Chipsmall

Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

JTAG-Booster for

Intel 386EX

Copyright © 1995..2002:

FS FORTH-SYSTEME GmbH
Postfach 1103, D-79200 Breisach, Germany

Release of Document: May 15, 2002
Author: Dieter Fögele
Filename: JTAG_386EXa.doc
Program Version: 4.xx

Table of Contents

1. General

The program JTAG386.EXE uses the JTAG port of the Intel 386EX embedded microprocessor in conjunction with the small JTAG-Booster:

- to program data into flash memory
- to verify and read the contents of a flash memory
- to load data to static memory
- to upload data from target to host
- to do a memory test
- to make a memory dump
- to access an I²C Device
- to test CPU signals

All functions are done without any piece of software running in the target. No firmware or BIOS must be written. Bootstrap software may be downloaded to initially unprogrammed memories.

For latest documentation please refer to the file README.TXT on the distribution disk.

1.1.   Ordering Information

The following related products are available

- 925   JTAG-Booster Intel 386EX, 5V, DOS/Win9x/WinNT,
  delivered with adapter type 227

- 977   JTAG-Booster Intel 386EX, 3.3V, DOS/Win9x/WinNT,
  delivered with adapter type 285

1.2.   System Requirements

To successfully run this tool the following requirements must be met:

- MSDOS, WIN3.x, WIN9x, WinME, WinNT or Win2000
  (WinNT/Win2000 is supported with an additional tool, see chapter 5)

- Intel 80386 or higher

- 205 kByte of free DOS memory

- Parallel Port

1.3.   Contents of Distribution Disk

- JTAG386.EXE   Tool for Intel 386EX
  JTAG386.OVL

- JTAG386.INI   Template configuration file for Intel 386EX. See chapter 1.8 "Initialization file JTAG386.INI"

- HEX2BIN.EXE   Converter program to convert Intel HEX and Motorola S-Record files to binary. See chapter 4 "Converter Program HEX2BIN.EXE"

- WinNT   Support for Windows NT and Windows 2000. See chapter 5 "Support for Windows NT and Windows 2000"

- JTAG_V4xx_FLAS   List of all supported Flash devices
  HES.pdf

- README.txt   Release notes, new features, known problems

1.4.   Connecting your PC to the target system

The JTAG-Booster can be plugged into standard parallel ports (LPT1-3) with a DB25-Connector.

LPT1-3
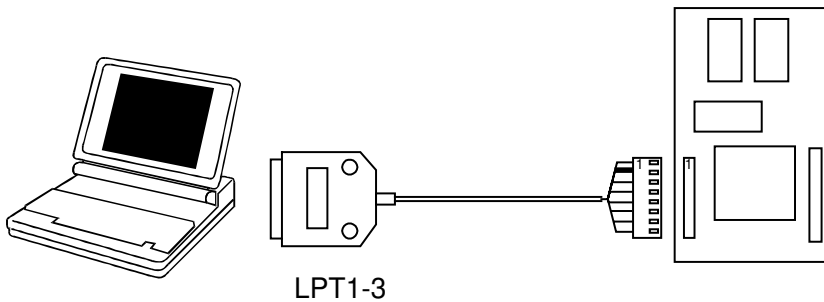
The target end of the cable has the following reference:

1	2*	3	4	5	6	7	8
TCK	GND	TMS	TRST#	NC	TDI	TDO	+3.3V / +5V

*PIN 2 can be detected by the white thick cable.

To connect your design to the JTAG-BOOSTER you need a single row berg connector with a spacing of 2.54mm on your PCB. The names refer to the target: Pin 7 is the target's TDO pin and is connected to the JTAG-Booster's TDI pin.

There are two versions of the JTAG-Booster available: A 5V version (FS part number 227) and a 3.3V version (FS part number 285). **Don't apply 5V to the 3.3V version of the JTAG-Booster!**

Your target must be able to power the JTAG-Booster, it draws about 100mA.

Before you start the program, the JTAG-BOOSTER must be plugged to a parallel interface of your PC and to the 8 pin JTAG connector on the target.

The utility is started with the general command line format:

JTAG386 /function [filename] [/option_1] ... [/option_n].

Note that the function must be the first argument followed (if needed) by the filename.

If you want to cancel execution of JTAG386, press CTRL-Break-Key.

On any error the program aborts with an MSDOS error level of one.

1.5.   First Example with Intel 386EX

In the following simple example it is assumed that the JTAG-Booster is connected to LPT1 of your PC and target power is on.

Typing

        JTAG386 /P MYAPP.BIN

at the DOS prompt results in the following output:

        JTAG386 --- JTAG utility for Intel 386EX
        Copyright © FS FORTH-SYSTEME GmbH, Breisach
        Version 4.xx of mm/dd/yyyy

(1)   Configuration loaded from file JTAG386.INI
(2)   Target: Module 386EX, FS FORTH-SYSTEME GmbH
(3)   Using LPT at I/O-address 0378h
(4)   JTAG Adapter detected

(5)   1 Device detected in JTAG chain
            Device 0: IDCODE=28270013   Intel 386EXTC (C-Step, 5V)
(6)   Sum of instruction register bits   : 4
(7)   CPU position   : 0
(8)   Instruction register offset   : 0
(9)   Length of boundary scan reg   : 202

        Looking for a known flash device. Please wait..
(10)   AMD 29F800T, Boot Block Top detected
(11)   Bus size is 16 Bit
(12)   Erasing Flash-EPROM Block #:0
        Programming File MYAPP.BIN
        65536 Bytes programmed
        Programming ok

        Erase Time   :   1.0 sec
        Programming Time   :   48.1 sec

(1)   The initialization file JTAG386.INI was found in the current directory.

(2)   The target identification line of the initialization file is printed here.

(3)   The resulting I/O-address of the parallel port is printed here.

(4)   A JTAG-Booster is found on the parallel port

(5)   The JTAG chain is analyzed. There may be several parts in the JTAG chain. The chain is analyzed and all parts except the Intel 386EX are switched to bypass mode.

(6)   The length of all instruction registers in the JTAG chain are added.

(7)   The position of the Intel 386EX in the JTAG chain is checked.

(8)   The position of the JTAG instruction register of the Intel 386EX is checked

(9)   The real length of the boundary scan register is displayed here and compared with the boundary scan register length of a Intel 386EX.

(10)   A Flash-EPROM AMD 29F800T selected with chip select UCS was found.

(11)   The resulting data bus size is printed here.

(12)   In this example one block must be erased.

1.6.   Trouble Shooting

Avoid long distances between your Host-PC and the target. If you are using standard parallel extension cable, the JTAG-BOOSTER may not work. Don't use Dongles between the parallel port and the JTAG-BOOSTER.

Switch off all special modes of your printer port (EPP, ECP, ...) in the BIOS setup. Only standard parallel port (SPP) mode is allowed.

On very fast PCs there could be verify errors. To avoid this, watch for the ´IO recovery time´-switch in the BIOS Setup which must be turned on. Otherwise try to slow down your PC by setting the turbo switch off.

If there are problems with autodetection of the flash devices use the /DEVICE= option. To speed up autodetection specify one of the options /8BIT or /16BIT.

Don't use hardware protected flash memories.

The used chip selects must be defined as output and inactive in the initialization file (see chapter 1.8 "Initialization file JTAG386.INI"). Also the address bits must be defined as output.

Use the option /NOWRSETUP to speed up flash programming.

1.7.   Error Messages

- **80386 or greater required**
  The JTAG-BOOSTER does not work on a 8088/8086 or a 80286 platform.

- **Adapter not connected or target power fail**
  The JTAG-Booster wasn't found. Please check connection to parallel port
  and connection to target. Check target power. Check your BIOS-Setup.

- **Can't open x:\yyy\zzz\JTAG386.OVL**
  The overlay file JTAG386.OVL must be in the same directory as
  JTAG386.EXE.

- **Configuration file XYZ not found.**
  The file specified with the option /INI= wasn't found.

- **Device offset out of range**
  The value specified with the option /OFFSET= is greater than the size of
  the detected flash device.

- **Disk full**
  Writing a output file was aborted as a result of missing disk space.

- **Do not specify option /NOCS with any other chip select**
  There is a conflict in the command line.

- **Do not specify option /BYTE-MODE. *Flash device* does not have a byte
  mode pin.**
  The flash device specified with the option /DEVICE= does not support
  switching between 16 (or 32) bit mode and 8 bit mode. In practice it does not
  have a pin with the name BYTE#

- **Error creating file:**
  The output file could not be opened. Please check free disk space or write
  protection.

- **Error: *Pin-Name* is an output only pin**
  The specified pin cannot be sampled. Check the command line. Check the
  initialization file.

- **Error: *Pin-Name* is an input only pin**
  The specified pin cannot be activated. Check the command line. Check the initialization file.

- **Error: *Pin-Name* may not be read back**
  The specified pin can be switched to tristate, but cannot be read back. Check the command line.

- **illegal function:**
  The first parameter of the command line must be a valid function. See chapter 2 "JTAG386 Parameter Description" for a list of supported functions.

- **illegal number:**
  The specified number couldn't be interpret as a valid number. Check the relevant number base.

- **illegal option:**
  See chapter 2 "JTAG386 Parameter Description" for a list of supported options.

- **illegal Pin Type:**
  The name specified with the option /PIN= must be one of the list of chapter 1.8 "Initialization file JTAG386.INI"

- **illegal Flash Type:**
  The name specified with the option /DEVICE= must be one of the list of chapter 1.9 "Supported flash devices".

- **Input file not found:**
  The specified file cannot be found

- **Input file is empty:**
  Files with zero length are not accepted

- **"   " is undefined**
  Please check the syntax in your configuration file. (See chapter 1.8 "Initialization file JTAG386.INI").

- **LPTx not installed**
  The LPT port specified with /LPTx cannot be found. Please check the LPT port or specify a installed LPT port. Check your BIOS setup.

- **missing filename**
  Most functions need a filename as second parameter.

- **missing option /I2CCLK=**
  Some functions need the option /I2CCLK= to be defined.

- **missing option /I2CDAT=**
  Some functions need the option /I2CDAT= or the options /I2CDATO= and /I2CDATI= to be defined.

- **missing option /LENGTH=**
  Some functions need the option /LENGTH= to be defined.

- **missing option /PIN=**
  Some functions need the option /PIN= to be defined.

- **More than 9 devices in the JTAG chain or TDI pin stuck at low level**
  The JTAG chain is limited to 9 parts. Check target power. Check the target's TDO pin.

- **No devices found in JTAG chain or TDI pin stuck at high level**
  A stream of 32 high bits was detected on the pin TDI. TDI may stuck at high level. Check the connection to your target. Check the target power. Check the target's TDO pin.

- **Option /CPUPOS= out of range**
  The number specified with the option /CPUPOS= must be less or equal to the number of parts minus 1.

- **Option /IROFFS= out of range**
  Please specify a smaller value

- **Part at specified position is not a Intel 386EX**
  The option /CPUPOS= points to a part not a Intel 386EX

- **Pins specified with /I2CCLK= and /I2CDAT= must have different control cells**
  The pin specified with the option /I2CDAT= must be able to be switched to high impedance while the pin specified with option /I2CCLK= is an active output. See chapter 1.8 "Initialization file JTAG386.INI".

- **Pins specified with /I2CCLK= and /I2CDATI= must have different control cells**
  The pin specified with the option /I2CDATI= must be able to be switched to high impedance while the pin specified with option /I2CCLK= is an active output. See chapter 1.8 "Initialization file JTAG386.INI".

- **Pins specified with /I2CDATO= and /I2CDATI= must have different control cells**
  The pin specified with the option /I2CDATI= must be able to be switched to high impedance while the pin specified with option /I2CDATO= is an active output. See chapter 1.8 "Initialization file JTAG386.INI".

- **Specify only one of that options:**
  Some options are exclusive (i.e. /8BIT and /16BIT). Don't mix them.

- **Sum of instruction register bits to low. Should be at least 4 bits for a Intel 386EX**
  The sum of all instruction register bits in the JTAG chain does not fit to the Intel 386EX. Check the target connection. Check the target CPU type. Check the settings for /IROFFS= and /CPUPOS= , if there are several parts in the JTAG chain.

- **Target no longer connected**
  There is a cyclic check of the JTAG chain. Check target power. Check target connection.

- **There are unknown parts in the JTAG chain. Please use the option /IROFFS= to specify the instr. reg. offset of the CPU.**
  If there are unknown parts in the JTAG chain, the program isn't able to determine the logical position of the CPU's instruction register.

- **There is no Intel 386EX in the JTAG chain**
  No Intel 386EX was found in the JTAG chain. Check the target power. Try with option /DRIVER=4 again.

- **Value of option /FILE-OFFSET out of range**
  The value of the option /FILE-OFFSET= points behind end of file.

- **wrong driver #**
  The value specified with the option /DRIVER= is out of range.

- **wrong Identifier (xxxx)**
  No valid identifier found. Check the specified chip select signal and the bus width. Try with the option /DEVICE= .

- **Wrong length of boundary scan register. Should be 202 for a Intel 386EX.**
  The length of the boundary scan register of the selected part (if there are more than one in the chain) does not fit to the Intel 386EX. Check the target connection. Check the target CPU type. Check the settings for /IROFFS= and /CPUPOS= , if there are several parts in the JTAG chain.

1.8. Initialization file JTAG386.INI

This file is used to define the default direction and level of all CPU signals. This file **must be carefully adapted** to your design with the Intel 386EX. The Target-Entry is used to identify your design which is displayed with most commands.

When the program JTAG386.EXE is started it scans the current directory for an existing initialization file named JTAG386.INI. If no entry is found the default values are used. You may also specify the initialization file with the option /INI= . If the specified file isn't found, the program aborts with an error message.

The CPU pins can also be used with the functions /BLINK (chapter 2.12), /PIN? (chapter 2.13) and /SAMPLE (chapter 2.14) to test the signals on your design.

The sample file below represents the values which are used for default initialization when no initialization file could be found in the current directory and no initialization file is specified with the option /INI=.

Changes to the structure of the file could result in errors. Remarks can be added by using //.

Sample File JTAG386.INI:

// Description file for Intel 386EX
Target: Module 386EX, FS FORTH-SYSTEME GmbH
// All chip select signals are set to output and inactive.
// All signals should be defined. Undefined signals are set to their defaults.
// Pin names are defined in upper case.
// Low active signals are signed with a trailing #.

```
M/IO#          Out,Hi
D/C#           Out,Hi
W/R#           Out,Lo
READY#         Out,Hi
BS8#           Inp
RD#            Out,Hi
WR#            Out,Hi
BLE#           Out,Hi
BHE#           Out,Hi
ADS#           Out,Hi
NA#            Inp
```

// The following pins must be set to output/active for flash programming
// and other memory accesses.
```
A1             Out,Hi
A2             Out,Hi
A3             Out,Hi
A4             Out,Hi
A5             Out,Hi
A6             Out,Hi
A7             Out,Hi
A8             Out,Hi
A9             Out,Hi
A10            Out,Hi
A11            Out,Hi
A12            Out,Hi
A13            Out,Hi
A14            Out,Hi
A15            Out,Hi
A16            Out,Hi    // = CAS0
A17            Out,Hi    // = CAS1
A18            Out,Hi    // = CAS2
A19            Out,Hi
A20            Out,Hi
```

```
A21             Out,Hi
A22             Out,Hi
A23             Out,Hi
A24             Out,Hi
A25             Out,Hi

SMI#            Inp
P3.0            Inp         // = TMROUT0
P3.1            Inp         // = TMROUT1
SRXCLK          Inp         // = DTR1#
SSIORX          Inp         // = RI1#
SSIOTX          Inp         // = RTS1#
P3.2            Inp         // = INT0
P3.3            Inp         // = INT1
P3.4            Inp         // = INT2
P3.5            Inp         // = INT3
P3.6            Inp         // = PWRDOWN
P3.7            Inp         // = COMCLK
PEREQ           Inp         // = TMRCLK2
NMI             Inp
TMROUT2         Inp         // = ERROR#
TMRGATE2        Inp         // = BUSY#
INT4            Inp         // = TMRCLK0
INT5            Inp         // = TMRGATE0
INT6            Inp         // = TMRCLK1
INT7            Inp         // = TMRGATE1
STXCLK          Inp         // = DSR1#
FLT#            Inp

P1.0            Inp         // = DCD0#
P1.1            Inp         // = RTS0#
P1.2            Inp         // = DTR0#
P1.3            Inp         // = DSR0#
P1.4            Inp         // = RI0#
P1.5            Inp         // = LOCK#
P1.6            Inp         // = HOLD
RESET           Inp
P1.7            Inp         // = HLDA
DACK1#          Inp         // = TxD1
EOP#            Inp         // = CTS1#
WDTOUT          Inp
DRQ0            Inp         // = DCD1#
DRQ1            Inp         // = RxD1
```

```
SMIACT#        Inp

P2.0           Out,Hi   // = CS0#
P2.1           Out,Hi   // = CS1#
P2.2           Out,Hi   // = CS2#
P2.3           Out,Hi   // = CS3#
P2.4           Out,Hi   // = CS4#
DACK0#         Out,Hi   // = CS5#
P2.5           Inp      // = RxD0
P2.6           Inp      // = TxD0
P2.7           Inp      // = CTS0#
UCS#           Out,Hi
CS6#           Out,Hi   // = REFRESH#
LBA#           Out,Lo
```

// The following pins are switched between input/inactive and output/active
// during flash programming and other memory accesses.

```
D0             Inp
D1             Inp
D2             Inp
D3             Inp
D4             Inp
D5             Inp
D6             Inp
D7             Inp
D8             Inp
D9             Inp
D10            Inp
D11            Inp
D12            Inp
D13            Inp
D14            Inp
D15            Inp
```

1.9. Supported flash devices

Type JTAG386 /LIST [optionlist]

to get a online list of all flash types which could be used with the /DEVICE= option.

See separate file JTAG_V4xx_FLASHES.pdf to get a complete list of supported flash types.

2. JTAG386 Parameter Description

When you start JTAG386.EXE without any parameters the following help screen
with all possible functions and options is displayed:

```
JTAG386 --- JTAG utility for Intel 386EX
Copyright © FS FORTH-SYSTEME GmbH, Breisach
Version 4.xx of mm/dd/yyyy


Programming of Flash-EPROMs and hardware tests on targets with the
Intel 386EX.


The JTAG-Booster is needed to connect the parallel port of the PC
to the JTAG port of the Intel 386EX.


Usage: JTAG386 /function [filename] [/option_1] ... [/option_n]
Supported functions:
        /P          : Program a Flash Device
        /R          : Read a Flash Device to file
        /V          : Verify a Flash Device with file
        /D          : Download to target memory
        /U          : Upload from target to host
        /SRAM       : Test target memory (SRAM)
        /DUMP       : Make a target dump
        /PI2C       : Program an I2C Device with file
        /RI2C       : Read an I2C Device to file
        /VI2C       : Verify an I2C Device with file
        /DUMPI2C    : Make a dump of an I2C Device
        /BLINK      : Toggle a CPU pin
        /PIN?       : Test a CPU pin
        /SAMPLE     : Test a CPU pin while the CPU is running
        /SNAP       : Test all CPU pins while CPU is running
        /LIST       : Print a list of supported Flash devices
```

Supported Options:

/UCS	/CS0	/CS1	/CS2	/CS3
/CS4	/CS5	/CS6	/NOCS	/NOWRSETUP
/TOP	/BYTE-MODE	/BM	/PAUSE	/P
/NODUMP	/NOERASE	/LATTICE	/ERASEALL	/LPT1
/LPT2	/LPT3	/LPT-BASE=	/16BIT	/8BIT
/NOMAN	/LENGTH=	L=	/FILE-OFFSET=	/FO=
/OFFSET=	/O=	/DELAY=	/DEVICE-BASE=	/DB=
/DRIVER=	/DATA-MASK=	/DM=	/IROFFS=	/CPUPOS=
/DEVICE=	/PIN=	/I2CCLK=	/I2CDAT=	/I2CDATI=
/I2CDATO=	/I2CBIG	/WATCH=	/OUT=	/INI=
/REP				

The following options are valid for most functions:

/DRIVER=x   with x = 1,2,3,4
A driver for the interface to the JTAG-BOOSTER on the parallel port may be specified. /DRIVER=1 selects the fastest available driver, /DRIVER=4 selects the slowest one. Use a slower driver if there are problems with JTAG-BOOSTER.
Default: /DRIVER=3

/INI=file
An initialization file may be specified. By default the current directory is searched for the file JTAG386.INI. If this file is not found and no initialization file is specified in the command line, default initialization values are used (see also chapter 1.8 "Initialization file JTAG386.INI").
Default: /INI=JTAG386.INI

/LATTICE
For demonstration purposes this software works with the Lattice ispLSI-Adapter, too. With the option /LATTICE you can simulate the speed achievable with the simple ispLSI-Adapter.

/LPT1 /LPT2 /LPT3
A printer port may be specified where the JTAG-Booster resides.
Default: /LPT1

/LPT-BASE
The physical I/O-Address of printer port may be specified instead of the logical printer name. Useful option, if you work with WinNT or Win2000, because the standard printer port is mapped as LPT2 here. Use the option /LPT-BASE=378 to get a command line which works independent of the operation system.

/OUT=file_or_device
All screen outputs are redirected to the specified file or device. Note that you can't redirect to the same parallel port where the JTAG-Booster resides.
Default: /OUT=CON

/PAUSE
With the option /PAUSE you can force the program to stop after each screen. Please do not use this option if you redirect the output to a file.
Abbreviation: /P

/WATCH=
With the option /WATCH= a pin can be specified, which is toggled twice per second, while the program is active. This pin may be the trigger of a watchdog. This pin must be specified as output in the initialization file.

/IROFFS=
Specifies the position of the Intel 386EX instruction register within the JTAG chain. In most cases this option is not needed.
Default: /IROFFS=0

/CPUPOS=
Specifies the position of the Intel 386EX within the JTAG chain.
Default: /CPUPOS=0

2.1.   Program a Flash Device

Usage: JTAG386 /P filename [optionlist]

The specified file is programmed into the flash memory. The flash status is polled after programming of each cell (cell=8 or 16 bit, depending on current data bus width). In case of a programming error, the contents of the flash memory is written to a file with the extension DMP.

If you want a complete verify after programming, please use an additional command line with the verify function. See chapter 2.3 "Verify a Flash Device with file". In most cases this additional verify step is not needed.

The type of the flash device is normally detected by the software. When autodetection fails you should use the /DEVICE= option together with /8BIT or /16BIT to set the right flash device and configuration. The known devices are shown in chapter 1.9 "Supported flash devices".

Options:

/DEVICE=devicename
The flash device is detected automatically by switching to autoselect mode. In case of trouble you should select the flash device by using this parameter to avoid autodetection. Combine this option with one of the following options which specify the data bus width and the option /BYTE-MODE if applicable.

/8BIT /16BIT
Specifies the data bus width to the target flash device. You can speed up autodetection, if you specify the correct data bus size. You need this option together with the option /DEVICE= to explicit specify a specific flash configuration.

/BYTE-MODE
If there is a flash device connected to the CPU which does have a byte mode pin (8 bit and 16/32 bit bus mode), you can force it to be used as 8 bit mode with the option /BYTE-MODE. In most cases this option will not be needed.