# ARM Experiment -10

## Building an SoC by interfacing Timer peripheral with ARM Cortex M0

### Introduction

The purpose of this experiment is to build a System on Chip-Integrating a Timer peripheral with ARM Cortex M0 processor using AHB Lite bus interface and to Synthesis and Check the behavior of the same.

### Objective

- Insert a value maximum count of 0F to load value register.
- Interface Timer peripheral that runs in Compare mode (periodic mode) without prescaler (ie, counter runs at HCLK-higher frequency) with ARM Cortex M0 processor.

### Software tools Requirement

**Modelsim (Siemens)/ Xilinx Vivado/ Icarus Verilog**

**arm Keil µvision 5.37**

### Software programming:

Program the Cortex-M0 processor using arm assembly language and generate the hex file using **arm Keil µvision 5.37**

### Synthesis

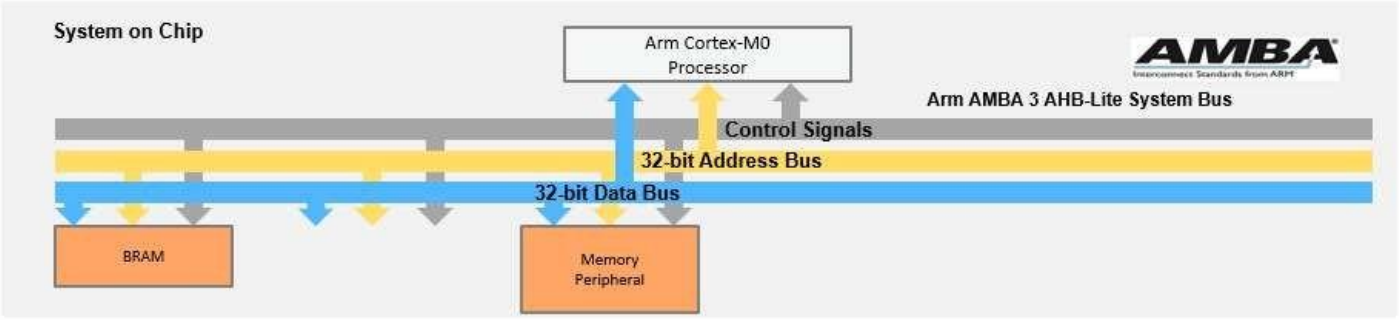
Synthesis the same on ARTY A7 FPGA Kit.
Results should have

Synthesis Report
RTL Schematic
Pin Mapping Report

Screenshot of the Remote Lab-Showing outputs (if any)

**Note:** You can use Virtual Input /Output to give inputs and View outputs or you may use Chipscope (ILA) for viewing the outputs.

## Block Diagram




## Memory Map of Peripherals

| Peripheral | Base address | End address | Size |
|---|---|---|---|
| SRAM | 0x0000_0000 | 0x00FF_FFFF | 16MB |
| Timer | 0x5200_0000 | 0x52FF_FFFF | 16MB |

## Timer Peripheral Registers

| Register | Base address | Size |
|---|---|---|
| Load value | 0x5200_0000 | 4 bytes |
| Current value | 0x5200_0004 | 4 bytes |
| Control value | 0x5200_0008 | 4 bytes |

## Outcome

After this experiment, the learner would get a basic idea about designing a simple SoC based on arm cores, how to interface peripherals to the core using the AHB Lite bus, how to program the processor using Assembly language and how to download the program onto FPGA and check the output.

## Reference

Demo video in session 13