RUNMODE S7JOYSTICK
Version 2.x

SEND JOYSTICK DATA
TO THE PLC

Communication utility suitable for

- Siemens S7-300/-400 CPUs
- Siemens TIA 1200/1500 CPUs

ETHERNET INTERFACE ONLY

PROFIBUS MPI/DP IS NOT SUPPORTED

Documentation last update: August 28, 2017

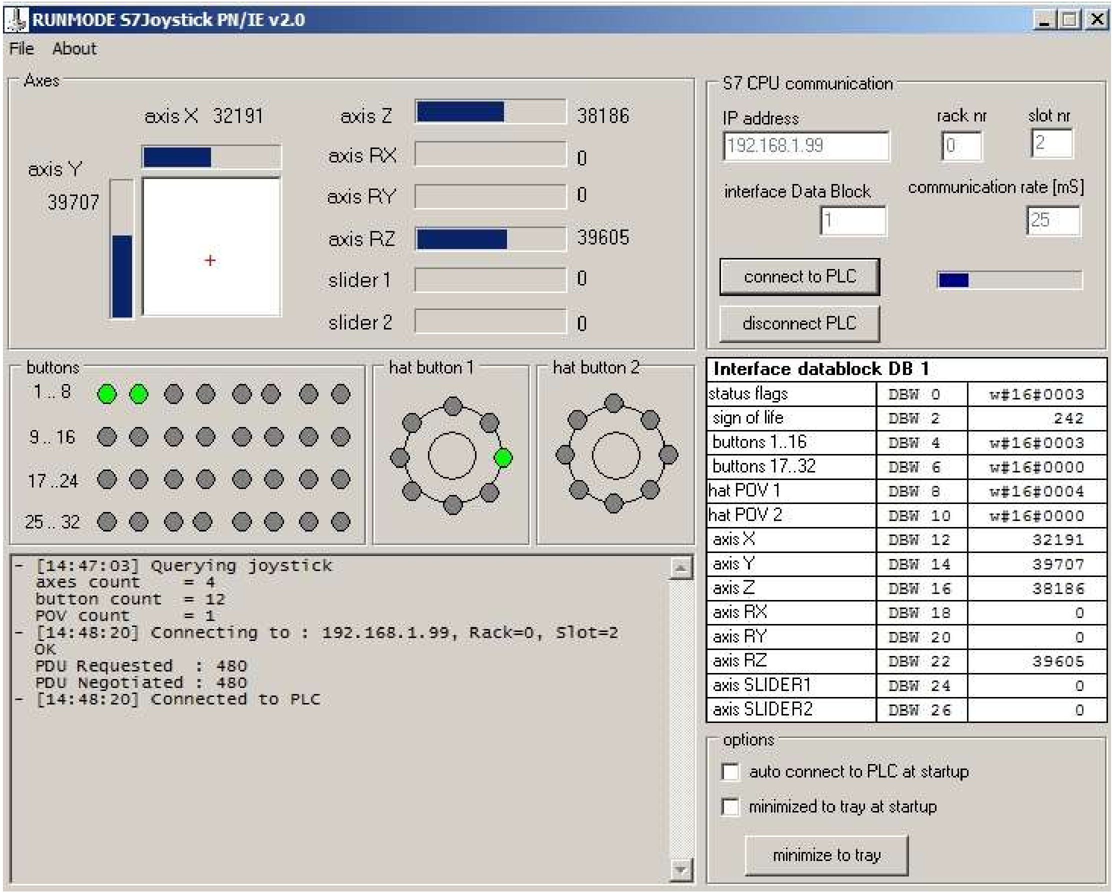
Copyright Luca Gallina
RUNMODE
Industrial Automation Software
Via C. B. Cavour, 7
31040 Volpago del Montello (TV)
ITALY
www.runmode.com

1 Contents

2 RUNMODE S7JOYSTICK FEATURES

The RUNMODE S7Joystick is a software tool that reads joystick axes position and buttons state and send them to the PLC memory.

A data block must be created in the PLC memory to receive the joystick data.



axis X 32191
axis Y 39707
axis Z 38186
axis RX 0
axis RY 0
axis RZ 39605
slider 1 0
slider 2 0

IP address: 192.168.1.99
rack nr: 0
slot nr: 2
interface Data Block: 1
communication rate [mS]: 25

```
- [14:47:03] Querying joystick
  axes count    = 4
  button count  = 12
  POV count     = 1
- [14:48:20] Connecting to : 192.168.1.99, Rack=0, Slot=2
  OK
  PDU Requested  : 480
  PDU Negotiated : 480
- [14:48:20] Connected to PLC
```

Interface datablock DB 1		
status flags	DBW 0	w#16#0003
sign of life	DBW 2	242
buttons 1..16	DBW 4	w#16#0003
buttons 17..32	DBW 6	w#16#0000
hat POV 1	DBW 8	w#16#0004
hat POV 2	DBW 10	w#16#0000
axis X	DBW 12	32191
axis Y	DBW 14	39707
axis Z	DBW 16	38186
axis RX	DBW 18	0
axis RY	DBW 20	0
axis RZ	DBW 22	39605
axis SLIDER1	DBW 24	0
axis SLIDER2	DBW 26	0

☐ auto connect to PLC at startup
☐ minimized to tray at startup

3 Connecting to the PLC

The RUNMODE S7 DBtoCSV utility works with Ethernet/Profinet connections only.

Communication based on Profibus MPI / DP is not supported.

3.1 Connecting to an S7-300/-400 CPU

The connection to S7 -300/-400 CPUs does not need any action, just create the necessary datablocks in the PLC memory.

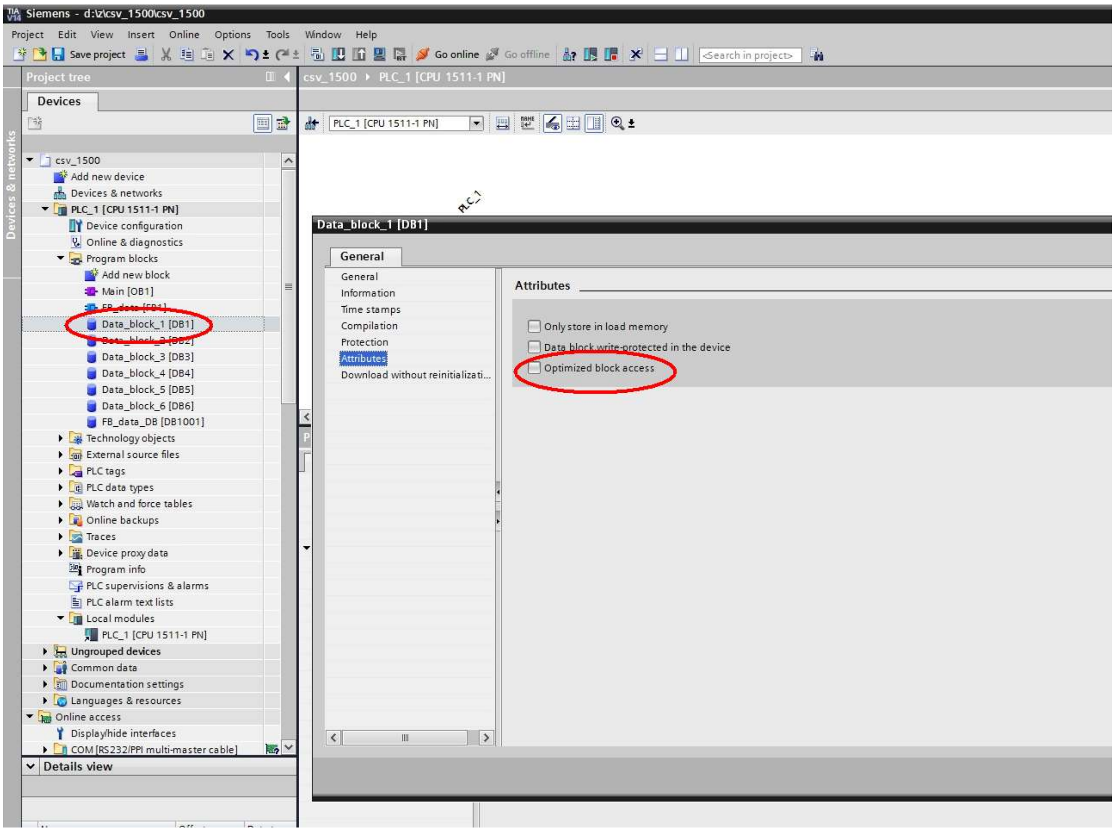
3.2 CONNECTING TO A TIA S7-1200/-1500 CPU

Connecting the "RUNMODE S7 DBtoCSV" to TIA S7-1200/-1500 need some actions both on the datablocks and the CPU itself. Without these actions the communication with a 1200/1500 PLC will not work.

3.2.1 Step 1, remove the "optimized" property

Due to the memory model, which differs from traditional Step7 -300/-400 CPUs, in TIA 1200/1500 CPUs the datablocks hosting the variables to be converted into CSV values must be set as "not optimized". In this way, all the datablock variables will be stored side by side within the PLC memory.

Ensure to remove the "optimized block access" from all the DBs that will be accessed by the "RUNMODE S7 DBtoCSV" utility.



Attributes

☐ Only store in load memory

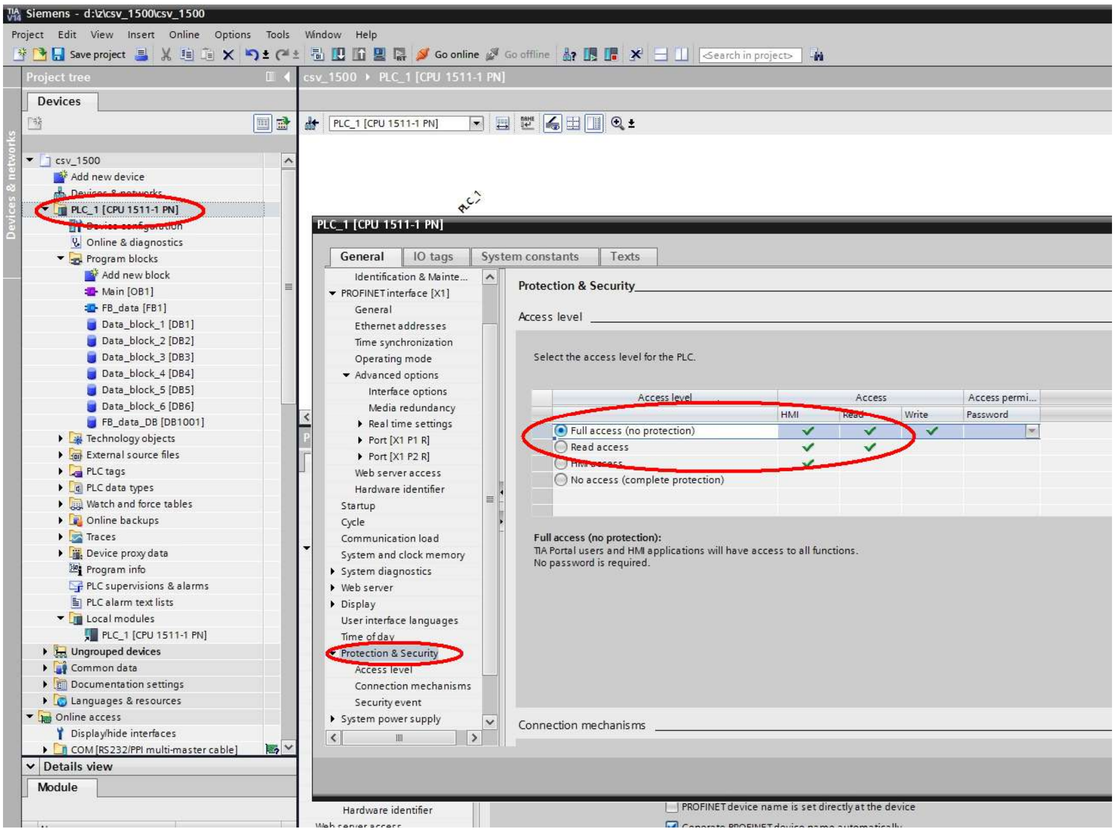
☐ Data block write-protected in the device

☐ Optimized block access

Hint: To save "work memory" space you may select the also "only store to load memory" option.

3.2.2 Step 2, allow data to be read from the CPU

In the CPU properties you must grant READ data access



Protection & Security

Access level

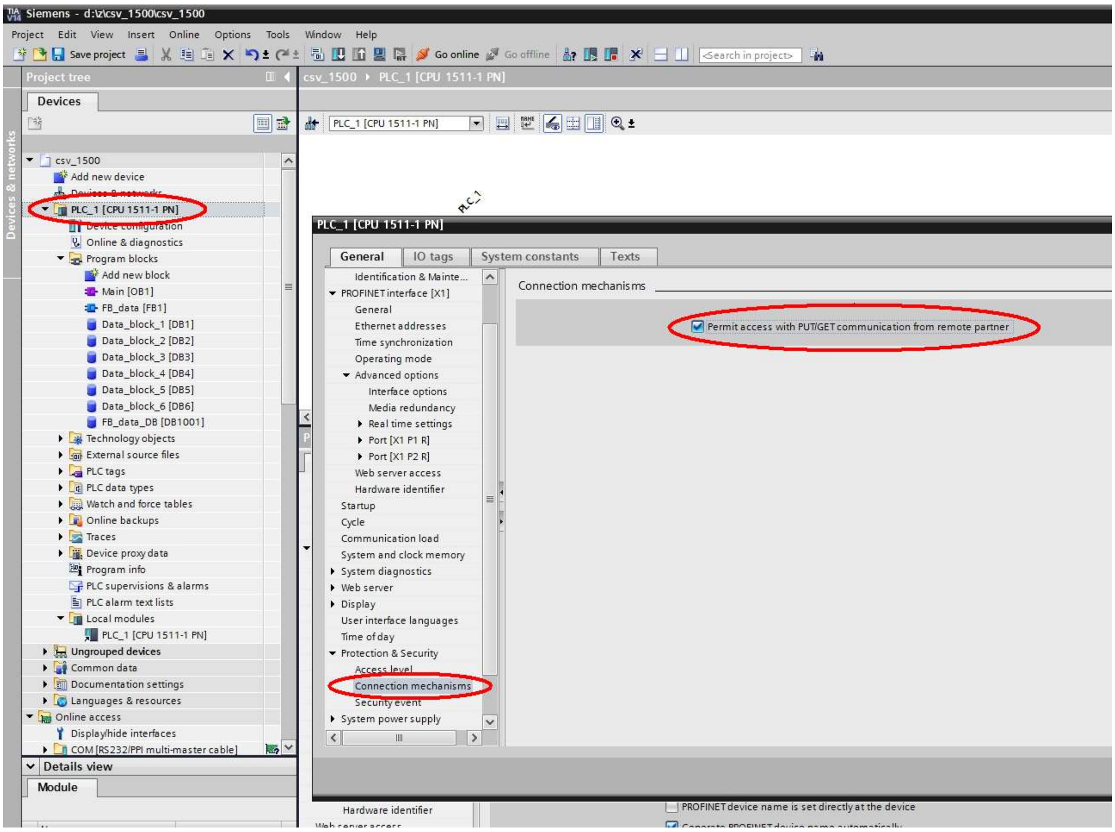
Select the access level for the PLC.

Access level	Access			Access permi...
	HMI	Read	Write	Password
● Full access (no protection)	✓	✓	✓	
○ Read access	✓	✓		
○ HMI access	✓			
○ No access (complete protection)				

Full access (no protection):
TIA Portal users and HMI applications will have access to all functions.
No password is required.

3.2.3 Step 3, allow GET/PUT communication with other partners

You must also grant GET/PUT memory access



Connection mechanisms

☑ Permit access with PUT/GET communication from remote partner

4 RUNNING THE S7JOYSTICK UTILITY

4.1 DATA CREATION IN PLC MEMORY

Prior to running the utility, you must create in the PLC a datablock to receive the joystick data. The datablock must be formatted as follows:

```
DATA_BLOCK "Joystick"
TITLE =
VERSION : 0.1


  STRUCT
   StatusFlags : STRUCT
    spare : BOOL ;
    spare1 : BOOL ;
    spare2 : BOOL ;
    spare3 : BOOL ;
    spare4 : BOOL ;
    spare5 : BOOL ;
    spare6 : BOOL ;
    spare7 : BOOL ;
    JoystickActive : BOOL ;   //TRUE=joystick is detected by Windows
    JoystickDataValid : BOOL ;       //TRUE=incoming joystick data are valid
    spare10 : BOOL ;
    spare11 : BOOL ;
    spare12 : BOOL ;
    spare13 : BOOL ;
    spare14 : BOOL ;
    spare15 : BOOL ;
    spare16 : BOOL ;
   END_STRUCT ;
   SignOflife : INT ;  //telegram counter
   Buttons : STRUCT
    Button25 : BOOL ;  //TRUE=button pressed
    Button26 : BOOL ;  //TRUE=button pressed
    Button27 : BOOL ;  //TRUE=button pressed
    Button28 : BOOL ;  //TRUE=button pressed
    Button29 : BOOL ;  //TRUE=button pressed
    Button30 : BOOL ;  //TRUE=button pressed
    Button31 : BOOL ;  //TRUE=button pressed
    Button32 : BOOL ;  //TRUE=button pressed
    Button17 : BOOL ;  //TRUE=button pressed
    Button18 : BOOL ;  //TRUE=button pressed
    Button19 : BOOL ;  //TRUE=button pressed
    Button20 : BOOL ;  //TRUE=button pressed
    Button21 : BOOL ;  //TRUE=button pressed
    Button22 : BOOL ;  //TRUE=button pressed
    Button23 : BOOL ;  //TRUE=button pressed
    Button24 : BOOL ;  //TRUE=button pressed
    Button9 : BOOL ;   //TRUE=button pressed
    Button10 : BOOL ;  //TRUE=button pressed
    Button11 : BOOL ;  //TRUE=button pressed
    Button12 : BOOL ;  //TRUE=button pressed
    Button13 : BOOL ;  //TRUE=button pressed
    Button14 : BOOL ;  //TRUE=button pressed
    Button15 : BOOL ;  //TRUE=button pressed
    Button16 : BOOL ;  //TRUE=button pressed
    Button1 : BOOL ;   //TRUE=button pressed
    Button2 : BOOL ;   //TRUE=button pressed
    Button3 : BOOL ;   //TRUE=button pressed
    Button4 : BOOL ;   //TRUE=button pressed
    Button5 : BOOL ;   //TRUE=button pressed
    Button6 : BOOL ;   //TRUE=button pressed
    Button7 : BOOL ;   //TRUE=button pressed
    Button8 : BOOL ;   //TRUE=button pressed
   END_STRUCT ;
   Hat1_POV : STRUCT
    spare : BOOL ;
    spare1 : BOOL ;
    spare2 : BOOL ;
    spare3 : BOOL ;
    spare4 : BOOL ;
    spare5 : BOOL ;
    spare6 : BOOL ;
    spare7 : BOOL ;
    North : BOOL ;     //TRUE=button pressed
    NorthEast : BOOL ; //TRUE=button pressed
    East : BOOL ;      //TRUE=button pressed
    SouthEast : BOOL ; //TRUE=button pressed
    South : BOOL ;     //TRUE=button pressed
    SouthWest : BOOL ; //TRUE=button pressed
    West : BOOL ;      //TRUE=button pressed
    NorthWest : BOOL ; //TRUE=button pressed
   END_STRUCT ;
   Hat2_POV : STRUCT
    spare : BOOL ;
    spare1 : BOOL ;
    spare2 : BOOL ;
    spare3 : BOOL ;
    spare4 : BOOL ;
    spare5 : BOOL ;
    spare6 : BOOL ;
    spare7 : BOOL ;
    North : BOOL ;     //TRUE=button pressed
    NorthEast : BOOL ; //TRUE=button pressed
    East : BOOL ;      //TRUE=button pressed
    SouthEast : BOOL ; //TRUE=button pressed
    South : BOOL ;     //TRUE=button pressed
    SouthWest : BOOL ; //TRUE=button pressed
    West : BOOL ;      //TRUE=button pressed
    NorthWest : BOOL ; //TRUE=button pressed
   END_STRUCT ;
   AxisX : WORD ;      //0..65535, 32767=center position
   AxisY : WORD ;      //0..65535, 32767=center position
   AxisZ : WORD ;      //0..65535, 32767=center position
   AxisRX : WORD ;     //0..65535, 32767=center position
   AxisRY : WORD ;     //0..65535, 32767=center position
   AxisRZ : WORD ;     //0..65535, 32767=center position
   AxisSLIDER1 : WORD ;       //0..65535
   AxisSLIDER2 : WORD ;       //0..65535
  END_STRUCT ;
```

4.2 JOYSTICK DATA DETAILS

4.2.1 StatusFlags

The status flags tell you the actual state of the joystick and must be evaluated by the PLC program in order to validate the data sent by the Windows PC.

- JoystickActive : if TRUE, the joystick is connected to the PC and correctly detected by Windows.
- JoystickDataValid : if TRUE, the joystick data sent by the PC are valid. Since Windows works on events, even if the joystick has been detected by windows the joystick position data are not valid until at least an axis has been moved or a button has been pressed.

NOTE: In your PLC program you MUST evaluate the JoystickDataValid flag, e.g. you will STOP all your actions if data is not valid.

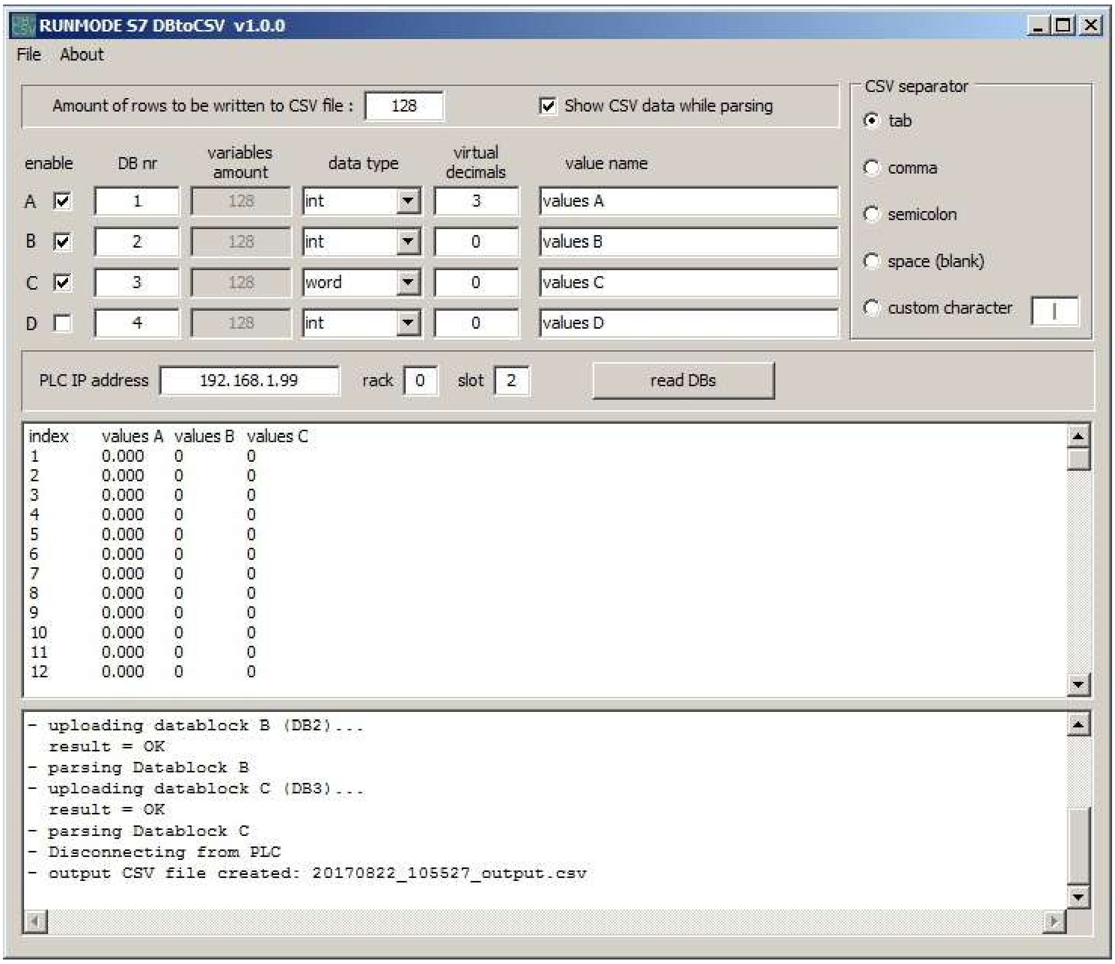
4.2.2 SignOfLife

The sign-of-life is actually the data transmission counter: each time the utility sends joystick data, the counter is increased by one. You can then monitor the counter in order to check if the joystick data in updated regularly.

NOTE: In your PLC program you MUST evaluate the SignOfLife counter, e.g. you will STOP all action

While datablocks will be made of a specific variable type (e.g. array of bytes, array of integers, array of floats, etc.) according to the PLC programming, the S7DBtoCSV will read the data in raw mode; therefore it is not important whether the data type declared in the datablocks is the same defined in S7DBtoCSV.

4.3 DATA READING AND CSV FILE CREATION



Amount of rows to be written to CSV file : 128

☑ Show CSV data while parsing

CSV separator: ● tab ○ comma ○ semicolon ○ space (blank) ○ custom character |

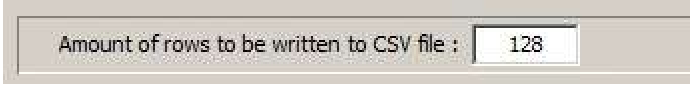
	enable	DB nr	variables amount	data type	virtual decimals	value name
A	☑	1	128	int	3	values A
B	☑	2	128	int	0	values B
C	☑	3	128	word	0	values C
D	□	4	128	int	0	values D

PLC IP address 192.168.1.99 rack 0 slot 2

```
index   values A  values B  values C
1       0.000     0         0
2       0.000     0         0
3       0.000     0         0
4       0.000     0         0
5       0.000     0         0
6       0.000     0         0
7       0.000     0         0
8       0.000     0         0
9       0.000     0         0
10      0.000     0         0
11      0.000     0         0
12      0.000     0         0
```

```
- uploading datablock B (DB2)...
  result = OK
- parsing Datablock B
- uploading datablock C (DB3)...
  result = OK
- parsing Datablock C
- Disconnecting from PLC
- output CSV file created: 20170822_105527_output.csv
```

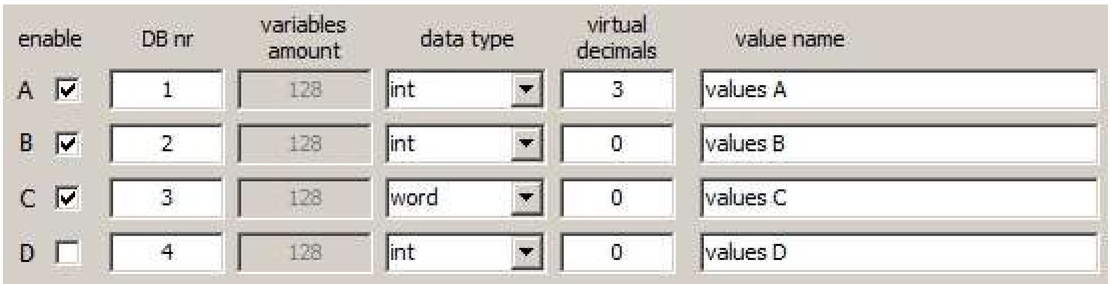
Set here the amount of rows to be included in the CSV file, in other words the amount of values to be read from the PLC. The datablocks in the PLC memory must be obviously large enough to contain the indicated amount of values.



Amount of rows to be written to CSV file : 128

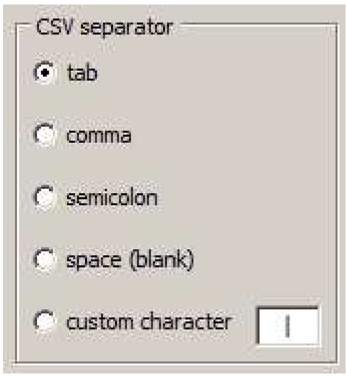
Set here which datablocks (A, B, C, D) you want to read and parse.

- “Enable” option will quickly include or exclude the DB from the reading procedure.
- “DB nr” is the number of the datablock to be read
- “variables amount” corresponds to the amount of lines to be included in the CSV file
- “data type” istructs the utility on the nature and size of the variables to be read. Allowed types are BYTE, INT, WORD, DINT, DWORD, REAL.
- “virtual decimals” allows to add a decimal point to the output value (e.g. integer value 12345 can be printed as 12.345 if “virtual decimals” is set to 3).
- “value name” is the name you want to be printed in the CSV file as value identifier (e.g. “speed”, “torque”, “position”, etc.).



	enable	DB nr	variables amount	data type	virtual decimals	value name
A	☑	1	128	int	3	values A
B	☑	2	128	int	0	values B
C	☑	3	128	word	0	values C
D	□	4	128	int	0	values D

The “CSV separator” allows you to selet the values separator in the output CSV file.



CSV separator

● tab

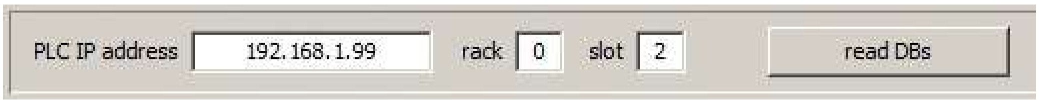
○ comma

○ semicolon

○ space (blank)

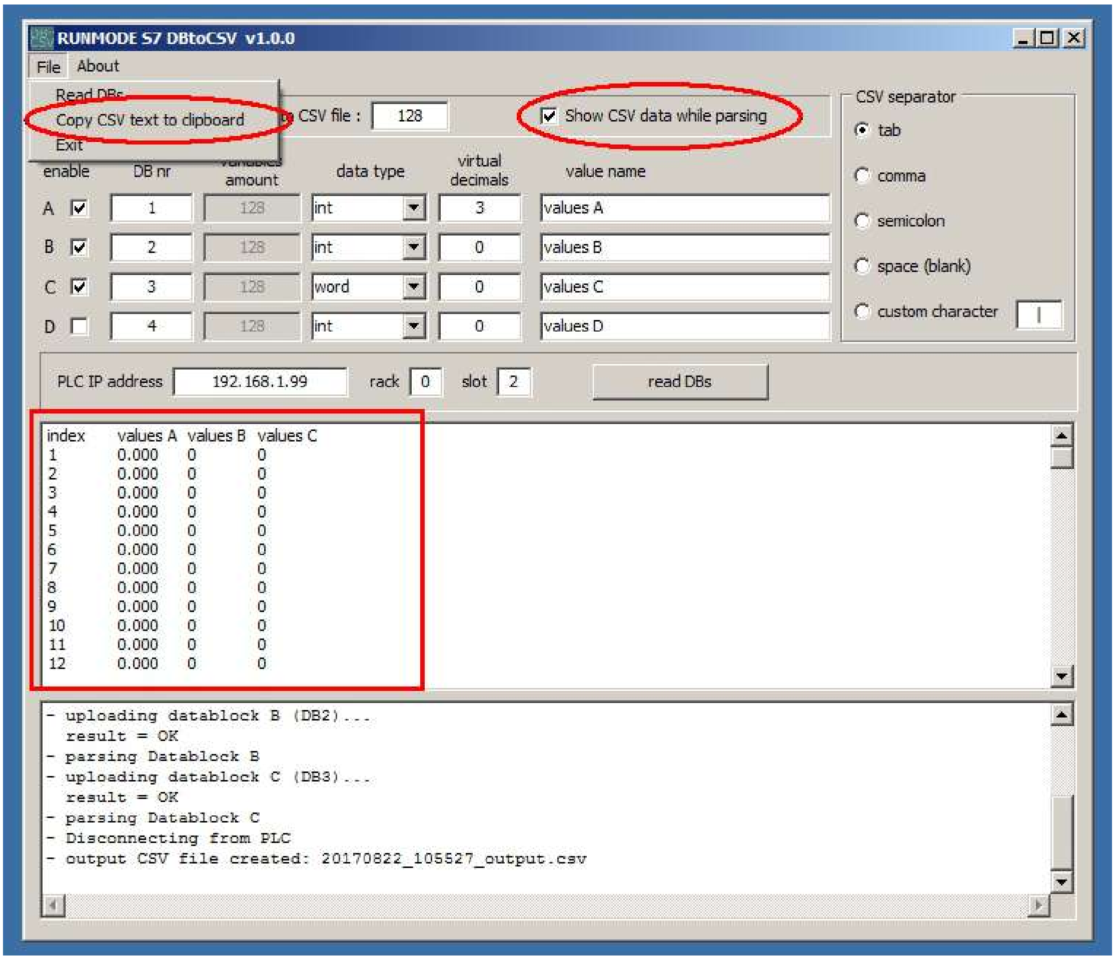
○ custom character |

Set here the IP address of the PLC, along with rack and slot indication. The "read DBs" butto wil start the reading and parsing process.



PLC IP address 192.168.1.99 rack 0 slot 2

If "Show CSV data while parsing" option is selected, the CSV data is displayed on screen during the parsing process and the "copy to clipboard" function is enabled.
If "Show CSV data while parsing" is not selected, no data is displayed on screen and the parsing process is much faster.
NOTE: the "index" column in the CSV file is generated automatically.



CSV file : 128

☑ Show CSV data while parsing

CSV separator
● tab
○ comma
○ semicolon
○ space (blank)
○ custom character |

	enable	DB nr	variables amount	data type	virtual decimals	value name
A	☑	1	128	int	3	values A
B	☑	2	128	int	0	values B
C	☑	3	128	word	0	values C
D	□	4	128	int	0	values D

PLC IP address 192.168.1.99 rack 0 slot 2

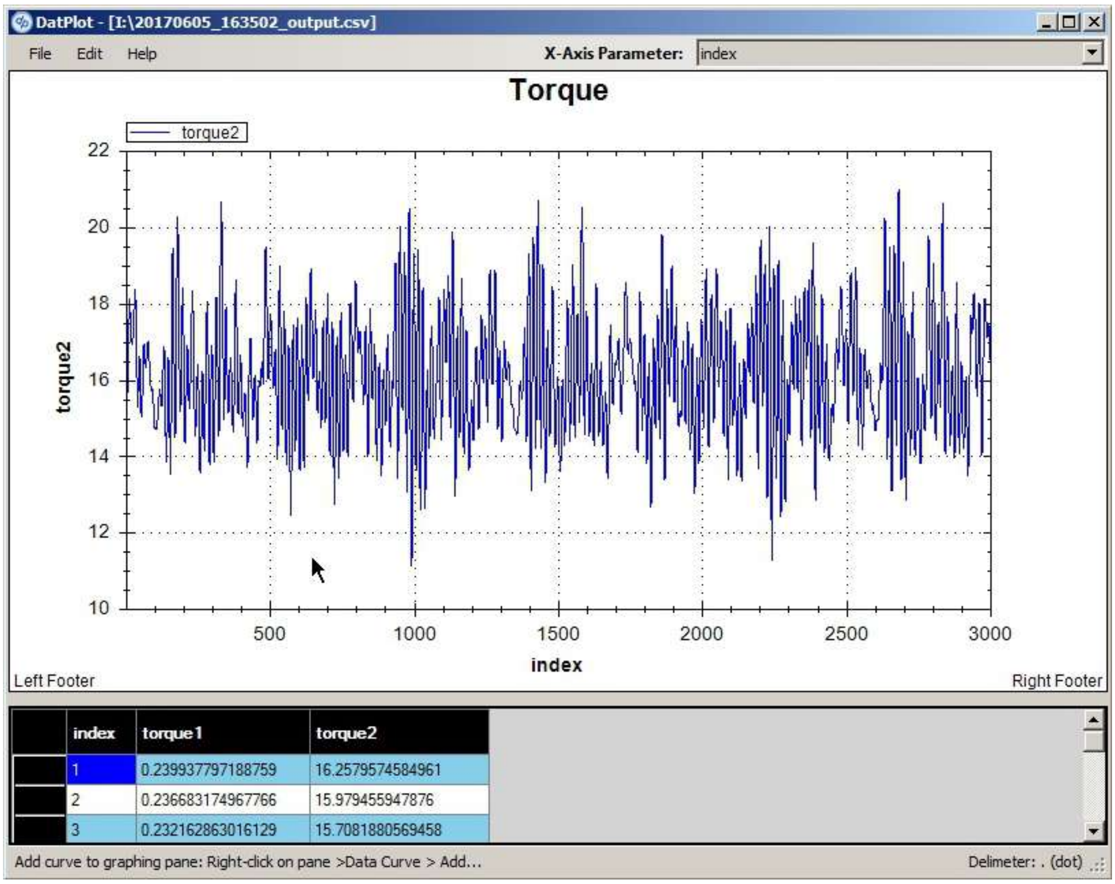
```
index   values A  values B  values C
1       0.000     0         0
2       0.000     0         0
3       0.000     0         0
4       0.000     0         0
5       0.000     0         0
6       0.000     0         0
7       0.000     0         0
8       0.000     0         0
9       0.000     0         0
10      0.000     0         0
11      0.000     0         0
12      0.000     0         0
```

```
- uploading datablock B (DB2)...
  result = OK
- parsing Datablock B
- uploading datablock C (DB3)...
  result = OK
- parsing Datablock C
- Disconnecting from PLC
- output CSV file created: 20170822_105527_output.csv
```

5 ANALYZING AND PLOTTING THE DATA

The generated CSV files can be imported and analyzed with any suitable software applications. While many are used to manipulate data using Microsoft Excel, I suggest to take a look at the freeware "DatPlot" application developed by Michael Vogt and available at the following link http://www.datplot.com/

Do not miss the video presentation at http://www.datplot.com/features/



index	torque1	torque2
1	0.239937797188759	16.2579574584961
2	0.236683174967766	15.979455947876
3	0.232162863016129	15.7081880569458

Add curve to graphing pane: Right-click on pane >Data Curve > Add...

Delemeter: . (dot)

6 CREDITS

This application uses SNAP7 communication library developed by Davide Nardella, who also edited a beautiful documentation on Siemens Step-7 native communication.

Check http://snap7.sourceforge.net/

7 LEGAL NOTES

The enclosed computer program (“Software”) is licensed, not sold, to you by the author for use only under the terms of this License. You own the media on which the Software is recorded or fixed, but the author retain ownership of the Software itself.

The Software is registered to each user by means of a unique license file. Illegal copies can therefore be tracked. Free distribution and upload to Internet is strictly forbidden.

7.1 LICENSE

This License allows you to:

- Use one copy of the Software on a single computer at a time. To “use” the Software means that the Software is either loaded in the temporary memory (i.e., RAM) of a computer and/or installed on the permanent memory of a computer (i.e., hard disk, etc.).

7.2 DISCLAIMER

The author is not liable for any use of the Software and takes no responsibilities for damages of any kind.