



Wisco DL2200 Protocol

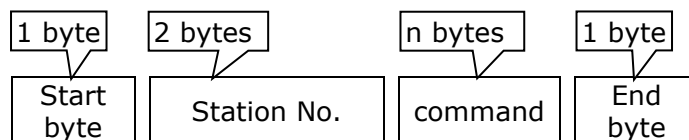


Data Logger DL2200 Protocol

There are 2 standards for connecting DL2200. First, RS-232 is peer to peer connection with PC. RS 485 is "one to many" connection, maximum connection is 32 nodes. However both of them are using the same standard protocol.

Wisco ASCII Protocol

Detail of one command as follow:



Start byte first byte, to tell module that this is start point of command set. Using symbol '#' at the beginning of command

Station No. machine number or station number in case there are more than 1 station. It can be set by using Wisco utility or device panel. Number start from 1 to 254 and station number cannot be repeated.

Command DL2200 Module command, there are 5 commands

End byte last byte of command set, using [CR] (Carriage Return) which is thirteenth alphabet of ASCII table

Character	#	0	0	R	A	I	CR
ASCII Code	23H	30H	30H	52H	41H	2AH	0DH

Example of command Wisco ASCII Protocol

Detail and Example Wisco ASCII Protocol

(= 1 byte, ... = n bytes, CR = Carriage Return)

1. **Command for read Analog Input (Floating Point)**

Command 'RAI' and end with '[CR]'. For example, read analog input from station '00'. Command set should look like '#00RAI [CR]'

#	0	0	R	A	I	CR
---	---	---	---	---	---	----

Module will respond with 'AI>' follow by readed value which is Floating Point. Each box separate by ',' and end with '[CR]' for example 'AI>50.58, 1.8, ..., 11.8 [CR]'

A	I	>	5	0	.	5	8	,	1	.	8	,	...	,	1	1	.	8	CR
---	---	---	---	---	---	---	---	---	---	---	---	---	-----	---	---	---	---	---	----

Value has to translate before user can get the correct value. For more detail of translate show at last pages.

2. **command for reading Digital Input**

Begin with 'RDI' and end with '[CR]'. command to read all Digital input:

#	0	0	R	D	I	CR
---	---	---	---	---	---	----

Module will respond as 'DI>' follow by reading digital. 1 byte for each box ('0' = OFF, '1' = ON) and end with '[CR]'. For example 'DI>0010[CR]'

D	I	>	1	0	1	0	CR
---	---	---	---	---	---	---	----

3. **command for reading Digital Output**

Begin with 'end with '[CR]' is command to read all digital output:

#	0	0	R	D	O	CR
---	---	---	---	---	---	----

Module will respond as 'DO>' follow by reading digital ('0' = OFF, '1' = ON) 1 byte for each box and end with '[CR]'. For example 'DO>0101[CR]'

D	O	>	0	1	0	1	CR
---	---	---	---	---	---	---	----

4. **command for reading Counter (DI 4)**

Begin with 'RCT' end with '[CR]' as follow:

#	0	0	R	C	T	CR
---	---	---	---	---	---	----

Module will respond as 'CT>' follow by reading value and end with '[CR]'. For example 'CT>15.8 [CR]'

C	T	>	1	5	.	8	CR
---	---	---	---	---	---	---	----

5. **Command for reading all Input/Output**

For read all AI, DI, DO, and CT starts with 'RAL' and end with '[CR]'. The example is '#00RAL [CR]'.

#	0	0	R	A	L	CR
---	---	---	---	---	---	----

Module will respond as 'AI>' follow by AI (Floating Point) for all 24 channel. Each channel will separate by ',' and end with ';' follow by 4 DI confine by ';' follow by 4 DO then separate by ',' and end with ';' then follow by CT value then ',' and end with ';' and end '[CR]'. For example 'ALL> AI, 21.57, ..., 8.21; DI, 1, 0, 0, 1; DO, 0, 1, 1, 1; CT, 15.57; [CR]'

A	L	L	>	A	I	,	2	1	.	5	7	,	...	,	8	.	2	1	;
D	I	,	1	,	0	,	0	,	1	;									
D	O	,	0	,	1	,	1	,	1	;									
C	T	,	1	5	.	5	7	;											

6. **write Digital Output**

Begin with 'WDO=' follow by value that wanted to write ('0' = OFF, '1' = ON) and each box separate by ',' then end '[CR]' for example write DO to station No. 11 channel 1=OFF, 2=ON, 3=ON, 4=OFF therefore command will be '#11WDO= 0,1,1,0 [CR]'

#	1	1	W	D	O	0	,	1	,	1	,	0	[CR]
---	---	---	---	---	---	---	---	---	---	---	---	---	------

Respond from module will be 'DO>OK' and end with '[CR]' as:

D	O	>	O	K	[CR]
---	---	---	---	---	------

Code which will respond when sending command to DL2200 is error

When sending command to DL2200 is error, module will not respond to the command and report an error code which is begin with 'ERR=' and follow by number 1-6 as detail below:

- | | |
|----------------------------|---|
| 1 (illegal function) | command is wrong or command is unknow |
| 2 (illegal data address) | value of the beginning position is out of range |
| 3 (illegal data value) | data of command is error such as value from DO is incorrect |
| 4 (invalid data frame) | command error such as missing ',' in DO command |
| 5 (check sum error) | check sum error (transfer data error) |
| 6 (invalid number of byte) | there is some data missing |

Connecting Module with MODBUS (ASCII) Protocol

DL2200 can run with Protocol Modbus which has command as: (CHAR = Character; 1 CHAR = 8 Data Bits, 1 Start Bit, and 1 Stop Bit)

ADDR	FUNCTION	DATA	ERROR CHECK	EOF	READY TO REC RESP
2-CHAR 16-BITS	2-CHAR 16-BITS	N x 4-CHAR N x 16-BITS	2-CHAR 16-BITS	CR	LF

DL2200 is supported 8 basic function of Modbus Protocol:

MODBUS ASCII

READ OUTPUT STATUS (CODE 01)
 READ INPUT STATUS (CODE 02)
 READ INPUT REGISTERS (CODE 04)
 FORCE SINGLE COIL (CODE 05)
 FORCE MULTIPLE COILS (CODE 15)

Wisco

= Read Digital Output
 = Read Digital Input
 = Read Analog Input
 = Write Digital Output
 = Write Digital Output

Address reference of module

Function Code	Reference	Address
01, 05, 15	Digital Output	0xxxx
02	Digital Input	1xxxx
04	Analog Input	3xxxx

Digital Output Table

Name	Address
Digital Output Channel 1	00001
Digital Output Channel 2	00002
Digital Output Channel 3	00003
Digital Output Channel 4	00004

Digital Input Table

Name	Address
Digital Input Channel 1	10001
Digital Input Channel 2	10002
Digital Input Channel 3	10003
Digital Input Channel 4	10004

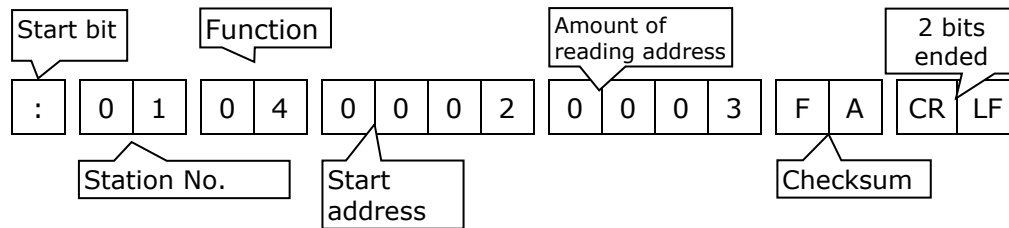
Analog Input Table (Floating Point)

Name	Address
Analog Input Channel 1	30001-30002
Analog Input Channel 2	30003-30004
Analog Input Channel 3	30005-30006
Analog Input Channel 4	30007-30008
Analog Input Channel 5	30009-30010
Analog Input Channel 6	30011-30012
Analog Input Channel 7	30013-30014
Analog Input Channel 8	30015-30016
.	.
.	.
.	.
Analog Input Channel 24	30047-30048

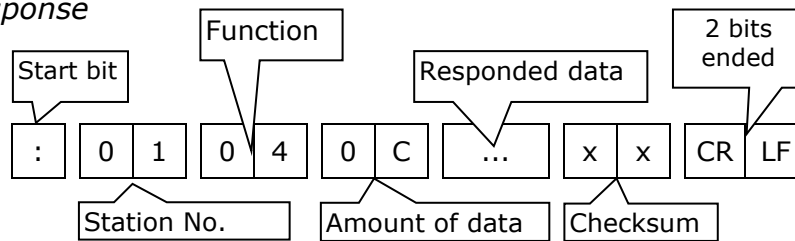
*** For more information of Modbus protocol please go to 'Modbus Reference Guide' or <http://www.modbus.org/specs.php>

Example of MODBUS (ASCII) PROTOCOL

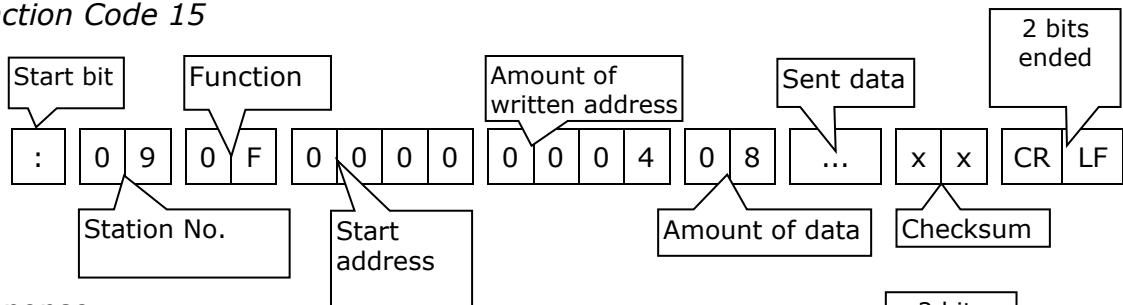
Function Code 04



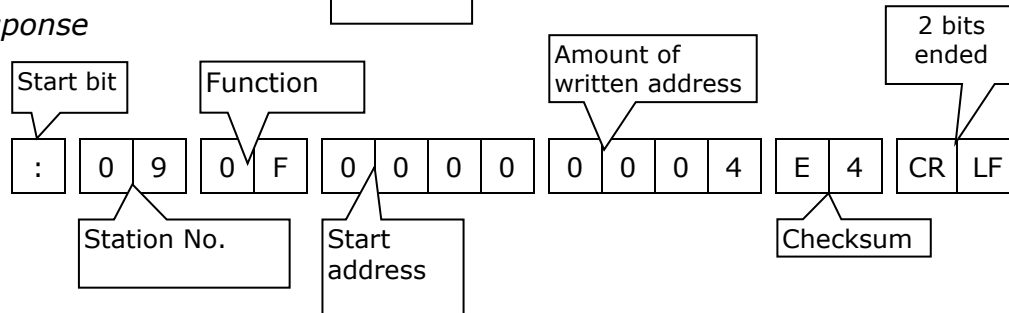
Response



Function Code 15



Response



CHECK SUM for MODBUS (ASCII) Protocol

CHECK SUM of Modbus is function checking correction of every sent data or command. CHECK SUM will sum all data (number only). each plus is equal to 1 byte, whatever is over than 1 byte will be cut and then complete 1's complement and 2's complement

Example: ': 0F 04 0001 0023 [CR] [LF]'

	HEXADECIMAL	BINARY
Start byte	0FH	0000 1111
	04H	0000 0100
	00H	0000 0000
	01H	0000 0001
	00H	0000 0000
	+ }	
Last byte	23H	0010 0011
result	37H	0011 0111
1 byte only (8 bit)	37H	0011 0111
1's complement (invert)	C8H	1100 1000
2' complement	C8H + 1	1100 1000 + 1
Check sum =	C9H	1100 1001

Sending data is ': 0F 04 0001 0023 C9 [CR] [LF]'

Edit: 26/06/2023