also 0
u8 data,
u32 Status;
Status = EMA7308_read_port ( CardID, port, &data);
```

5.7.2 Visual basic

The file EMA7308.bas contains definitions for constants required for obtaining LSI Card information and declared functions and variable as global variables. You should use these constants symbols in the EMA7308.bas, do not use the numerical values.

In Visual Basic, you can add the entire EMA7308.bas file into your project. Then you can use any of the constants defined in this file and call these constants in any module of your program. To add the EMA7308.bas file for your project in Visual Basic 4.0, go to the **File** menu and select the **Add File...** option. Select EMA7308.bas, which is browsed in the EMA7308 \ api directory. Then, select **Open** to add the file to the project.

To add the EMA7308.bas file to your project in Visual Basic 5.0 and 6.0, go to the **Project** menu and select **Add Module**. Click on the Existing tab page. **Select** EMA7308.bas, which is in the EMA7308 \api directory. Then, select **Open** to add the file to the project.

If you want to use under .NET environment, please download “

5.7.3 Borland C++ builder

To use Borland C++ builder as development tool, you should generate a .lib file from the .dll file by implib.exe.

```
implib EMA7308bc.lib EMA7308.dll
```

Then add the **EMA7308bc.lib** to your project and add

#include "EMA7308.h" to main program.

Now you may use the dll functions in your program. For example, the Read Input function has the following format:

```
Status = EMA7308_read_port ( CardID, port, &data);
```

where **CardID** and **port**, are input parameters, and **data** is an output parameter. Consider the following example:

```
u8 CardID=0, port=0 ; //assume CardID is 0 and port also 0
```

```
u8 data,
```

```
u32 Status;
```

```
Status = EMA7308_read_port ( CardID, port, &data);
```

* If you are using Delphi, please refer to <http://www.drbob42.com/headconv/index.htm> for more detail about the difference of C++ and Delphi.

5.8 EMA7308 Functions

Initialization and close

● **EMA7308 Initial**

Format : u32 status =EMA7308_Initial (u8 CardID,u8 IP_Address[4] , u16 Host_Port,u16 Remote_port,u16 TimeOut_ms, u8 *CardType)

Purpose: To map IP and PORT of an existing EMA7308 to a specified CardID number.

Parameters:

Input:

Name	Type	Description
CardID	u8	Assign CardID to the EMA7308 of a corresponding IP address.
IP_Address[4]	u8	4 words of IP address For example: if IP address is "192.168.0.100" then IP_Address[0]=192 IP_Address[1]=168 IP_Address[2]=0 IP_Address[3]=100 Default:192.168.0.100
Host_Port	u16	Assign a communicate port of host PC Default:15120
Remote_port	u16	Assign a communicate port of EMA7308 Default:6936
TimeOut	u16	Assign the max delay time of EMA7308 response message,1000~10000 ms.

Output:

Name	Type	Description
CardType	u8	Get the Card Type of EMA7308 1: EMA-7308DA 2: EMA-7308D 3: EMA-7308A 4: EMA-7308

● **EMA7308_close**

Format : u32 status =EMA7308_close (u8 CardID)

Purpose: Release the EMA7308 resource when closing the Windows applications.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function

Analog input/output function

● **EMA7308 AD read**

Format : u32 status = EMA7308_AD_read(u8 CardID, AD_Data data[16])

Purpose: Read A/D raw data and conversion result (data).

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function

Output:

Name	Type	Description
data	AD_Data	<pre>typedef struct _AD_Data{ i16 ad_data; f32 ad_value; } //ad_data the AD raw data //ad_value the AD value converted according //the ad_mode select //data[0] for AD00 //data[1] for AD01 //..... //data[7] for AD07 ////////// //data[8] for AD10 //data[9] for AD11 //..... //data[15] for AD17</pre>

● **EMA7308 parameters set**

Format : u32 status = EMA7308_parameters_set(u8 CardID , u8 filter_mode , u8 ad_range[16] , u8 ad_config)

Purpose: setup the EMA7308 AD channel parameters.

Parameters:

Input:

Name	Type	Description		
CardID	u8	CardID assigned by EMA7308_initial function		
filter_mode	u8		Conversion rate	RMS noise
		0	7.03KHZ	23uV
		1	3.52KHZ	3.6uV
		2	1.76KHZ	2.1uV
		3	897HZ	1.5uV
ad_range[16]	u8	ad_mode set:		
		EMD7308D, EMD7308DA	EMD7308, EMD7308A	
		0: -5V ~5V	0: 0V ~ 5V	
		1: -10V ~10V	1: 0V ~10V	
		2: 0 ~20mA	2: 0 ~20mA	
		3: 4 ~20mA	3: 4 ~20mA	
ad_config	u8	Parameter only valid for EMA-7308 and EMA-7308A. 0: Both AI00~AI07 and AI10~AI17 are in single end mode 1: AI00~AI07 in differential mode, AI10~AI17 in single end mode 2: AI00~AI07 in single end mode, AI10~AI17 in differentail mode 3: Both AI00~AI07 and AI10~AI17 are in differential mode		

Note: In differential mode, the AIn0 is paired with AIn1, ... AIn6 is paired with AIn7.

● **EMA7308 parameters read**

Format : u32 status = EMA7308_parameters_read(u8 CardID , u8 *filter_mode , u8 ad_range[16] , u8 *ad_config)

Purpose: read the EMA7308 parameters.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function

Output:

Name	Type	Description		
filter_mode	u8		Conversion rate	RMS noise
		0	7.03KHZ	23uV
		1	3.52KHZ	3.6uV
		2	1.76KHZ	2.1uV
		3	897HZ	1.5uV
ad_range[16]	u8	ad_mode set:		
		EMD7308D, EMD7308DA	EMD7308, EMD7308A	
		0: -5V ~5V	0: 0V ~ 5V	
		1: -10V ~10V	1: 0V ~10V	
		2: 0 ~20mA	2: 0 ~20mA	
		3: 4 ~20mA	3: 4 ~20mA	
ad_config	u8	Parameter only valid for EMA-7308 and EMA-7308A. 0: Both AI00~AI07 and AI10~AI17 are in single end mode 1: AI00~AI07 in differential mode, AI10~AI17 in single end mode 2: AI00~AI07 in single end mode, AI10~AI17 in differentail mode 3: Both AI00~AI07 and AI10~AI17 are in differential mode		

● **EMA7308 DA set**

Format : u32 status = EMA7308_DA_set(u8 CardID , u8 channel , i16 data[2])

Purpose: Set the value of the D/A data.

Parameters:

Input:

Name	Type	Description			
CardID	u8	CardID assigned by EMA7308_initial function			
channel	u8	channel number: 0: DA0 1: DA1 2: DA0 + DA1			
data[2]	i16	model	EMA7308, EMA7308D	EMA7308A, EMA7308DA	
		resolution	12bit DA	16bit DA	
		data range	-2048 ~ 2047	-32768 ~ 32767	
		output	data	data	
		0V	0	0	
		+10V	2047	32767	
		-10V	-2048	-32768	
		Note:			
		data[0] for DA0			
		data[1] for DA1			

● **EMA7308 DA read**

Format : u32 status = EMA7308_DA_read(u8 CardID , u8 channel , i16 data[2])

Purpose: Read back the setting of the D/A data.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function
channel	u8	channel number: 0: DA0 1: DA1 2: DA0 + DA1

Output:

Name	Type	Description
data[2]	i16	EMA7308 , EMA7308D 12bit DA → -2048 ~ 2047 0: 0V 2047: +10V -2048: -10V EMA7308A , EMA7308DA 16bit DA → -32768 ~ 32767; 0: 0V 32767: +10V -32768: -10V

Miscellaneous function

● EMA7308 socket port change

Format : `u32 status = EMA7308_socket_port_change (u8 CardID,u16 Remote_port);`

Purpose: To change the communicate port number of EMA7308.

After using this function, please wait for reboot(about 10s) to validate the change.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function
Remote_port	u16	The new port number to be set Default port is: 6936

● EMA7308 IP change

Format : `u32 status = EMA7308_IP_change (u8 CardID,u8 IP[4]);`

Purpose: To change the communicate IP of EMA7308.

After using this function, please wait for reboot(about 10s) to validate the change.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function
IP[4]	u8	The new IP to be set Default IP is: 192.168.0.100 IP_Address[0]=192 IP_Address[1]=168 IP_Address[2]=0 IP_Address[3]=100

● **EMA7308 reboot**

Format : `u32 status = EMA7308_reboot(u8 CardID);`

Purpose: To reboot EMA7308(about 10s).

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function

Software key function

● EMA7308 security unlock

Format : u32 status = EMA7308_security_unlock (u8 CardID,u8 password[8])

Purpose: To unlock security function and enable the further operation.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function
password[8]	u8	The password previous set Default :password[8] = {'1','2','3','4','5','6','7','8'};

● EMA7308 security status read

Format : u32 status = EMA7308_security_status_read(u8 CardID,u8 *lock_status);

Purpose: To read security status for checking if the card security function is unlocked.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function

Output:

Name	Type	Description
lock_status	u8	0: security unlocked 1: locked

● **EMA7308 password change**

Format : `u32 status = EMA7308_password_change(u8 CardID,u8 Oldpassword[8],
u8 password[8])`

Purpose: To replace old password with new password.

After using this function, please wait for reboot(about 10s) to validate the change.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function
Oldpassword [8]	u8	The previous password
password[8]	u8	The new password to be set

● **EMA7308 password set default**

Format : `u32 status = EMA7308_password_set_default (u8 CardID)`

Purpose: Set password to default.

After using this function, please wait for reboot(about 10s) to validate the change.

Parameters:

Input:

Name	Type	Description
CardID	u8	CardID assigned by EMA7308_initial function default :password[8] = {'1','2','3','4','5','6','7','8'};

5.9 DLL list

	Function Name	Description
1.	EMA7308_Initial ()	Map IP and get model parameter
2.	EMA7308_close()	EMA7308 close
3.	EMA7308_AD_read ()	Read A/D raw data and conversion result
4.	EMA7308_parameters_set ()	setup the EMA7308 AD channel parameters
5.	EMA7308_parameters_read ()	read the EMA7308 parameters
6.	EMA7308_DA_set ()	Set the value of the D/A data
7.	EMA7308_DA_read ()	Read back the setting of the D/A data
8.	EMA7308_socket_port_change ()	To change the communicate port number of EMA7308
9.	EMA7308_IP_change ()	To change the communicate IP of EMA7308
10.	EMA7308_reboot ()	To reboot EMA7308
11.	EMA7308_security_unlock ()	Unlock security
12.	EMA7308_security_status_read ()	Read lock status
13.	EMA7308_password_change ()	Change password
14.	EMA7308_password_set_default ()	Rest to factory default password

6. EMA7308 Error codes summary

6.1 EMA7308 Error codes table

Error Code	Symbolic Name	Description
0	JSDRV_NO_ERROR	No error.
1	INITIAL_SOCKET_ERROR	Sock can not initialized, maybe Ethernet hardware problem
2	IP_ADDRESS_ERROR	IP address is not acceptable
3	UNLOCK_ERROR	Unlock fail
4	LOCK_COUNTER_ERROR	Unlock error too many times
5	SET_SECURITY_ERROR	Fail to set security
100	DEVICE_RW_ERROR	Can not reach module
101	NO_CARD	Can not reach module
102	DUPLICATE_ID	CardID already used
300	ID_ERROR	CardID is not acceptable
301	PORT_ERROR	Port parameter unacceptable or unreachable
302	IN_POINT_ERROR	Input point unreachable
303	OUT_POINT_ERROR	Output point unreachable
305	PARAMETERS_ERROR	Parameter error
306	CHANGE_SOCKET_ERROR	Can not change socket
307	UNLOCK_SECURITY_ERROR	Fail to unlock security
308	PASSWORD_ERROR	Password mismatched
309	REBOOT_ERROR	Can not reboot
310	TIME_OUT_ERROR	Too long to response
311	CREAT_SOCKET_ERROR	Socket can not create