Overview

The XJAPI gives you low-level access to the JTAG chain via either the XJLink or the PXI hardware, allowing easy integration with a wide range of other systems such as development/evaluation boards, test systems...

The USB to JTAG hardware and software interface is composed of a high-speed USB to JTAG hardware module (XJLink2 or XJLink) and a simple-to-use DLL Application Program Interface (XJAPI) designed to allow you to access and control the JTAG chain directly.

The PXI to JTAG hardware and software interface is composed of a high speed PXI to JTAG hardware module (PXI XJLink2) and the same simple-to-use DLL API (XJAPI).

XJAPI functions

Initialising & Terminating

XJAPI_HardwareSetup
Function to set up the hardware and the pin mapping, the desired frequency, and whether power should be applied to the board.

XJAPI_HardwareRelease
The function to release the hardware should be called before exiting.

XJAPI_SetPinMap
Function to set the pin map. Allows you to assign any JTAG function to any of the 16 configurable pins.

Low Level JTAG Access

XJAPI_SetFrequency
Function to set TCK frequency - between 100kHz and 60MHz.

XJAPI_TmsReset
Function to apply a TMS reset.

XJAPI_GotoState
Function to go to a specific JTAG TAP state.

XJAPI_SetEndState
Function to set the final TAP state that the system goes to after a DR or IR scan operation.

XJAPI_ClockChain
Function to clock the JTAG chain a specific number of times.

High Level Scan functions

XJAPI_Scan
Function to execute a JTAG DR/IR scan cycle. By default, it will leave the system in the JTAG_IDLE state after the scan. To specify a different end state, use XJAPI_SetEndState.

XJAPI_ScanMultiple
Function to implement multiple scans. This function is used to scan multiple (nScans) chains of mixed type (DR and IR scans) and of mixed length.

Miscellaneous functions

XJAPI_AutoSkew
Function to automatically compensate for clock skew for the current TCK frequency.

XJAPI_GetLastError
XJAPI_GetVersion
XJAPI_ReadPins
XJAPI_SetPins
XJAPI_SetTrst
XJAPI_Shutdown
XJAPI_Startup
XJAPI_Trst

USB to JTAG interface

Key Benefits

- Faster communication / download USB (480 Mbps), JTAG (60 Mbps peak)
- USB to JTAG: Small, lightweight, portable hardware design — ideal for lab and field work
- PXI to JTAG form factor also available: fully software compatible with the USB to JTAG version (3U/32 bit PXI/PCI bus interface)
- Self-contained licence allowing you to use the XJTAG system on multiple machines
- Can be used with any pinout, ARM, Xilinx, Altera, etc.
- Easily customisable

Features

- JTAG/IEEE 1149.x compliant
- High speed USB 2.0 interface, backwards compatible with USB 1.0 & 1.1
- USB bus-powered (no external PSU)
- Can supply power to the target board (3.3V, <100 mA)
- TCK clock frequencies up to 60 MHz
- Adjustable JTAG signal termination
- Automatic signal skew control
- Software configurable pin mapping
- JTAG signals are +5V tolerant
- Spare signals on JTAG connector can be used to control other items e.g. hold a board in reset / turn on a PSU
- Provided with all needed files, libraries and an application example
- Designed to be used in C or C++ applications
- XJDemo board available
- Runs on Windows 8 / 7 / XP / Vista

Overview of the XJAPI

The XJAPI gives you low-level access to the JTAG chain via either the XJLink or the PXI hardware, allowing easy integration with a wide range of other systems such as development/evaluation boards, test systems...

The USB to JTAG hardware and software interface is composed of a high-speed USB to JTAG hardware module (XJLink2 or XJLink) and a simple-to-use DLL Application Program Interface (XJAPI) designed to allow you to access and control the JTAG chain directly.

The PXI to JTAG hardware and software interface is composed of a high speed PXI to JTAG hardware module (PXI XJLink2) and the same simple-to-use DLL API (XJAPI).

XJAPI functions

Initialising & Terminating

XJAPI_HardwareSetup
Function to set up the hardware and the pin mapping, the desired frequency, and whether power should be applied to the board.

XJAPI_HardwareRelease
The function to release the hardware should be called before exiting.

XJAPI_SetPinMap
Function to set the pin map. Allows you to assign any JTAG function to any of the 16 configurable pins.

Low Level JTAG Access

XJAPI_SetFrequency
Function to set TCK frequency - between 100kHz and 60MHz.

XJAPI_TmsReset
Function to apply a TMS reset.

XJAPI_GotoState
Function to go to a specific JTAG TAP state.

XJAPI_SetEndState
Function to set the final TAP state that the system goes to after a DR or IR scan operation.

XJAPI_ClockChain
Function to clock the JTAG chain a specific number of times.

High Level Scan functions

XJAPI_Scan
Function to execute a JTAG DR/IR scan cycle. By default, it will leave the system in the JTAG_IDLE state after the scan. To specify a different end state, use XJAPI_SetEndState.

XJAPI_ScanMultiple
Function to implement multiple scans. This function is used to scan multiple (nScans) chains of mixed type (DR and IR scans) and of mixed length.

Miscellaneous functions

XJAPI_AutoSkew
Function to automatically compensate for clock skew for the current TCK frequency.

XJAPI_GetLastError
XJAPI_GetVersion
XJAPI_ReadPins
XJAPI_SetPins
XJAPI_SetTrst
XJAPI_Shutdown
XJAPI_Startup
XJAPI_Trst

USB to JTAG interface

Key Benefits

- Faster communication / download USB (480 Mbps), JTAG (60 Mbps peak)
- USB to JTAG: Small, lightweight, portable hardware design — ideal for lab and field work
- PXI to JTAG form factor also available: fully software compatible with the USB to JTAG version (3U/32 bit PXI/PCI bus interface)
- Self-contained licence allowing you to use the XJTAG system on multiple machines
- Can be used with any pinout, ARM, Xilinx, Altera, etc.
- Easily customisable

Features

- JTAG/IEEE 1149.x compliant
- High speed USB 2.0 interface, backwards compatible with USB 1.0 & 1.1
- USB bus-powered (no external PSU)
- Can supply power to the target board (3.3V, <100 mA)
- TCK clock frequencies up to 60 MHz
- Adjustable JTAG signal termination
- Automatic signal skew control
- Software configurable pin mapping
- JTAG signals are +5V tolerant
- Spare signals on JTAG connector can be used to control other items e.g. hold a board in reset / turn on a PSU
- Provided with all needed files, libraries and an application example
- Designed to be used in C or C++ applications
- XJDemo board available
- Runs on Windows 8 / 7 / XP / Vista
**XJAPI Data types**

**XJTAG_STATE**
This enumeration defines the possible states for the JTAG TAP controller as defined in the IEEE 1149.1 specification.

**XJAPI_ERROR**
This enumeration contains error codes that can be returned from the various API functions.

**XJAPI_PIN_DEF**
This structure is used to define an individual pin in a user-defined pinmap.

**XJAPI_PIN_DRIVE**
Enumeration of the two different pin output impedance values.

**XJAPI_PIN_TYPE**
Enumaration of the 8 different pin types available when creating a user-defined pinmap.

**XJAPI_PIMAP**
Enumeration of the different standard or user-defined pinmaps. Used as an argument to the XJAPI_HardwareSetup and XJAPI_SetPinMap functions.

**XJAPI_SCAN_TYPE**
An enumeration of the different scan types available. Used as an argument to XJAPI_Scan and XJAPI_ScanMultiple functions.

**XJAPI_USER_MAP**
A datatype for describing a user-defined pinmap.

---

**Supplied files**

- **xjapi.h**
  - Header file describing the XJAPI functions and datatypes.

- **jtag.h**
  - Header file with the states defined in IEEE 1149.1 JTAG specification.

- **xjapi.dll, hwif.dll, common.dll**
  - The DLLs required to use XJAPI.

- **xjapi.lib**
  - The XJAPI import library in COFF format (used by Microsoft Visual Studio).

- **xjapi_OMF.lib**
  - The XJAPI import library in OMF format (used by Borland’s C/C++ compilers).

- **xjapi_example.c**
  - C code example demonstrating how to use most of XJAPI’s functionality (works on XJDemo board for validation).

---

**Broadcom Videocore® platform**

Application example using the USB to JTAG HW & SW interface (courtesy of Broadcom).