MachXO JTAG Programming and Configuration User Guide

Technical Note

FPGA-TN-02167-1.5

November 2022

Disclaimers

Contents

Figures

Tables

1. Introduction

The MachXO™ is a reconfigurable programmable logic device. The MachXO uses SRAM memory cells to allow configuring the device to any required functionality. The MachXO also provides non-volatile Flash memory cells to store the configuration data. The data retrieved from the Flash memory rapidly loads the SRAM memory at powerup or at the request of the user. This combination of memory types provides several unique programming and operational capabilities not found in antifuse or SRAM-only based FPGA devices. Some of the capabilities provided by the MachXO:

- "Instant-on" SRAM loading at power-up
- Bitstream security. Since there is no external PROM there is no access to the bitstream used to program the MachXO device.
- "Instant" reconfiguration to a known good state from the Flash memory, i.e. recovery from "soft upset events"
- Reprogramming of the Flash while in normal SRAM operation

This technical note describes how to take advantage of these unique capabilities.

2. Programming Overview

The MachXO contains two types of memory, SRAM and Flash (refer to Figure 2.1.). SRAM contains the active configuration, essentially the "fuses" that define the circuit connections; Flash provides an internal storage space for the configuration data.

The SRAM can be configured in two ways; by using IEEE 1532 mode via the IEEE 1149.1 compliant ispJTAG™ port or from data stored in the on-chip Flash memory. While writing SRAM via the ispJTAG port the state of the FPGA's I/Os is determined by the BSCAN registers. The state of each BSCAN (boundary scan) register can be determined by the user; available options are high, low, tristate (default), or current value (also called leave alone).

The SRAM may be read without disturbing the operation of the device. This is called transparent, or background readback. Care must be exercised when reading EBR or distributed RAM, as it is possible to cause conflicts with accesses from the user design, causing possible data corruption. It is recommended that read/write enables to the EBR or distributed RAM be turned off any time these resources are being read via the JTAG port.

The on-chip Flash can also be programmed using IEEE 1532 mode via the IEEE 1149.1 compliant ispJTAG port. If the SRAM portion of the device is blank then the Flash will be programmed using direct mode. In direct mode the state of the device's I/Os is determined by the BSCAN registers. The state of each BSCAN register can be determined by the user; available options are high, low, tristate (default), or current value. If the SRAM portion of the device is not blank then the Flash will be programmed in background (transparent) mode.

Both SRAM and Flash memory in the MachXO have multiple security fuses to prevent unauthorized readback of the configuration data. Once set, the only way to clear these security bits is to erase the memory space. A secured device will read out all zeros.

Note that very early production releases of the MachXO1200 and MachXO2280 marked with a "4W" in the part number field did not support writes to, or reads from, the SRAM fabric using JTAG. These parts do, however, support all other programming modes.

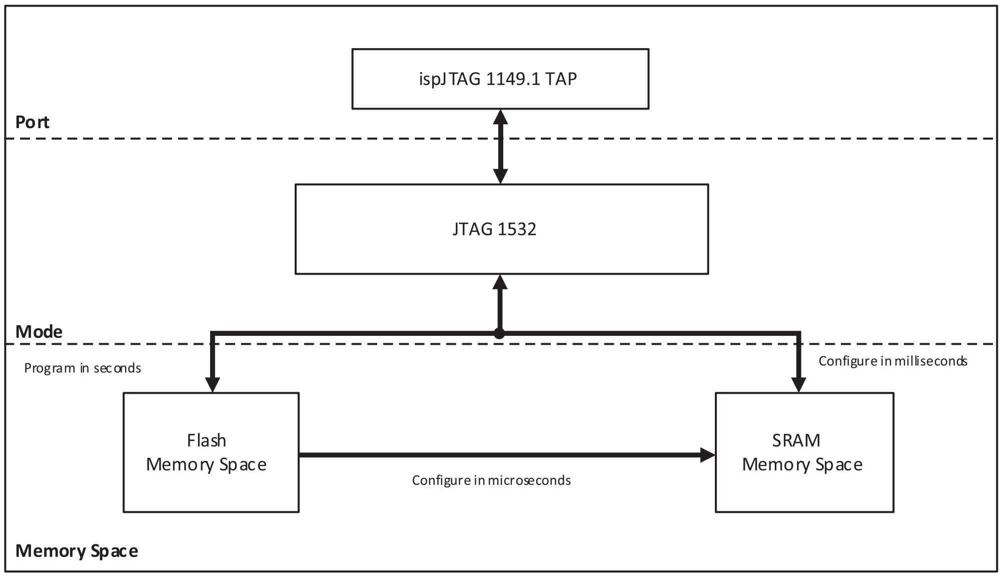


Figure 2.1. Programming Block Diagram

3. ispJTAG

The ispJTAG pins are standard IEEE 1149.1 TAP (Test Access Port) pins. The ispJTAG pins are dedicated pins and are always accessible when the MachXO device is powered up.

Table 3.1. ispJTAG Pins Definition

Pin Name	I/O Type	Pin Type
TDO	Output, weak pull-up	JTAG
TDI	Input, weak pull-up	JTAG
TMS	Input, weak pull-up	JTAG
TCK	Input	JTAG

Note: Weak pull-ups consist of a current source of 30 µA to 150 µA. The pull-ups for TDO, TDI, and TMS track the associated VCCIO. A pull-down of 4.7K is recommended on TCK.

3.1. TDO

The Test Data Output pin is used to shift serial test instructions and data out of the MachXO. TDO is clocked out on the falling edge of TCK. When TDO is not being driven by the internal circuitry, the pin will be in a high impedance state (tristate). An internal pull-up resistor on the TDO pin is provided. The internal resistor is pulled up to the associated V_{CCIO}.

3.2. TDI

The Test Data Input pin is used to shift serial test instructions and data into the MachXO. TDI is clocked into the device on the rising edge of TCK. An internal pull-up resistor on the TDI pin is provided. The internal resistor is pulled up to the associated V_{CCIO}.

3.3. TMS

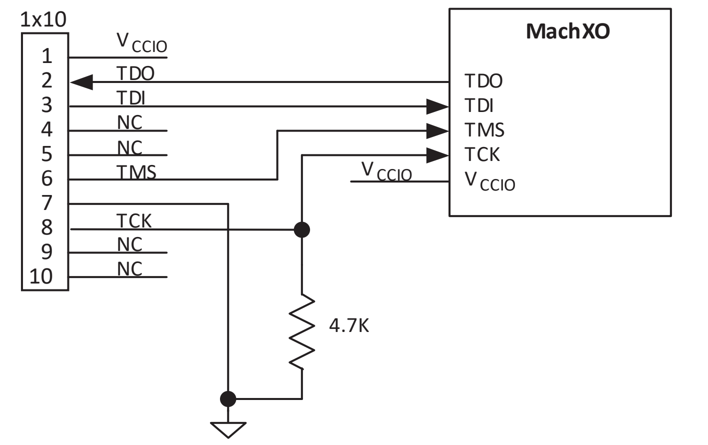
The Test Mode Select pin controls test operations on the TAP controller. On the rising edge of TCK, depending on the state of TMS, a transition will be made in the TAP controller state machine. An internal pull-up resistor on the TMS pin is provided. The internal resistor is pulled up to the associated V_{CCIO}.

3.4. TCK

The test clock pin provides the clock to run the TAP controller, which loads and unloads the data and instruction registers. TCK can be stopped in either the high or low state and can be clocked at speeds up to the frequency indicated in the MachXO Family Data Sheet. The TCK pin does not have a pull-up. A pull-down on the PCB of 4.7 K is recommended to avoid inadvertent clocking of the TAP controller as VCC ramps up.

3.5. V_{CC} Supply for JTAG

V_{CC} for the internal JTAG logic is supplied by the associated bank's V_{CCIO}. To determine which bank contains the JTAG pins please refer to the MachXO Family Data Sheet. Valid voltage levels are 3.3 V, 2.5 V, 1.8 V, 1.5 V, and 1.2 V, but check that the desired voltage is compatible with the ispDOWNLOAD® Cable connected to the download header.



Note: Place a decoupling capacitor close to the connector's V_{CCIO} pin. Any standard ceramic capacitor value may be used, for example 0.1 µF, 0.01 µF, etc.

Figure 3.1. Download Header to MachXO Wiring

3.6. Download Cable Pinout

Standard pinouts for the 1x10, 1x8, and 2x5 download headers are shown in the ispDOWNLOAD Cable Data Sheet. All new ispDOWNLOAD Cables have uncommitted "flywire" connections, so they can be attached to any of the header styles. Refer to the ispDOWNLOAD Cable Data Sheet for additional details.

3.7. BSDL Files

BSDL files for this device can be found on the Lattice Semiconductor web site. The boundary scan ring covers all of the I/O pins.

4. Device Wake Up

When configuration is complete (the SRAM has been loaded), the device will wake up in a predictable fashion. The wake up sequence is driven by, and is synchronous to, an internal clock.

5. Software Options

5.1. Preference Options

Preference options are set by opening the Preference Editor within the ispLEVER® or Lattice Diamond™ design software and selecting the global options tab.

Table 5.1. Preference Options for the MachXO

Preference Name	Default Setting [List of All Settings]
CONFIG_SECURE	OFF [off, on]
Usercode Format	Bin[Bin, ASCII, Hex]
Usercode	00000000000000000000000000000000

5.2. Security

When CONFIG_SECURE is set to ON the on-chip security fuses will be set and no readback of the general contents (SRAM or Flash) will be supported through the ispJTAG port. The ispJTAG DeviceID area (including the Usercode) and the device Status Register are readable and not considered securable. Default is OFF.

5.3. Usercode

Usercode is a user defined 32-bit register, sometimes called the UES (User Electronic Signature). The Usercode can be thought of as a user “notepad”. The Usercode is supported as part of the IEEE 1149.1 definition. Lattice incorporated the Usercode to store such design and manufacturing data as the manufacturer's ID, programming date, programmer make, pattern code, checksum, PCB location, revision number, and product flow. The intent is to assist users with the complex chore of record maintenance and product flow control. In practice, the Usercode can be used for any of a number of ID functions. The Usercode can be easily edited using the Usercode/UES Editor in ispVM® by clicking on **ispTools -> ispVM Editors**.

Note that the Usercode is included in the file transmission checksum, but not the fuse checksum.

5.4. Configuring SRAM or Programming Flash

The final step in the design process is to load the JEDEC file created by ispLEVER or Diamond into the MachXO. The JEDEC file can be used to configure the SRAM or program the Flash. Simply connect a Lattice download cable to the ispJTAG connector that is wired to your MachXO and apply power to the FPGA. Then start ispVM and follow these steps:

1. Click on **File -> New**.
2. Click on **Edit -> Add New Device**.
3. **Select** the MachXO device, **Browse** to the JEDEC file created in ispLEVER or Diamond, and then select either SRAM or Flash via the **Device Access Options** drop down box.
4. Click **OK**. You are now ready to program the MachXO by clicking on the green **GO** button on the toolbar.
5. For other options, such as Read and Save, Verify ID, etc., return to **Device Information** by double clicking on the MachXO in **Device List** and use the **Operation** drop-down box.

If you need to create an SVF (Serial Vector File), a file to allow programming from your ATE (Automated Test Equipment), or a file to support VME (ispVM Embedded), simply perform steps 1, 2, and 3 above but click on SVF, ATE, or VME instead of GO.

More information on ispVM and ispVM Embedded can be found by starting ispVM and clicking on Help. Here you will find several tutorials as well as the help facility.

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Revision History

Revision 1.5, November 2022

Section	Change Summary
All	• Changed document number from TN1086 to FPGA-TN-02167. • Changed document title to *MachXO JTAG Programming and Configuration User Guide*. • Updated document template.
Disclaimers	Added this section.

Revision 1.4, June 2010

Section	Change Summary
All	Updated for Lattice Diamond design software support.

Revision 1.3, February 2007

Section	Change Summary
ispJTAG	Updated Download Cable Pinout section.

www.latticesemi.com