

TCPSimDBG 1.0 (beta) - AVR Studio Plug-in

License:

This plug-in was developed under AVR Studio SDK License, so no violation or abuse is permitted.

You have to read the ATMEL License Agreement ("AVR SDK License agreement.pdf").

This build was made for sharing development with AVR users (not in the point of view source-code).

Contact informations:

Developer: Ahmed Jouirou, Electrical/ Micro-electronics Eng.

Location: Montreal – Canada

E -mail: ahmed.jouirou@hotmail.com

AvrFreaks user: ahmedjouirou

Description:

TCPSimDBG (TCP Simulation and Debugging) , is an AVR Studio plug-in , that allow AVR users to debug their code remotely and to create their own simulation models that communicate with AVR Studio over IP, so they can simulate TFT/LCDs , MOTOs, LEDs, BUTTONs.....

These models are not a subject of development / are not included in this plug-in , so the user can use TCP commands (listed in this document) to read and write , all registers , memories while debugging, and can also execute some commands like "Start debugging" ,"Stop debugging".

Installation:

You have to close AVR Studio before performing the plug-in installation.

All you have to do about installation is to put these files:

```
"AvrPluginTCPSimDBG.dll"  
"TCPSimDBG_RECONNECT.bmp"
```

in the same directory, (let's say for example in "C:\TCPSimDBG\").

Now you have to register it, to do this, you have to run windows command line (cmd.exe) as administrator privilege, and enter this command:

```
regsvr32 C:\TCPSimDBG\AvrPluginTCPSimDBG.dll
```

After that it is recommended to set up windows registers to configure the port/address of the server that the **TCPSimDBG** will connect to. To do this, you only have to execute the following file:

```
"TCPSimDBG_regs.reg"
```

It will place two keys in the registry of windows (you can see them using "regedit.exe" of windows) in the following location **[HKEY_LOCAL_MACHINE\SOFTWARE\Atmel\AVRTools]**

```
"AvrPlugin_TCPSimDBG_IP"="127.0.0.1"
```

```
"AvrPlugin_TCPSimDBG_PORT"="888"
```

Notice: if you the plug-in will not find these file in the registry, it will use its default port/address (127.0.0.1 : 333)

Now you are ready to execute AVR Studio, and communicate with the plug-in. ENJOY!

NOTE: To uninstall completely the plug-In from you system, you have to close AVR Studio, then run windows command line (cmd.exe) as administrator privilege, and enter this command:

```
regsvr32 /U C:\TCPSimDBG\AvrPluginTCPSimDBG.dll
```

Then you can delete the files that you added (C:\TCPSimDBG\AvrPluginTCPSimDBG.dll).

Or to disable the plug-in, you can use the AVR Studio Plug-in Manager to deactivate it.

How to use:

After performing the plug-in installation, you are ready to begin with remote debugging simulating with **TCPSimDBG**.

When AVR Studio starts, you have to make for example a new Assembler project, with a simulation platform.

You have to load a TCP server- in listening- (i use “Hercules Setup” multi terminal from www.HW-group.com)

After opening your project, you will see a new icon in the AVRStudio toolbar (and the same in the menu->tools-> **TCPSimDBG**->Reconnect), when clicking on this icon, the plug-in will force to connect on a TCP server (IP and port as specified in the registry configuration or by default).

Each time you click on “Reconnect” the plug-in will disconnect and reconnect (even if it was not connected before).

Once connected to the server, (you will be notified in the message window of AVR Studio as a warning) you can use the commands in the tables below to perform data transfer between AVR Studio and the TCP terminal (example “Hercules Setup” terminal).

When you are in the debugging mode of AVR Studio:

	Application to AVR Studio (HEX)	AVR Studio to Application (HEX)
Is target present	01	1byte (00 01) [see Note 1]
Get emul/sim state	02	1byte (01 02 04 08) [see Note 2]
Write data memory	03 [aa aa aa aa] [dd]	
Read data memory	04 [aa aa aa aa]	
Write program memory	05 [aa aa aa aa] [dd]	
Read program memory	06 [aa aa aa aa]	
Write EEPROM memory	07 [aa aa aa aa] [dd]	
Read EEPROM memory	08 [aa aa aa aa]	
Write Cycle Counter	09 [dd dd dd dd]	
Read Cycle Counter	0A	4bytes([dd dd dd dd])
Write Program Counter	0B [dd dd dd dd]	
Read Program Counter	0C	4bytes([dd dd dd dd])
Run	0D	
Single step	0E [00 01]	
Step over	0F [00 01]	
Step out	10 [00 01]	
Goto address	11 [aa aa aa aa]	
Stop	12 [00 01]	
Reset	13 [00 01]	

When you are in or not in the debugging mode of AVR Studio :

	Application to AVR Studio (HEX)	AVR Studio to Application (HEX)
Is in debug mode	30	1byte (00 01) [see Note 1]
Start debug	31	
Stop debug	32	
Get Part Name	33	String (ex : ATmega16)
Get Target Name	34	String (ex : AVR Simulator)

Note1 : 00 means TRUE , 01 means FALSE.

Note2: 01 : Target is in stop mode

02 : Target is in run mode

04 : Target is in sleep mode

08 : Target is busy, (ex: configuring)