**Intro to System-on-Chip Design Course**

**Guide to using update\_bitstream.tcl**

**Issue 1.0**

Contents

[1 Introduction 1](#_Toc89172074)

[2 File structure 1](#_Toc89172075)

[2.1 Script pre-requisites 1](#_Toc89172076)

[3 Steps to generate the new bitstream 1](#_Toc89172077)

# Introduction

This document describes how to use the update\_bitstream.tcl file supplied along with the Education Kit.

The update\_bitstream.tcl file allows the user to bypass resynthesis, reimplementation and bitstream regeneration whenever a new code.hex file is generated. The update\_bitstream.tcl file when run, will repopulate the BRAM with the new content in the code.hex, thus saving time.

**NOTE:** The supplied update\_bitstream.tcl file was tested with Vivado 2019.1.

# File Structure

The following files are included in the update bitstream package and are required to run this flow:

* update\_bitstream.tcl, the top level flow,
* update\_bitstream\_header.tcl, the header file containing the custom procedures.

Calling “source update\_bitstream.tcl” with all the pre-requisites met will produce a file called reflash.bit in the project directory (this is the same directory as update\_bitstream.tcl) that can be used to reflash the FPGA.

## Script pre-requisites

The script requires the following to run successfully:

* A valid data file named ***code.hex*** which must be in the same directory as update\_bitstream.tcl.
  + The ***code.hex*** file is created when you compile your code in keil MDK.
  + You must copy this file over to the Vivado project.
* An up-to-date bitstream file called ***AHBLITE\_SYS.bit***
  + This file is generated after running Implementation in Vivado.

# Steps to Generate a New Bitstream

1. Ensure that all three files (update\_bitstream.tcl, update\_bitstream\_header.tcl and code.hex) are in the same directory as the Xilinx project ‘***.xpr***‘ file.
2. In Vivado, select the Tcl Console tab. See *Fig 1*.
3. In the console window type the following two commands, **press Enter** after each:
   * *cd [get\_property DIRECTORY {current\_project}]*
   * *source update\_bitstream.tcl*
4. If run is successful, a new bitstream ‘***reflash.bit***’ will be created in the same directory as the update bit stream files.
5. The new bitstream now contains the updated code from Keil and can be used to program the FPGA.
6. To program the board with the newly created bitstream file, **see Step 2** in the Program and Debug subsection in the Getting Started Guide. This time, select the newly created bitstream ‘***reflash.bit***’.

Graphical user interface, text, application

Description automatically generated

**2**

**1**

Figure 1: Tcl Console Tab (1) and Console Window (2) highlighted in Red.