



KNOWLEDGE RESOURCES  
Switzerland GmbH

# KRC3701 Template Generator

---

*User Guide*

Knowledge Resources GmbH  
Ackerstrasse 30  
CH – 4057 Basel  
Switzerland

[www.knowres.com](http://www.knowres.com)



---

## Table of Contents

Revision History.....	3
Disclaimer.....	3
Assumptions.....	4
Acronyms .....	4
Reference documents .....	4
Support.....	4
Introduction .....	5
Compatibility .....	5
On Windows .....	5
On Linux .....	5
Script Flow.....	6
Script Resources .....	6
Top Level Module.....	7
Quick Example: Hello World .....	8

## Revision History

Date	Document revision	Changes
December 15 <sup>th</sup> 2014	1.0	Initial document
February 2 <sup>nd</sup> 2015	1.1	First public release, Minor changes
February 8 <sup>th</sup> 2016	1.2	Update for newer Vivado versions Added USB Hub reset and I2C, RGB LED signals Minor changes
April 7 <sup>th</sup> 2016	1.3	Minor changes
May 4 <sup>th</sup> 2016	1.4	Removed module family specific references
October 3 <sup>rd</sup> 2016	1.5	Update for newer Vivado version 2016.2 Top Level description update Branch off specific KRC3701 document
August 14 <sup>th</sup> 2019	1.6	Update for Vivado version superior to 2016.2 Added support for GNU/Linux based operating system Updated the address of Knowledge Resources GmbH
February 14 <sup>th</sup> 2020	1.7	Update Quick Example for Vivado version 2019.1 Correct hyperlinks
February 20 <sup>th</sup> 2020	1.8	Update for Zynq MPSoC Ultrascale+ series KR modules

## Disclaimer

Copyright © Knowledge Resources GmbH. All rights reserved.

All provided data is for information purposes only and not guaranteed for legal purposes.

Information has been carefully checked and is believed to be accurate;

However, no responsibility is assumed for inaccuracies.

Referenced brand and product names are trademarks or registered trademarks of their respective owners.

Specifications are subject to change without notice.

Trademark Acknowledgement:

Brand and product names are trademarks or registered trademarks of their respective owners.

## Assumptions

The reader is familiar with Xilinx FPGA and SoC components and the related terminology in common use.

## Acronyms

FOM:	FPGA on Module
FU:	Future Use
KR:	Knowledge Resources GmbH
MIG:	Memory Interface Generator, a tool of Xilinx to easily implement a DDR3 controller
NA:	Not Applicable
PL:	Programmable Logic
PS:	Processing Subsystem
SoC:	System on Chip

## Reference documents

ZYNQ all programmable SoC, Xilinx, [www.xilinx.com](http://www.xilinx.com)  
DDR3 SDRAM, Micron, [www.micron.com](http://www.micron.com)  
DDR4 SDRAM, Micron, [www.micron.com](http://www.micron.com)  
QSPI Flash memory, N25Q128, Micron, [www.micron.com](http://www.micron.com)

## Support

KR will provide free of charge to qualified customers:

- Schematic and PCB libraries with Module and carrier board design components (Altium)
- 3D STEP models of the module and heat spreader plate
- LINUX BSP and LINUX port (Plug and Boot ready)
- Reference schematics of the evaluation boards (Altium native and PDF)
- Reference designs for on-board PL memory use
- Constraints files (pinning) to accelerate design starts

Further support to aid in customer specific design in's is available at competitive rates, please contact KR for details: + 41 61 545 2080 or mail to [office@knowres.com](mailto:office@knowres.com)

## Introduction

The KRM Template Generator is a collection of simple scripts which facilitate creating new Xilinx Vivado Projects for KRM Modules and Carrier Boards. They require minimal user input and generate a template project which includes the correctly configured processing system, a top-level VHDL file and a complete XDC file for IO-standard and location constraints.

## Compatibility

These scripts have been tested on Zynq 7000 series KR modules with the following versions of Vivado:

- Vivado 2016.2,
- Vivado 2017.2.1,
- Vivado 2018.2.1
- Vivado 2019.1.

The Zynq MPSoC Ultrascale+ series KR modules have been verified with Vivado 2019.1.

They may work with newer and older versions of Vivado if they use a compatible PS IP Block revision as the versions listed above.

The information regarding X1\_MGT0 and X1\_MGT1 only applies to the modules that feature MGTs:

- KRM-3Z7030, KRM-3ZU5CG, KRM-3ZU4CG features X1\_MGT0
- KRM-3Z7035, KRM-3Z7045 feature X1\_MGT0 and X1\_MGT1

### On Windows

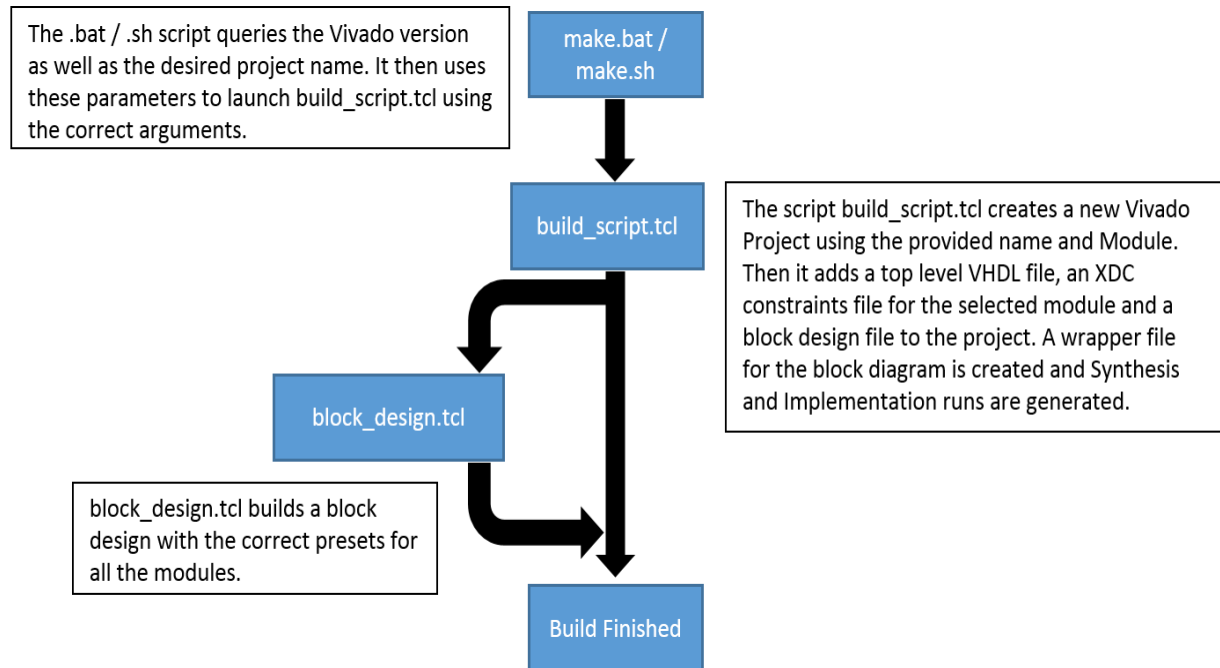
The make.bat script expects the Xilinx Tools to be installed in C:\Xilinx\Vivado. If they are installed in a different location the path needs to be modified in the make.bat script.

### On Linux

The make.sh script expects the Xilinx Tools to be installed in /opt/Xilinx/Vivado. If they are installed in a different location the path needs to be modified in the make.sh script.

## Script Flow

The Template Generator goes through multiple layers of scripts to generate the template projects.



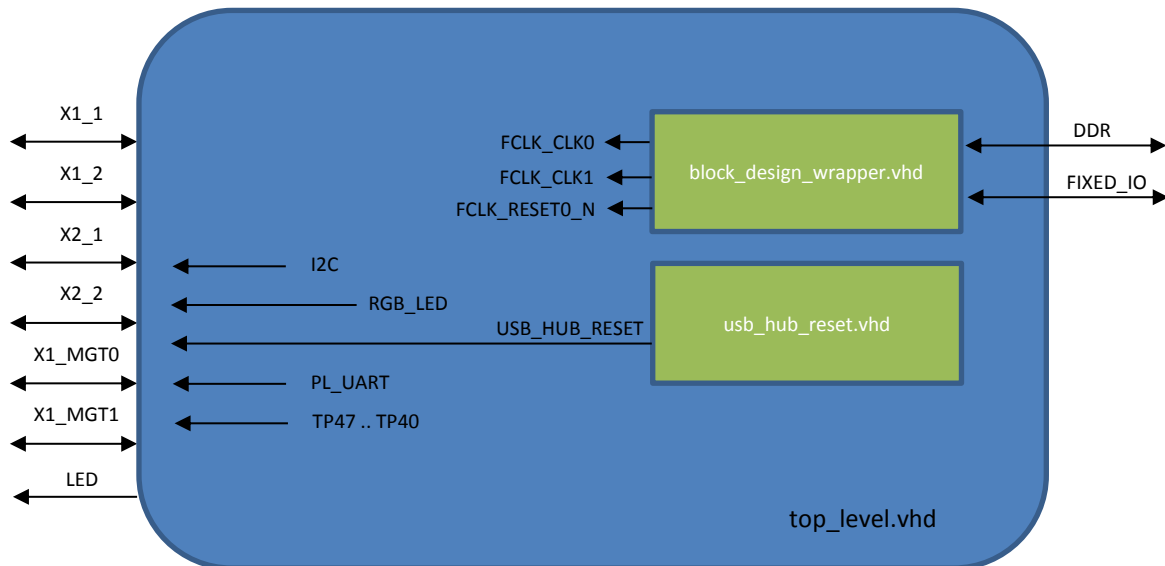
## Script Resources

The resources used by the scripts can all be found in the folder src.

Name	Contents
<b>bd</b>	Vivado IP integrator block diagrams in TCL script form.
<b>constraints</b>	Xilinx constraint files for pin assignment, setting IO standards etc.
<b>hdl</b>	HDL top level module and board package.
<b>presets</b>	Presets for the Zynq 7000 series processing system for specific modules and PL-DDR pinout constraints. These are not used in build.tcl because all this information is in the block diagram TCL scripts. Provided for users that wish to generate their project by themselves.

## Top Level Module

The default Top Level Module contains the definition of all IO ports as well as one instance of the Processing-System Block Design.



The IO-groups `X1_1`, `X1_2`, `X2_1`, `X2_2` as well as the MGT-groups `X1_MGT0` and `X1_MGT1` refer to the IO-Group naming of the carrier board schematics. Their location attributes are set in the added XDC constraints file. In order to connect something to these ports the user can simply connect VHDL modules to these Top Level ports.

`X1_1`, `X1_2`, `X2_1` and `X2_2` are of type `iogroup_t`. They each contain 4 Byte lanes of 12 bits labeled `t0`, `t1`, `t2` and `t3`. In order to access a single bit of an `iogroup_t` record, use the following syntax: `IOgroup.ByteLane(index)`. For example to access bit 4 of Byte lane `t2` of IOgroup `X2_1` use: `"X2_1.t2(4)"`. The Definition of `iogroup_t` can be found in the carrier package (`krcXXXX_pkg.vhd`).

The `mgtgroup0_t` port `X1_MGT0` and `mgtgroup1_t` port `X1_MGT1` consist of one reference clock and four RXTX pairs each. The naming of the MGT signals follows the carrier schematic. For example: in order to access the `N` pin of TX 01 use the following syntax: `"X1_MGT0.TX_01_N"`. To access pin `P` of reference clock 1 use: `"X1_MGT1.CLK_1_P"`.

The Definition of `mgtgroup0_t` and `mgtgroup1_t` can be found in the carrier package (`krcXXXX_pkg.vhd`).

`USB_HUB_RESET` is generated by the `usb_hub_reset` block and automatically resets the USB hub at startup on the KRCXXXX boards.

`I2C` and `RGB_LED` are both defined and inactive by default. To use them they need to be connected to something in `top_level.vhd`.



## Quick Example: Hello World

- Double click make.bat.
- Follow the on screen instructions, hit enter and wait for a few moments.
- Open the newly created project folder and open the .xpr file.
- Run Synthesis
- Run Implementation
- Generate Bitstream
- Open the Hardware Manager
- File -> Export -> Export Hardware (check include Bitstream)
- File -> Launch SDK (Default Settings)
- In SDK: File -> New -> Application Project
- Choose a name, click next
- Select Hello World, click finish
- Establish a serial connection (115200 Baud, 8N1) between the KRC3701 carrier board and the host computer.
- Xilinx Tools -> Program FPGA -> Program
- Right Click Application Project -> Run As -> Launch on Hardware
- You should see the "Hello World" message pop up in the console.