

Technical Note

NXP S32R294: eFuse programming

May 2023



A **TASKING** Company

This Technical Note describes how to read or write eFuses on NXP S32R294Microcontroller via Python script.

Tool requirements:
 ✓ winIDEA 9.17.140 or newer
 ✓ BlueBox iC5700, iC5000
 ✓ MPC5xxx/SPC5 Aurora Active Probe

eFuses are OTP (One-Time Programmable) memory and can be used for several purposes, including unique chip identifiers, mask revision numbers, cryptographic keys, boot configurations, various control signals requiring permanent non-volatility. The OCOTP (On-Chip One Time Programmable Controller) is the primary interface to the On-Chip fuses. It starts automatically during the reset process, fetches eFuses contents and loads that information into the OCOTP shadow registers. The OCOTP also provides the ability to program and read eFuses directly, ignoring the shadow cache.

BlueBox debugger can read or program required values to the selected eFuses by running a simple Python script via winIDEA or [isystem.connect API](#).

 Programming results in blowing an eFuse. This is an operation with permanent result which cannot be reversed. Refer to the microcontroller data sheet for more information about what eFuses are available and which values should be programmed.

Python script example

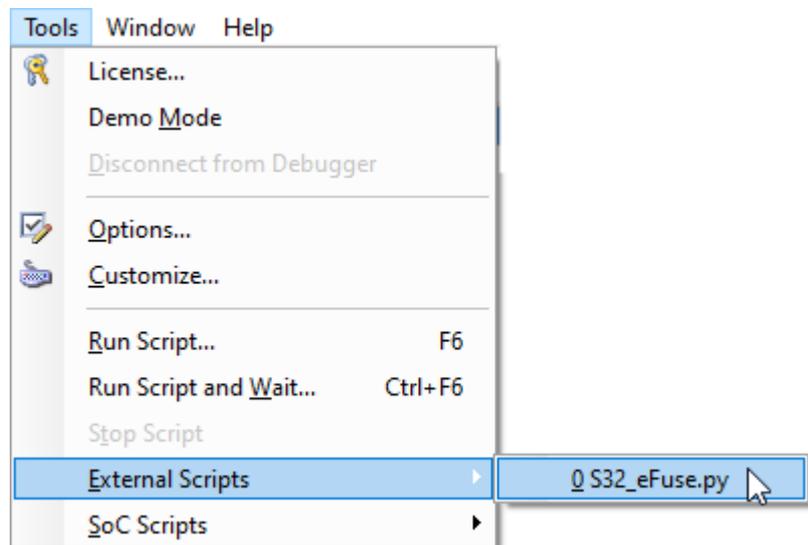
The script should be used as a reference and changed according to the required operation (READ or WRITE). Download and unzip the script example via the link below:



winIDEA Configuration

Follow these steps to run the script in winIDEA:

1. Modify the script according to the required configuration.
2. Save it in the same folder as winIDEA workspace.
3. Start Debug session (Debug status must be STOP).
4. Select your Python script in *Tools / External Scripts*. The script will execute the required operation.



 For more information on how to run Python scripts in winIDEA refer to [winIDEA Help](#). You can use [isystem.connect API](#) for running Python scripts.