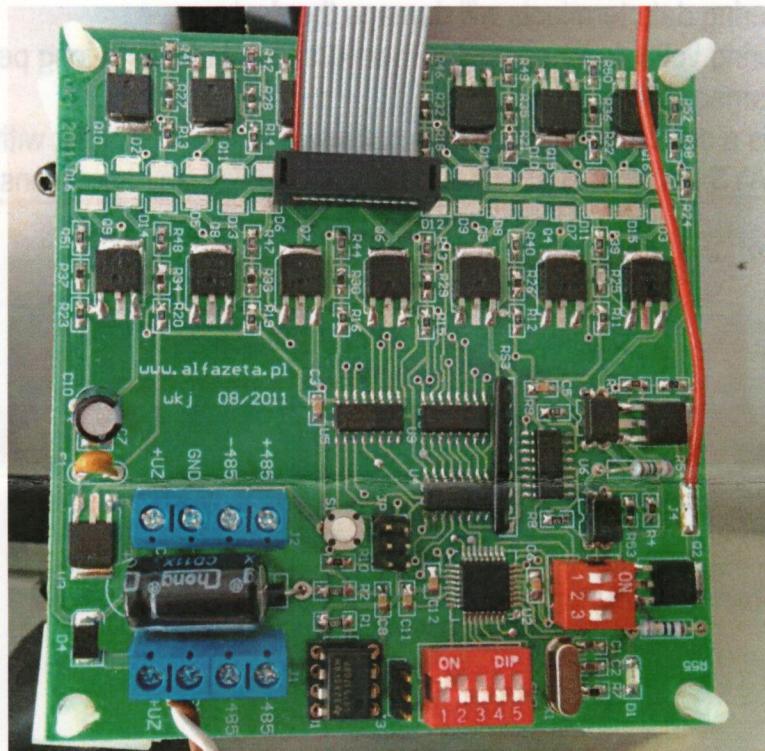


Large Seven Segment Bar (LSB) Controller



MANUAL

Attention: Please read the manual carefully.

In case of any doubts or questions pls contact

Marcin Krysinski at info@alfazeta.pl or +48 42 6891200,

Please note that Poland is GMT+1

Rev. 2 / 03 JUNE 2015

Alfa-Zeta Co. Ltd.
ul. Starorudzka 6a
93-418 Łódź
POLAND

tel. +48 42 689 12 00
tel. +48 42 689 12 01
tel. +48 42 689 12 02
fax +48 42 689 12 03

 <http://www.alfazeta.pl>
 <http://www.flipdots.com>
 <http://www.swiatlowody.com>
 <http://led.alfazeta.pl>

info@alfazeta.pl
info@flipdots.com
info@swiatlowody.com
info@alfazeta.pl

corporate website
visual information displays
fiber optic lighting
LED lighting



Handling precautions

- this is a delicate electromechanical device. Please always handle boards with care.
- This is an electrostatic sensitive device. If you do not have ESD equipment, please make sure at least that you discharge all static electricity by touching metal installation (like home radiator) before handling them.
- While connecting voltage and data cable always make sure that you are observing polarity of power supply and that you connect power supply to power supply connectors.
- Powering data terminals will damage the device.
- Powering voltage connectors with reversed polarity for a long period of time will damage the device.
- This is a device which should be put into a housing together with digits.
- ALWAYS disconnect power supply while making any operations.

MANUAL

azulnet: fibra óptica para el hogar y empresas

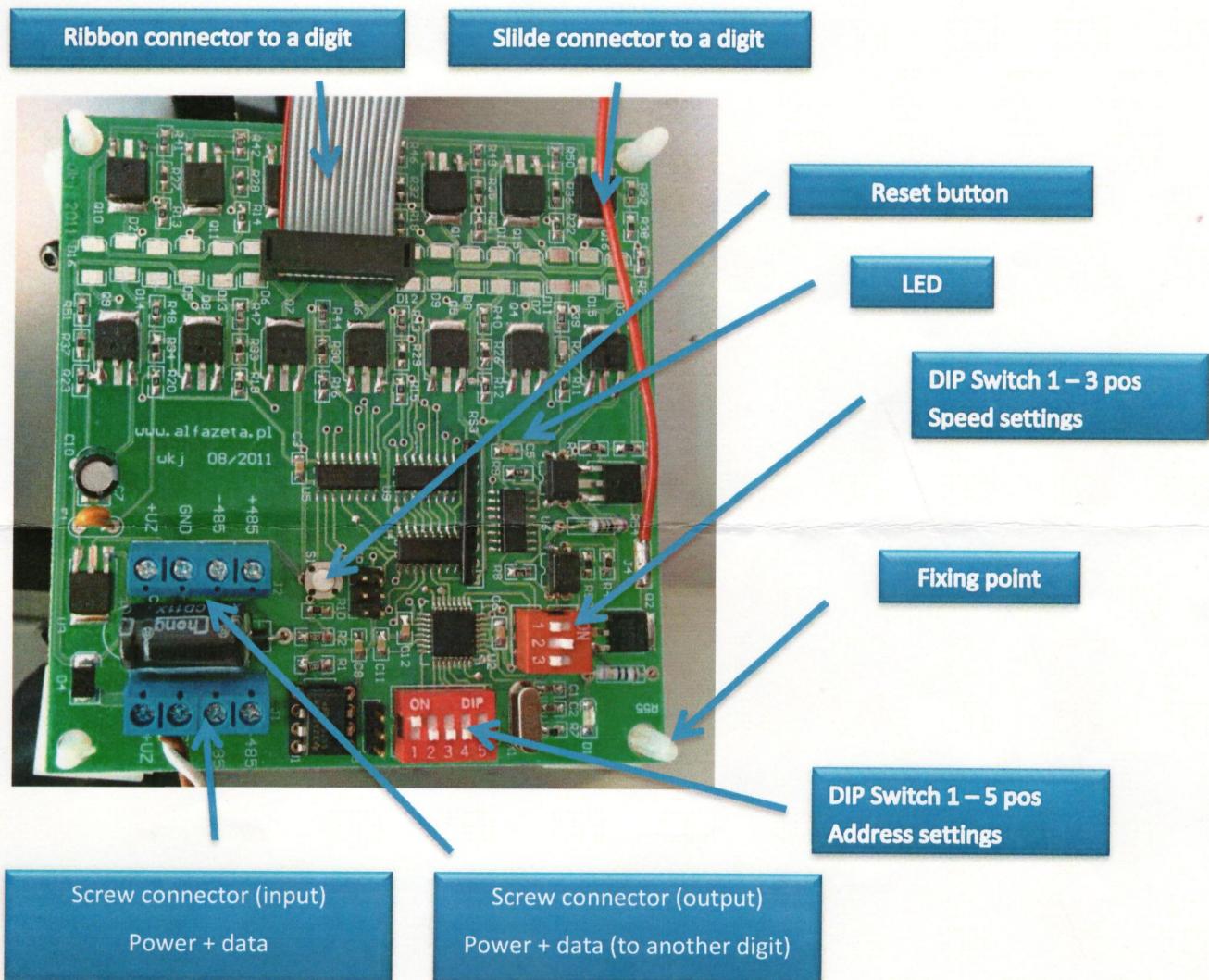


ALFAZETA

Technical data

Supply voltage	16V DC – 24V DC
Power consumption	peak 250mW
Control interface	RS485

General layout



General info

The controller is designed to control one (or two showing the same) electromagnetic large seven segment modules. It is designed to be installed at the back of a digit. It is using RS485 to communicate a master unit which can be any rs485 data sending device (a PC with RS485 interface, Arduino with rs485 extension, Raspberry Pi with rs485 extension, etc) Each controller has got its own unique address and communication speed. In order to set these parameters you need to set dip switches 1 and 2

Settings

Before using you have to set speed of communication and address.

All rs485 transmission is 8 bits of data, no parity and 1 bit of stop: 8,N,1.

Speed – Dip Switch #1 - 3 positions



(showing 4800 bit/s)

Possible communications speeds set with dip switch #1 (3 positions) in the following way:

Value	ON			Speed
	1	2	3	
0	↓	↓	↓	1200
1	↑	↓	↓	2400
2	↓	↑	↓	4800
3	↑	↑	↓	9600*
4	↓	↓	↑	19200
5	↑	↓	↑	38200
6	↓	↑	↑	9600
7	↑	↑	↑	9600
	OFF			

*) This is a suggested settings

Address – Dip Switch #2 - 5 positions



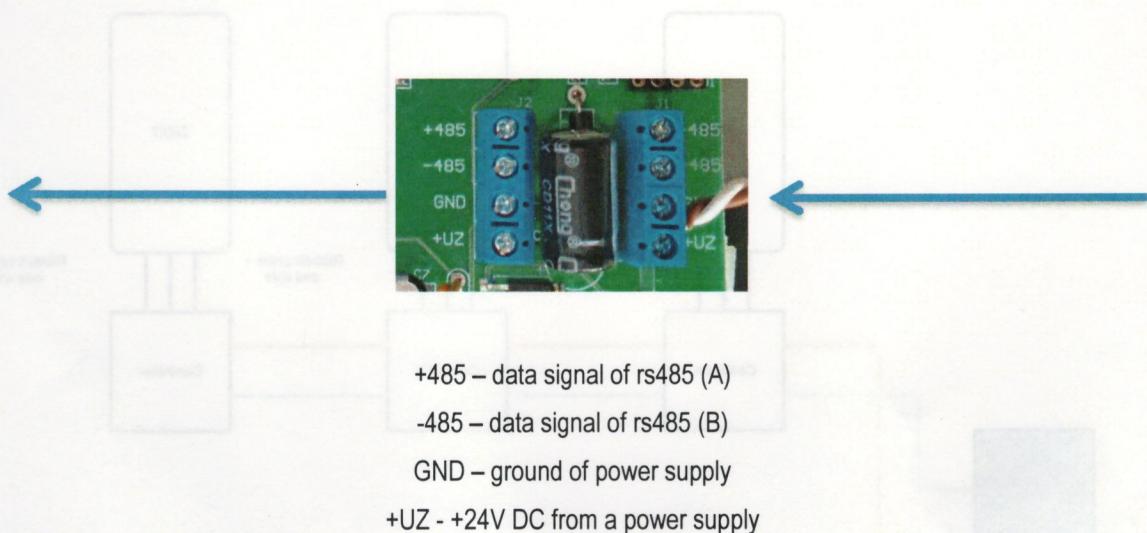
(showing address 1)

Possible addresses set with dip switch #2 (5 positions) in the following way (binary coded):

Address	ON				
	1	2	3	4	5
0	↓	↓	↓	↓	↓
1	↑	↓	↓	↓	↓
2	↓	↑	↓	↓	↓
3	↑	↑	↓	↓	↓
4	↓	↓	↑	↓	↓
5	↑	↓	↑	↓	↓
6	↓	↑	↑	↓	↓
7	↓	↑	↑	↓	↓
8	↓	↑	↑	↓	↓
...					
26	↓	↑	↓	↑	↑
27	↑	↑	↓	↑	↑
28	↓	↓	↑	↑	↑
29	↑	↓	↑	↑	↑
30	↓	↑	↑	↑	↑
31	↑	↑	↑	↑	↑
	OFF				

See details on the meaning of the address in a protocol description

Connector layout

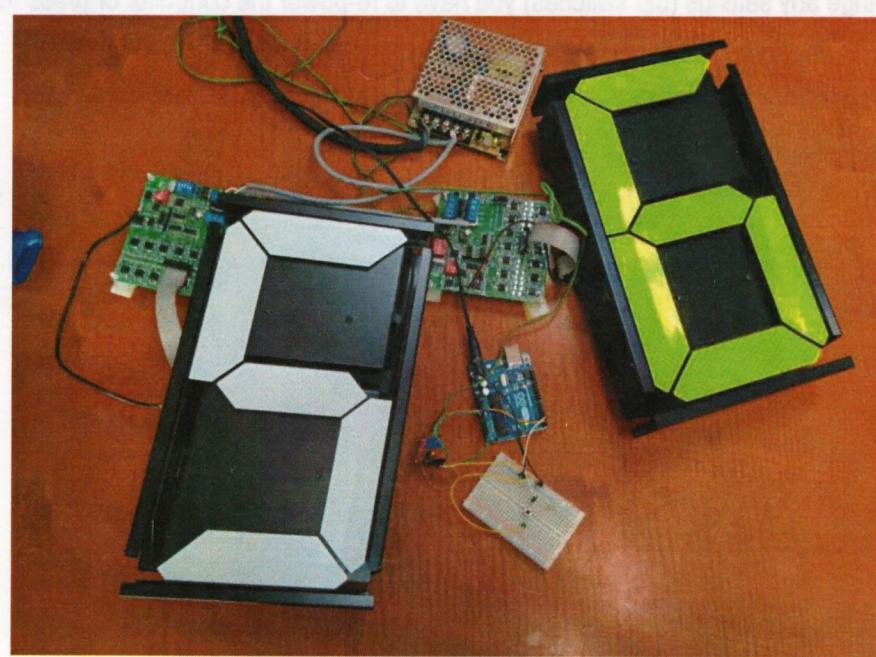
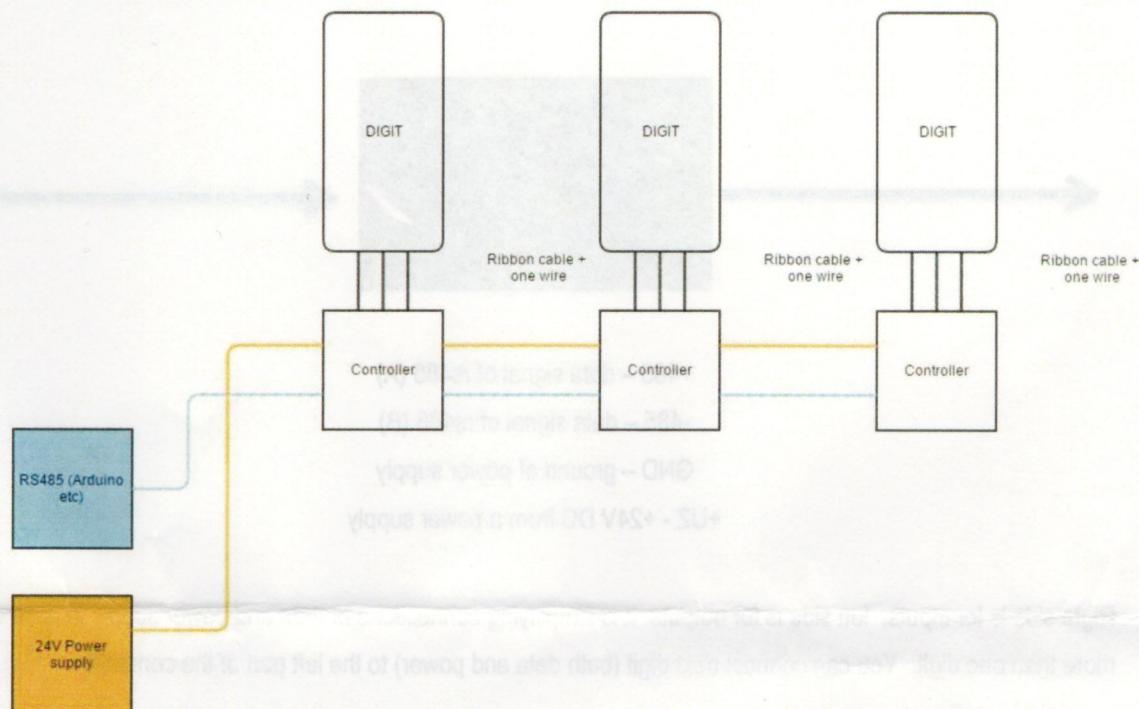


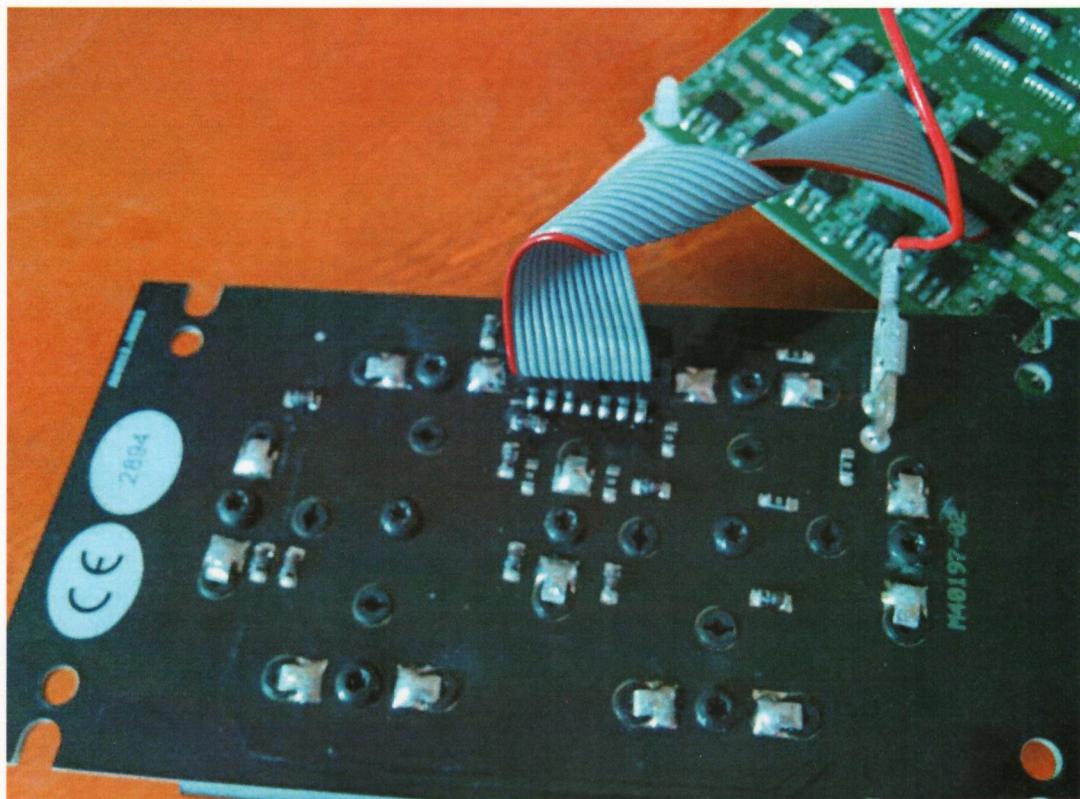
Right side is for inputs, left side is for outputs. It is simplifying connections of data and power supply in case of more than one digit. You can connect next digit (both data and power) to the left part of the connector.

After powering the display starts to operate and flips all digits to blank and then back to memorized content.

When you change any settings (dip switches) you have to re-power the controller or press reset button.

Example of connections





Control protocol for seven segment electromagnetic displays.

The DISPLAY is a set of connected MODULES. Each DISPLAY has its own unique address.

The display can be controlled in one of the following two modes:

- segment mode – control software is sending information about a state of each segment of a digit. As a result it is possible to show any combination of segments.
- ASCII mode – control software is sending ASCII characters and some of them are defined in a memory of a controller (a... f, -, 0... 9, space). If the defined character is being transmitted, it is shown on the display, if undefined character is transmitted it is shown as 'space'

Segment mode frame

Value	Name	Description
0xAA	Starting byte	
0x55	Mode indicator	0x55 for segment mode
0x00 ... number_of_connected_digits	Count	Number of transmitted characters (0x00 for 1 character, 0x01 for 2 characters, etc)
0x00 ... number_of_connected_digits	Offset	Starting position of the first character to be presented with respect to the first digit (usually 0x00)
0x00	Address of a display	0x00 for all displays
[data1]	Data bytes	
[data2]		
...		
[dataN]	N=Count+1	
[checksum]	Checksum	Sum of data[1..N] modulo 0x100

In this mode definition of digits is based on the following segment layout (0 – hidden, 1 – shown)

bit0 - segment g
 bit1 - segment f
 bit2 - segment e
 bit3 - segment d
 bit4 - segment c
 bit5 - segment b
 bit6 - segment a
 bit7 - segment DP Or N/A

resulting in the following digits definition:



Alfa-Zeta Co. Ltd.
ul. Starorudzka 6a
93-418 Łódź
POLAND



tel. +48 42 689 12 00
tel. +48 42 689 12 01
tel. +48 42 689 12 02
fax +48 42 689 12 03



<http://www.alfazeta.pl>
<http://www.flipdots.com>
<http://www.swiatlowody.com>
<http://led.alfazeta.pl>

info@alfazeta.pl
info@flipdots.com
info@swiatlowody.com
info@alfazeta.pl

corporate website
visual information displays
fiber optic lighting
LED lighting

'0' → 0x7E
'1' → 0x30
'2' → 0x6D
'3' → 0x79
'4' → 0x33
'5' → 0x5B
'6' → 0x5F
'7' → 0x70
'8' → 0x7F
'9' → 0x7B

Example:

To show on the first three modules '123' you have to send the following:

0xAA	Start character
0x55	Mode indicator
0x02	Number of characters -1
0x00	Offset of the first character
0x00	Address of the display
0x30	'1'
0x6D	'2'
0x79	'3'
0x16	checksum

Character mode frame

Value	Name	Description
0xAA	Starting byte	
0x56	Mode indicator	0x56 for character mode
0x00 ... number_of_connected_digits	Count	Number of transmitted characters (0x00 for 1 character, 0x01 for 2 characters, etc)
0x00 ... number_of_connected_digits	Offset	Starting position of the first character to be presented with respect to the first digit (usually 0x00)
	Address	0x00 for all modules
[data1]	Data bytes	
[data2]		
...		
[dataN]	N=Count+1	
[checksum]	Checksum	Sum of data[1..N] modulo 0x100

Example:

To show '123' starting from the second module you have to transmit:

0xAA	Start character
0x56	Mode indicator
0x02	Number of characters -1
0x01	Offset of the first character
0x00	Address of the display
0x31	'1'
0x32	'2'
0x33	'3'
0x96	checksum

0xAA, 0x56, 0x02, 0x00, 0x00, 0x31, 0x32, 0x33,
0x96