Producing an Animation Using AXI Timer Interrupt

**Goal**

* Integrate AXI Timer into the design from VGA tutorial
* Enable and use interrupt from AXI Timer to control animation

**Requirements**

* Xilinx Vivado software
* Xilinx Nexys 4 board and a programming cable
* Enough disk space for the project files
* Existing VGA tutorial project

**Background**

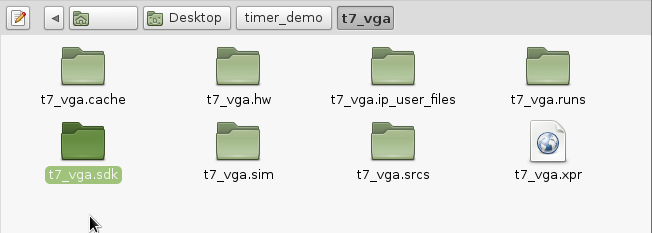
Recall from the VGA tutorial that we used C loops to delay the drawing process in order to produce an animation. The software will loop for multiple cycles doing no effective work to achieve a frame rate close to 30 FPS. This is not effective as we are consuming unnecessary processing power. Note that this is similar to the polling method used in devices. An alternative method to achieve delays in animation is to use a device counter and produce an **interrupt** signal to the processor and only update the drawings when interrupted. During the counting process, the processor can carryout other meaningful computations instead of polling.

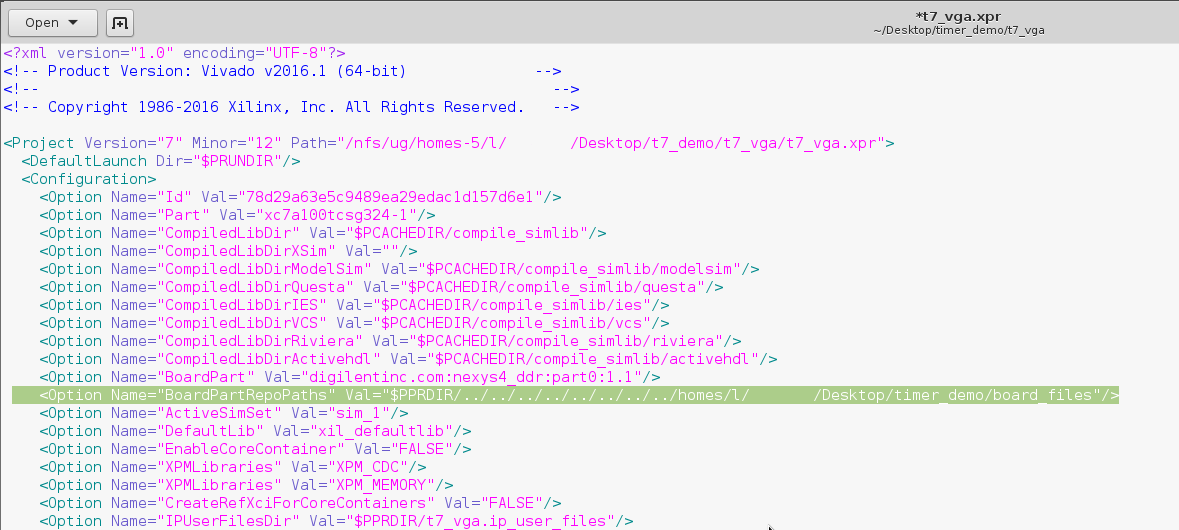
**Introduction**

In this tutorial you will reproduce the animation done in the VGA tutorial using a device timer and interrupts. Instead of creating a new project, we will work on top of the old VGA project to save time.

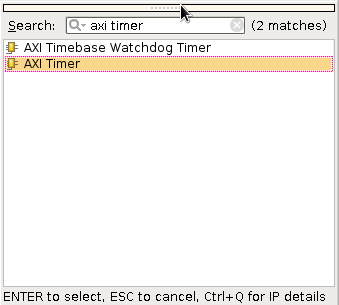
1. **Update the Hardware**

In this section, we will update the hardware aspect of the VGA tutorial to include an AXI Timer

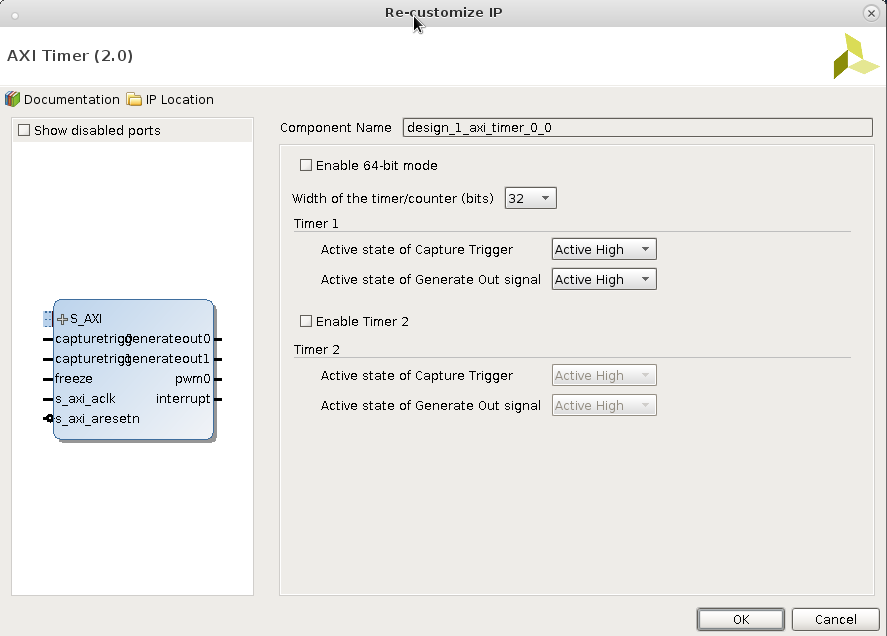
* 1. Create a backup of the VGA lab
  2. Open the VGA lab project folder and delete the SDK file
  3. In the project folder find the .xpr project file and use a text editor to open it
  4. Go to line 20, BoardPartRepoPaths, make sure that the directory to your board file Nexys 4 DDR is still effective.



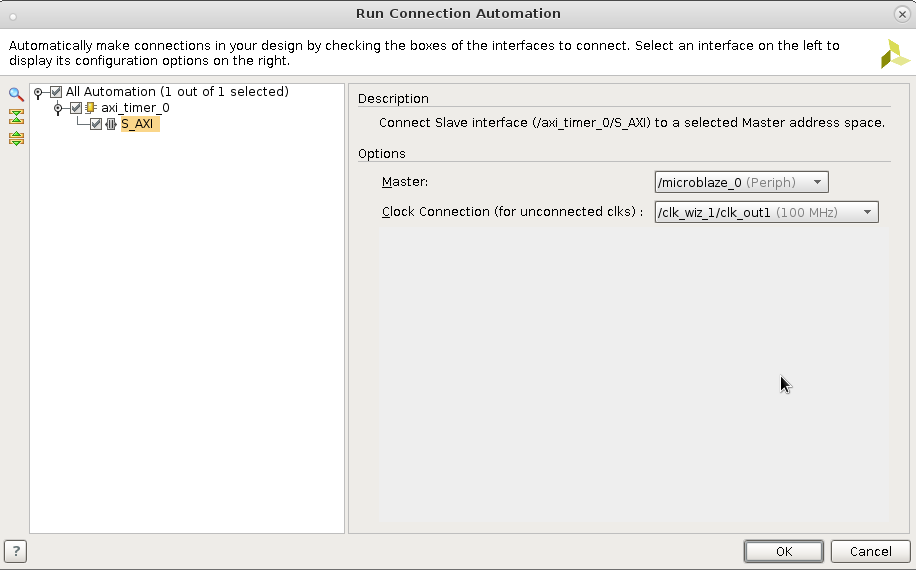
* 1. Open Vivado and open the new VGA project
  2. Open the block design
  3. Right click anywhere in the block design and select Add IP to add AXI Timer



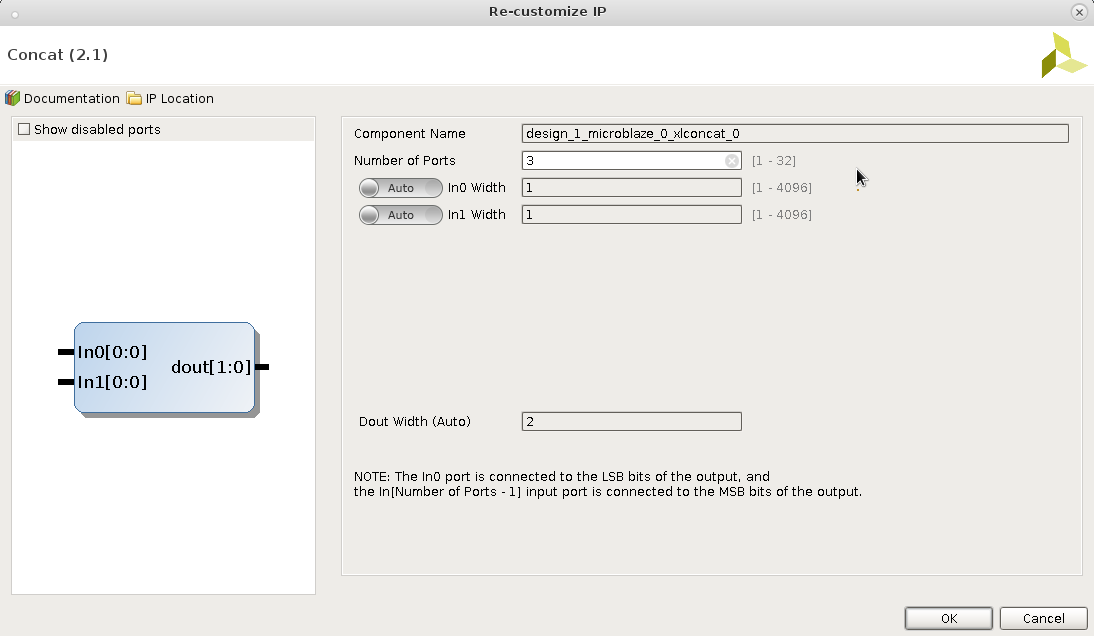
* 1. Double click the AXI Timer to reconfigure
  2. Uncheck Enable Timer 2 as we only will only use one timer and click OK.



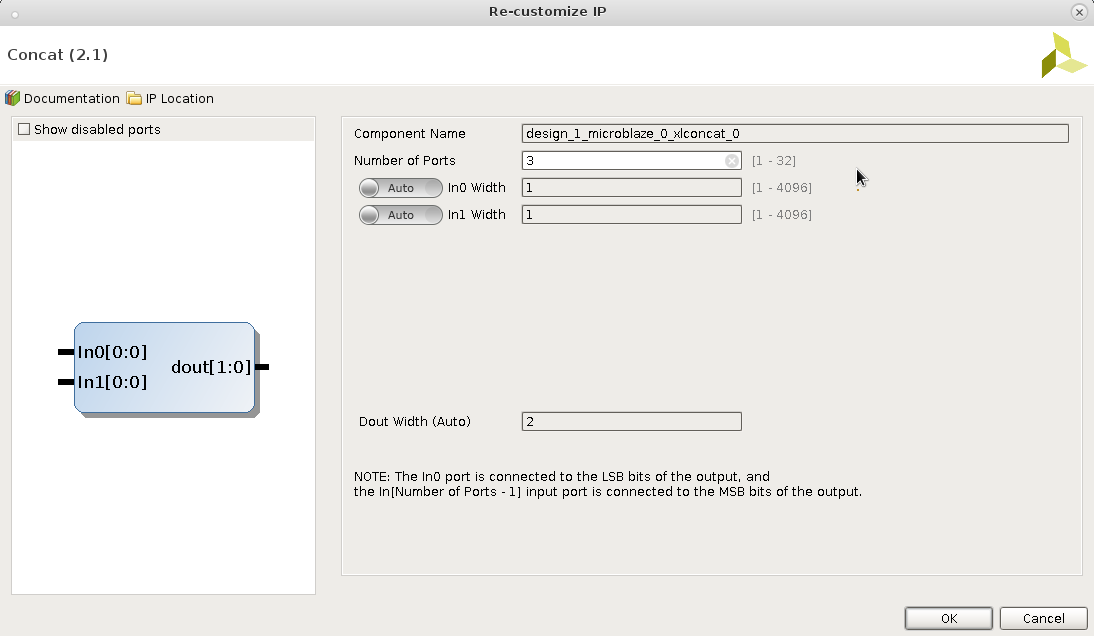
* 1. Run connection automation
  2. For the clock connection in AXI Timer, select the 100Mhz system clock and click OK



* 1. Double click the concat IP and add in additional input wire: In2



* 1. Manually connect the interrupt of the AXI Timer to In2 of concat

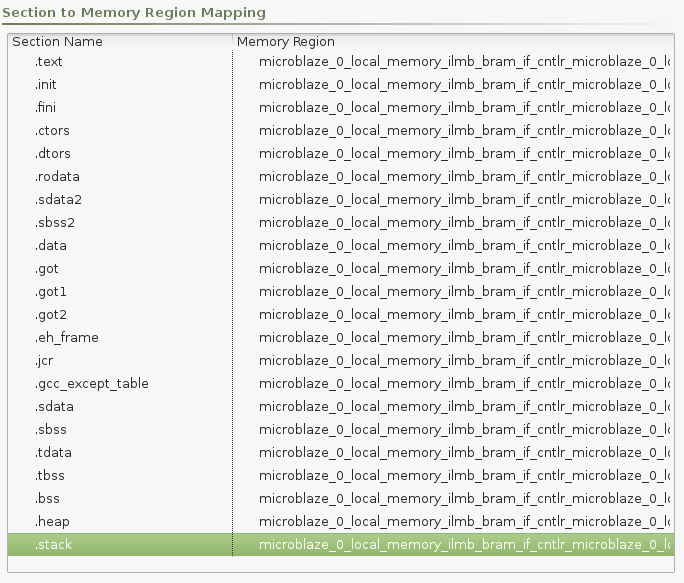


* 1. Run Validate and ensure that it was successful
  2. Regenerate bit stream
  3. Export to hardware including the bit stream
  4. Run SDK to set up the software

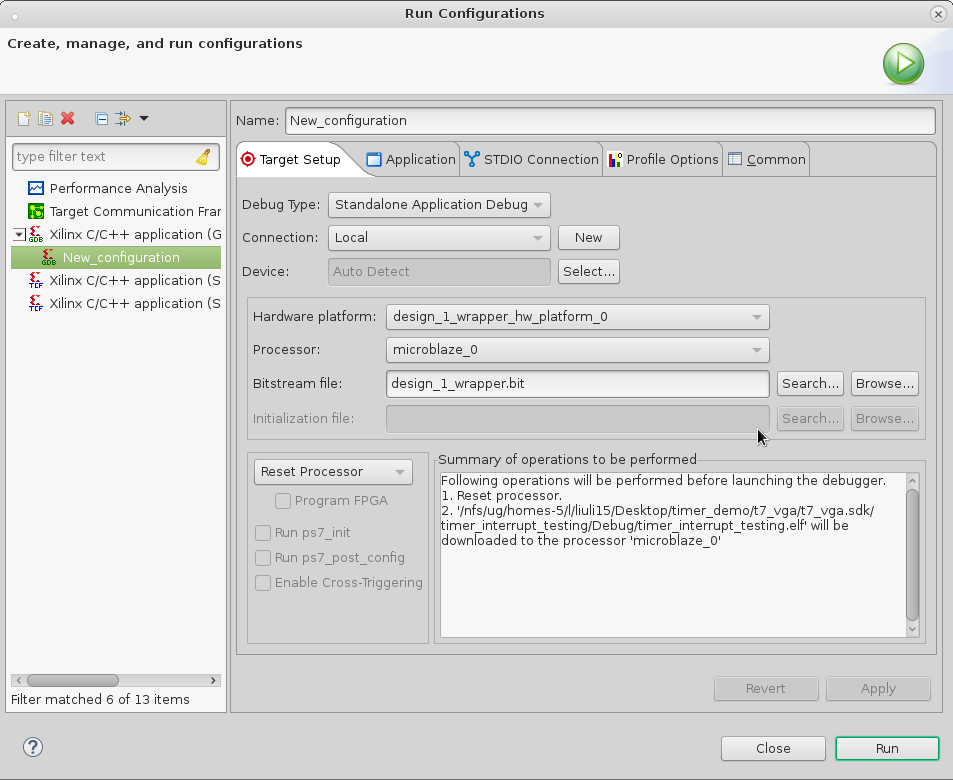
1. **Setup the Software**

In this section, we will setup the software aspect to enable AXI Timer and allow the processor to accept interrupt from AXI Timer

* 1. Create a new empty Xilinx Application
  2. Under Source, create a new C source
  3. Copy and paste the provided source code into your new C source file
  4. Double click the linker file: lscript.ld and change all values under Memory Region Mapping to MicroBlaze local memory



* 1. Save and compile
  2. Under Xilinx Tools, click Program FPGA. Make sure your board is connected to the computer and is ON.
  3. Under Run, click Run Configuration
  4. Double click on Xilinx Application GDB
  5. Under Target Setup select Reset Entire System and your Xilinx application



* 1. Make sure that the CPU reset (SW5) is in high state and click Run

1. **Exercise**

Carefully examine the provided source code and the AXI Timer documentation. Instead of using interrupt, change the operating mode of the Timer to polling mode instead.