Jam Programming & Test Language Overview

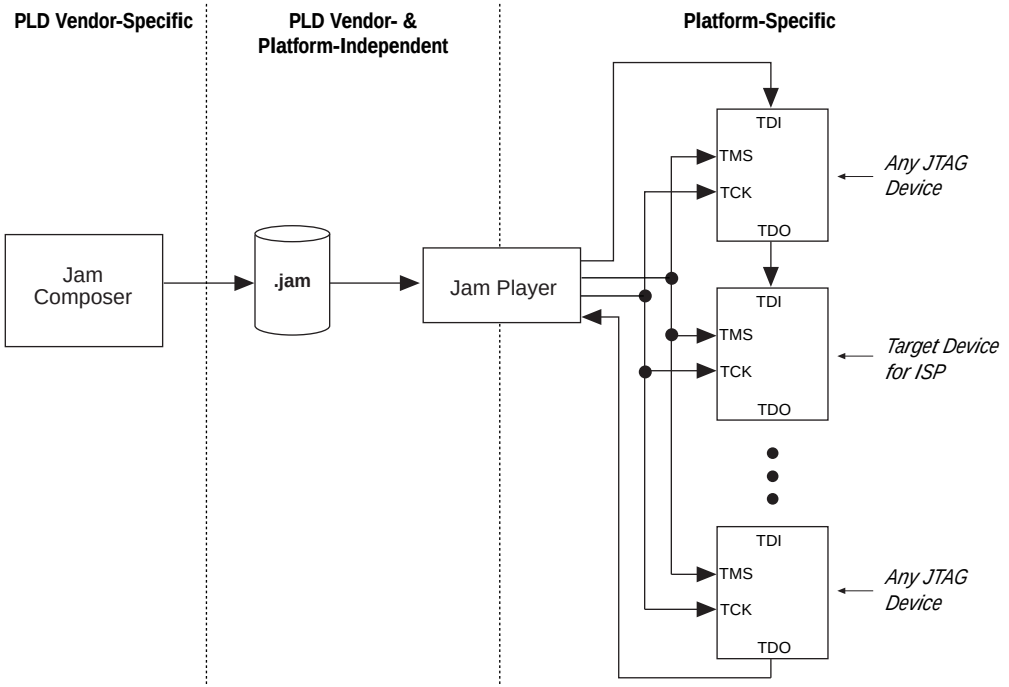
February 1998, ver. 1 Product Information Bulletin 27

Introduction

The Jam™ programming and test language, a new standard file format for in-system programmability (ISP) purposes, is designed to support programming or configuration of programmable devices and testing of electronic systems, using the IEEE 1149.1 Joint Test Action Group (JTAG) interface. The Jam language is a freely licensable open standard.

The Jam solution consists of two software components: the Jam Composer and the Jam Player. The Jam Composer is a software program, generally written by a programmable logic device (PLD) vendor, that writes a Jam File (**.jam**) containing the user data and programming algorithm required to program or configure a device. The Jam Player is software that reads the Jam File and applies vectors for programming and testing devices in a JTAG chain. Embedded system developers can also use a Jam Player to program devices in their systems. Most of the source code required for the Jam Player is contained in the Jam Device Programming & Test Language Developer's Kit (see "Jam Language Developer's Kit" on page 15 for more information). Moreover, the only software routines required to complete the Jam Player are those needed to access the JTAG chain. Figure 1 shows a basic Jam flow.

Figure 1. Basic Jam Flow



For more information on the Jam Composer and Jam Player, see the *Jam Programming & Test Language Specification* or *AN 88 (Using the Jam Language for ISP via an Embedded Processor)* in this handbook, or go to the Jam web site at **http://www.jamisp.com**.

Benefits of the Jam Language

The Jam language provides vendor- and platform-independence, small file sizes, fast programming times, and support for existing and future PLDs.

Vendor- & Platform-Independence

The Jam language is vendor- and platform-independent. Therefore, new software is not required to support new PLDs, and a single programming file is generated for different programming platforms. With vendor- and platform-independence, any Jam Player can read any Jam File and a Jam File is compatible with any Jam Player running on any platform (i.e., in-circuit testers and embedded processors.)

Small File Size

Although Jam Files are written in ASCII format and contain both programming data and algorithms, they are very compact files. Jam Files are so compact that they can fit on a FLASH memory card (e.g., for in-system programming via an embedded processor). A small file size is also important in test environments, where larger files take longer to process.

Jam File sizes are reduced by describing repetitive sequences of JTAG waveforms algorithmically, and by compressing programming data. The Jam language supports looping, branching, and subroutine instructions such as CALL/RETURN and FOR/NEXT. It also supports run-length encoding and advanced data compression.

Fast Programming Times

The Jam language supports branching instructions, which allows the Jam Player to use shorter programming pulses. Because of different manufacturing tolerances, the programming pulse used to program PLDs varies over time. The Jam language speeds up programming times by reading the programming pulse width from the device prior to programming. Without this capability, the worst-case programming pulse width must be used for all devices; this pulse is often an order-of-magnitude longer than the typical programming pulse.

Existing & Future PLD Support

The Jam language is a completely generic programming language and is not customized for any specific PLD architecture. Therefore, any programming algorithm can be described in a Jam File. This flexibility ensures that all existing and future PLD devices can be supported by the Jam language.

Jam Language Developer's Kit

The Jam Device Programming & Test Language Developer's Kit allows programmers to develop software and hardware that support the Jam language. The kit contains the documentation and software necessary to develop the Jam Composer and Jam Player.

For more information on obtaining the Jam Device Programming & Test Language Developer's Kit, go to the Jam web site at **http://www.jamisp.com**. For more information on using the Jam language with Altera devices, go to the Altera web site at **http://www.altera.com**.