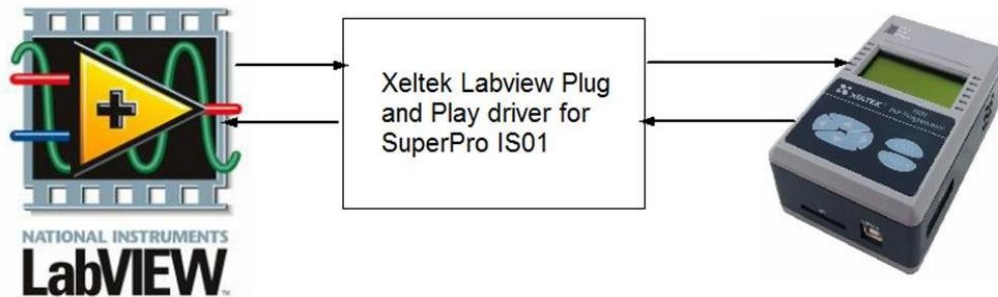


In-System Programming Made Easy with LabVIEW

Be the first to test out our LabVIEW driver for **FREE!**

Available on a first-come-basis

Xeltek in-system (ISP) programmer, SuperPro IS01, can now program serial devices under LabVIEW! Integrating SuperPro IS01 onto an existing LabVIEW environment can now be a simple task. Previously, users had to purchase DLL functions and create a set of custom VIs for integration, taking days and resulting in expensive engineering time. The LabVIEW driver package is ready to be used in the test program with very little need of coding and customization.



- SuperPro IS01 is an industrial and production grade Universal Serial programmer
- Capable of being integrated into an In-Circuit Test environment for automated production operation
- Multiple units may be installed and operated in stand-alone mode for programming of various devices concurrently
- 9,000+ devices including serial E(E)PROMs, MCUs and other types are supported as of November 2013
- SuperPro IS01 is designed to operate with a PC or in stand-alone mode for production setup
- Built-in ARM9 and Linux O/S provide fast programming of most devices.

Xeltek is introducing a set of LabVIEW VIs for all programming functions necessary to run SuperPro IS01 serial programmer. Generate project file with SuperPro IS01 software (refer to the SuperProIS01 handbook for more details), then use LabVIEW to operate the ISP programmer.

Standard Functions	Auxiliary Functions	Advanced Functions (Optional)
Auto Program	Initialize	Create Project
Read	Ping	Multi-Devices Control
Verify	Get Status	Customized Serial Number Generation
Blank Check	Get DLL Version	UpdateFile
Erase	Get ISP S/N	UpdateFileB
Secure (Protect)	Get Client IP	UpdateFileC
Save File (Set Read Save)	Get Server IP	
Checksum	Set Client IP	
Load Project	Set Server IP	
Unload Project		
Release Buffer		
Reload Buffer		
Read Buffer		

Table 1: LabVIEW Functions

Standard Functions

Auto	Perform AUTO function sequence which was set at the time of project construction. All devices have a basic batch process operation "Auto", this function is to execute other operation functions of devices in the already edited sequence. Generally, Edit Auto mode is selected for all devices. Example sequence: Program, Read, Verify, Blank Check, Erase, Protect
Program (Run)	Programming device is a process to burn the data in the buffer into the storage unit of the chip according to the manufacturer's requirements.
Read	Read data in the chip.
Verify	Compare whether content of written chip is consistent with that in BUFFER.
Blank Check	Verify that the chip has no data written in it.
Erase	Electrically erase the chip content.
Secure (Protect)	Encrypt the chip with security function.
Save File (Set Read Save)	Save project file.
Checksum	Calculate the checksum of data files in the project and compare it with CHECKSUM previously stored and calculated in current project, if there is a difference, CHECKSUM values are displayed, respectively. Both physical and man-made causes are likely to result in contamination or damage of SD card data, which results in scrapping of the burned chip. Therefore, it is suggested to check the CHECKSUM at least once each time before starting the system.
Load Project	Load project file before calling other functions. This is usually the initial step.
Unload Project	Unload project file.
Release Buffer	BUFFER content is displayed.
Reload Buffer	Content is being reloaded into the BUFFER
Read Buffer	Content is read from the BUFFER

Auxiliary Functions

Initialize	Initialization / getting the system ready for programming operations.
Ping	Communicate with ISP programmer before system INITIALIZE and LOAD PROJECT.
Get Status	Obtain ISP status.
Get DLL Version	Obtain DLL version.
Get ISP S/N	Obtain the programmer's serial number.
Get Client IP	Obtain Client IP address.
Get Server IP	Obtain Service IP address.

Set Client IP	Set Client IP address.
Set Server IP	Set Server IP address.
Advanced Functions (Optional)	
Multi-Devices Control	Running multiple programmers at once. Argument “devno” represents the SuperProIS01 number ID. If have two SuperProIS01s are to be controlled, the devno may be 1 and 2. Every device has its own project file, and the devno represents the number of devices and project files. After system init, every SuperProIS01 will display its own ID number (such as 1, 2 ...) on the LCD.
Customized Serial Number Generation	Assigned customized serial number to chip.
UpdateFile	Replace datefile of project
UpdateFileB	Replace date file of project. Similar to UpdateFile, but does not have arguments for calcs and pDevInfo. UpdateFile calculates the data file check sum and this function does not.
UpdateFileC	Replace date file of project. Similar to UpdateFile, but does not have arguments for calcs and pDevInfo. UpdateFile calculates the data file check sum and this function does not.

Table 2: LabVIEW Function Descriptions

For additional information, please contact info@xeltek.com.