On Chip Memory Generator (Beta Release)

Version 0.1

March 4, 2024

Copyright

Trademarks

Disclaimers

Contents

IP Summary

Introduction

The On Chip Memory Generator is a pre-designed and pre-verified memory IP that can be easily integrated into FPGA and ASIC designs. It is based on the Block Random Access Memory (BRAM) technology and provides a large amount of memory resources for storing data that is frequently accessed or changed. On Chip Memory IP Core is customizable in terms of data width and depth, allowing designers to choose the best option for their specific design requirements.

Features

- On Chip Memory can be configured as Single Port RAM, Simple Dual Port RAM or True Dual Port RAM.
- Support to have multiple read and write ports, allowing multiple operations to occur simultaneously.
- Support configurable width and depth at design time, allowing for flexibility in memory size.
- Support symmetric and asymmetric read/write widths.
- Support independent clocking on each Port.
- Support to access 1 to 128 bit wide memory in symmetric mode and 9, 18, 36, 72, 144, 288 and 576 bit wide memory in asymmetric mode.
- Support depths from 2 to 32768 in symmetric mode and 1024, 2048, 4096, 8192, 16384 and 32768 in asymmetric mode.
- Support mapping feature on Block RAM or Distributed RAM (LUTs).

Overview

On Chip Memory Generator

On Chip Memory Generator can generate the RTL (register-transfer level) code for on-chip memory components. It allows user to build any size of memory using embedded BRAM with custom parameters for depth, bit-width, and number of ports, without the need for manual RTL coding. The Figure 1 shows the block diagram of On Chip Memory Generator.

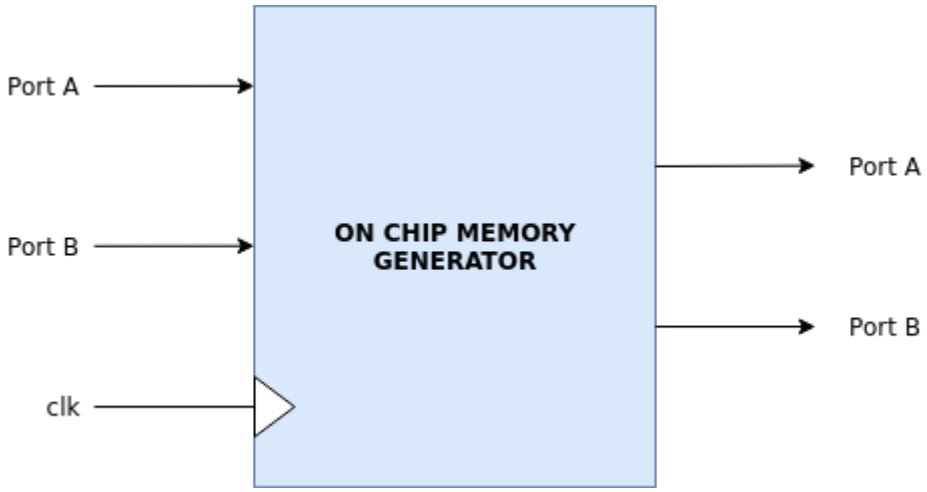

Figure 1: OCM Generator Block Diagram

IP Specification

On-Chip Memory Generator supports data widths from 1 to 128 in symmetric mode and 9, 18, 36, 72, 144, 288 and 576 bits in asymmetric. Additionally, it allows for write depth from 2 to 32768 in symmetric and 1024, 2048, 4096, 8192, 16384 and 32768 in asymmetric mode provides a flexible and scalable memory solution. It supports separate clocks for two ports, allowing for independent control of each port's read and write operations. It also offers three different memory types: Single Port, Simple Dual Port, and True Dual Port. Single Port provides access to the memory through a single clock and data port, Simple Dual Port provides basic two-port functionality with independent clock, and True Dual Port provides two independent ports with separate clocks for maximum flexibility and performance. It is a versatile and high-performance solution for embedded systems and integrated circuits with memory requirements. The Figure 2 shows the top level diagram of On Chip Memory.

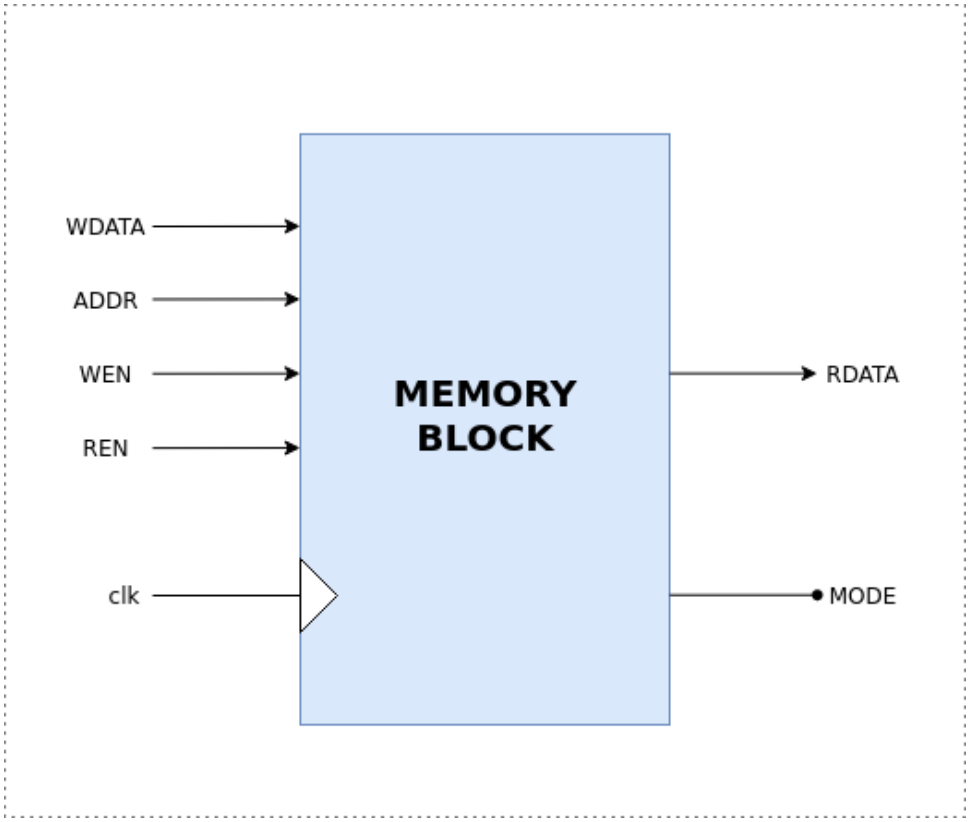

Figure 2: Top Module

Memory Types

Three types of memories are supported by On Chip Memory Generator. These are:

- **Single Port RAM:**
  A Single-Port RAM is a type of Random Access Memory that has only one port for reading and writing data. It allows only one operation (read or write) at a time, thus making it less flexible compared to a dual-port RAM which has two ports allowing for both read and write operations to occur simultaneously. Single-port RAM is commonly used in applications where cost and/or chip area are a concern, as they are simpler to implement and consume less resources compared to a dual-port RAM. It is shown in Figure 3.

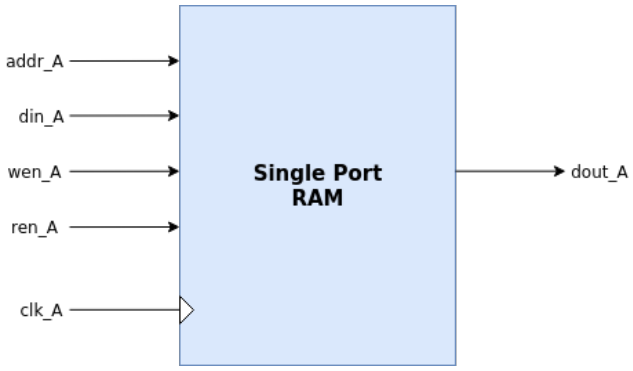

Figure 3: Single Port RAM

- **Simple Dual Port RAM:**
  A Simple Dual-Port RAM is a type of Random Access Memory that has two independent ports, allowing for simultaneous read and write operations. This makes it more flexible compared to a single-port RAM, as it can handle multiple access requests at the same time. It is commonly used in applications where multiple processors or systems need to access shared memory simultaneously. It is shown in Figure 4.

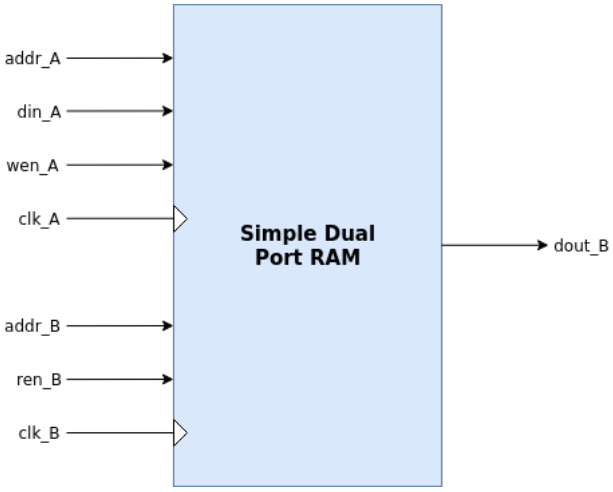

Figure 4: Simple Dual Port RAM

- **True Dual Port RAM:**
  A True Dual-Port RAM is a type of Random Access Memory that has two completely independent ports, allowing for truly simultaneous and non-interfering read and write operations. Unlike a simple dual-port RAM, which may share certain elements such as data buses or address decoders, a true dual-port RAM has completely separate components for each port, allowing for truly independent and parallel access to the memory. True dual-port RAMs are commonly used in applications where the highest level of parallelism is required, such as in high-speed communication systems, real-time video processing, and other high-performance applications. It is shown in Figure 5.

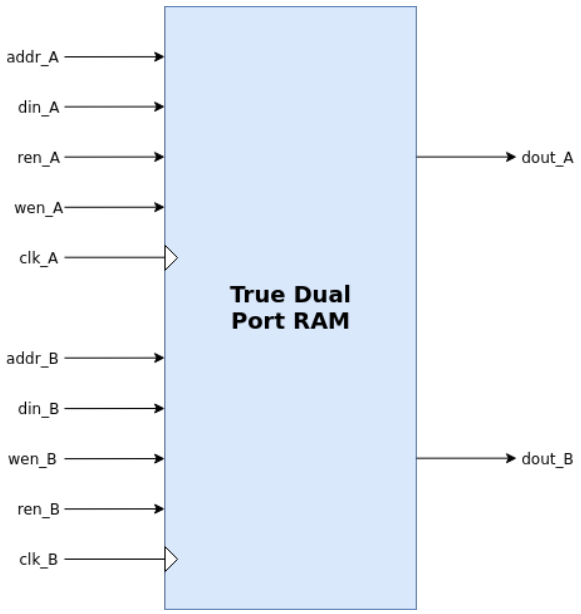

Figure 5: True Dual Port RAM

Common Clock Versus Uncommon Clock

COMMON_CLK parameter allows user to control the synchronization of read and write operations within the memory. It serves a critical role in maintaining data integrity and ensuring predictable behavior.

- When the COMMON_CLK parameter is enabled, a single clock signal governs both read and write operations. This synchronization ensures that all data access occurs at precisely defined moments in time, eliminating the risk of data glitches or conflicts that could arise from asynchronous access attempts. This mode is ideal for applications requiring high data integrity and deterministic behavior.
- When the COMMON_CLK parameter is disabled, read and write operations can utilize separate clocks. This mode provides flexibility for specific use cases where independent clock domains are necessary. However, it is crucial to carefully design your system to account for potential timing hazards that may arise due to asynchronous access. Using separate clocks requires implementing additional logic to ensure that read and write operations do not conflict, which can add complexity to the design and increase the risk of errors.

Block RAM Versus Distributed RAM

Two types of memory blocks that can be generated with On Chip Memory Generator are Block RAM and Distributed RAM. When opting for Block RAM, the logic is mapped onto embedded BRAM, forming a large, rectangular array of memory cells optimized for rapid access. On the other hand, selecting Distributed RAM results in the mapping of logic onto Look-Up Tables (LUTs), constituting a flexible array of smaller, distributed memory cells suitable for storing modest amounts of data. The decision between these memory block types hinges on the specific design requirements.

IP Support Details

The Table 1 gives the support details for On Chip Memory.

Compliance		IP Resources				Tool Flow		
Device	**Interface**	**Source Files**	**Constraint File**	**Testbench**	**Simulation Model**	**Analyze and Elaboration**	**Simulation**	**Synthesis**
VIRGO	-	Verilog	-	Verilog	TDP_RAM36K	Raptor	Raptor	Raptor

Table 1: Support Details

Resource Utilization

The parameters for computing the maximum and the minimum resource utilization are given in Table 2, remaining parameters have been kept at their default values.

Tool	Raptor Design Suite			
FPGA Device	VIRGO			
Configuration			**Resource Utilization**	
Resources (Distributed)	**Options**	**Configuration**	**Resources**	**Utilized**
	MEMORY_TYPE	Single_Port	BRAMS	0
	PORT_TYPE	Symmetric	LUTS	12014
	MEMORY_MAPPING	Dis-tributed_RAM	REGISTERS	32821
	DATA_WIDTH	32	-	-
	WRITE_DEPTH	1024	-	-
Minimum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	MEMORY_TYPE	Single_Port	BRAMS	1
	PORT_TYPE	Symmetric	LUTS	0
	MEMORY_MAPPING	Block_RAM	REGISTERS	0
	DATA_WIDTH	36	-	-
	WRITE_DEPTH	1024	-	-
Maximum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	MEMORY_TYPE	Single_Port	BRAMS	144
	PORT_TYPE	Asymmetric	LUTS	0
	MEMORY_MAPPING	Block_RAM	REGISTERS	0
	WRITE_WIDTH_A	576	-	-
	READ_WIDTH_A	576	-	-
	WRITE_DEPTH_A	8192	-	-

Table 2: Resource Utilization

Ports

Table 3 lists the top interface ports of the On Chip Memory.

Signal Name	Bit Width	Input/Output	Description
clk	[0:0]	Input	Clock signal for Port A, B
Port A			
clk_A	[0:0]	Input	Clock for Port A
addr_A	[log2(WRITE_DEPTH)-1:0]	Input	Address Width of Port A
din_A	[DATA_WIDTH-1:0]	Input	Input Data of Port A
wen_A	[0:0]	Input	Write Enable to Port A
ren_A	[0:0]	Input	Read Enable to Port A
dout_A	[DATA_WIDTH-1:0]	Output	Output Data of Port A
Port B			
clk_B	[0:0]	Input	Clock for Port B
addr_B	[log2(WRITE_DEPTH)-1:0]	Input	Address Width of Port B
din_B	[DATA_WIDTH-1:0]	Input	Input Data of Port B
wen_B	[0:0]	Input	Write Enable to Port B
ren_B	[0:0]	Input	Read Enable to Port B
dout_B	[DATA_WIDTH-1:0]	Output	Output Data of Port B

Table 3: Ports

Parameters

Table 4 lists the parameters of the On Chip Memory.

Parameter	Values	Default Value	Description
MEMORY_TYPE	Single Port, Simple Dual Port, True Dual Port	Single Port	Desired Memory Type
DATA_WIDTH	1 to 128	32	Data Width of Memory
WRITE_DEPTH	2 to 32768	1024	Depth of Memory
WRITE_WIDTH_A	9,18,36,72,144, 288,576	36	Write Data Width of Port A
READ_WIDTH_A	9,18,36,72,144, 288,576	36	Read Data Width of Port A
WRITE_WIDTH_B	9,18,36,72,144, 288,576	36	Write Data Width of Port B
READ_WIDTH_B	9,18,36,72,144, 288,576	36	Read Data Width of Port B
WRITE_DEPTH_A	1024,2048,4096, 8192,16384,32768	1024	Write Depth of Port A
PORT_TYPE	Symmetric, Asymmetric	Symmetric	Symmetric vs Asymmetric Ports of Memory
MEMORY_MAPPING	Block_RAM, Distributed_RAM	Block_RAM	Block RAM vs Distributed RAM Mapping
COMMON_CLK	True/False	False	Common Clock for Port Synchronization
IP_TYPE	OCM	OCM	Type of Peripheral
IP_VERSION	1	1	Version of Peripheral
IP_ID	<unique_id>	<unique_id>	ID of Peripheral for Raptor

Table 4: Parameters

Design Flow

IP Customization and Generation

On Chip Memory Generator is a part of the Raptor Design Suite Software. A customized memory can be generated from the Raptor's IP configuration window as shown in Figure 6.

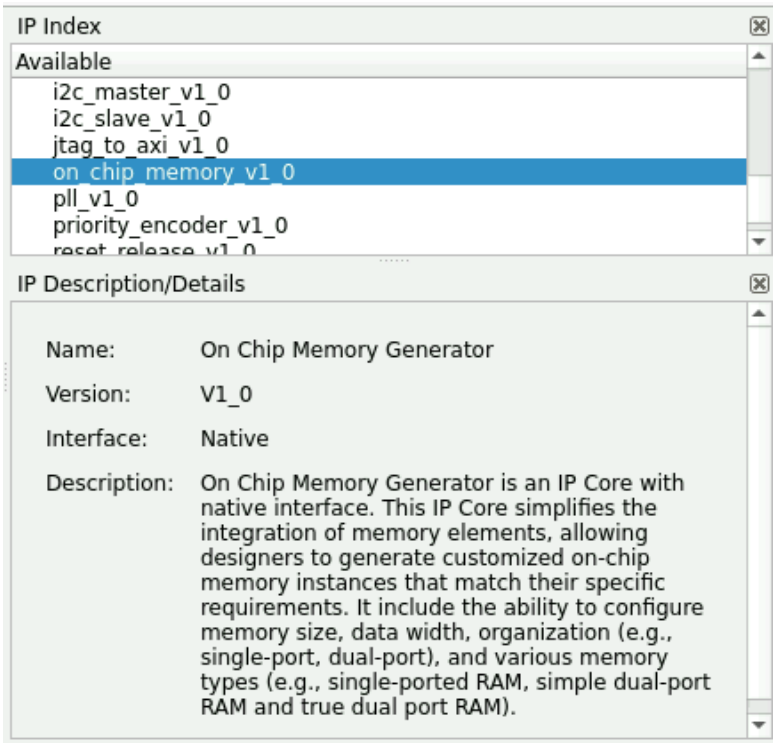

Figure 6: IP List

Parameters Customization

From the IP configuration window, the parameters of the On Chip Memory Generator can be configured and its features can be enabled for generating a customized IP that suits the user application requirements. All parameters are shown in Figure 7.
In Figure 7, Documentation button access IP User Guide and IP Location access the IP path. Module Name specifies the name of both the Verilog file and the top-level IP name that will be generated based on above configured parameters. With MEMORY_TYPE parameter, user may choose the desired type of memory. PORT_TYPE configures the Symmetric and Asymmetric ports for generated memory. In Symmetric mode, user have the flexibility to choose between Block RAM and Distributed RAM. However, in Asymmetric mode, the option is limited to only Block RAM. COMMON_CLK option is available to synchronize read/ write logic. DATA_WIDTH parameter configures the data width for memory and WRITE_DEPTH configures depth of the memory. FILE_PATH is the field for memory initialization through .hex or .bin file. User may provide path to memory file to initialize the memory.

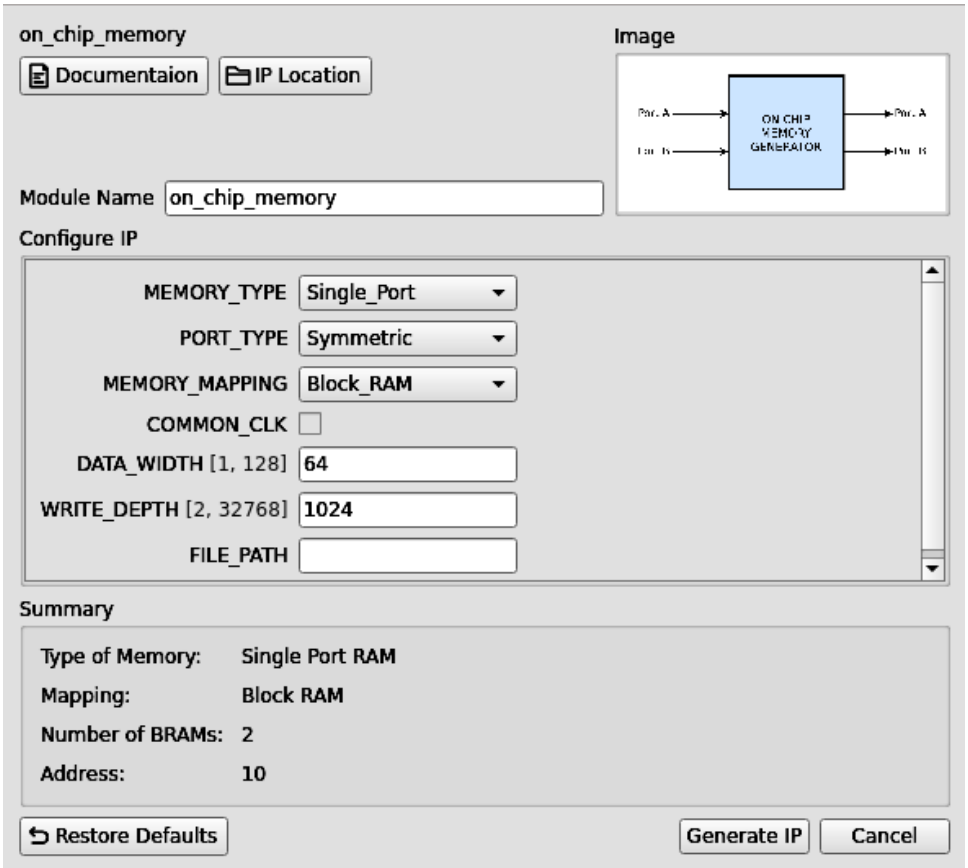

Figure 7: IP Configuration

Synthesis, Place and Route

Raptor Design Suite is armed with tools for Synthesis along with Place and Route capabilities. Raptor uses Yosys as a synthesis engine and VPR for Placement and Routing. The generated post-syntn and post-route netlists may be viewed and analyzed from within the Raptor. The generated bit-stream may be used to program FPGA device.

Testbench

Test

An example testbench may be generated by the user to simulate a Single Port RAM 1024x32. The testbench is written to do a comparison between a behavioural model of single port and generated RTL from On Chip Memory Generator. This test generates 1024 random inputs to addr_A and din_A of both the RTLs and in the end, it compares all the outputs. If all the outputs match, then test will be passed otherwise it'll be failed.

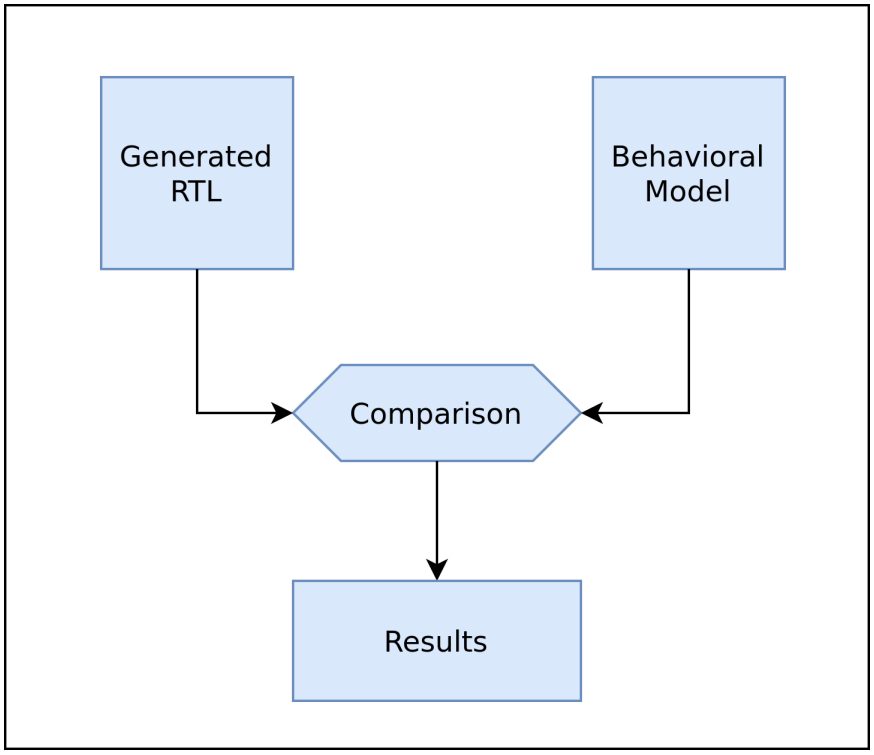

Figure 8: Example Diagram

The results of this test are attached below in Figure 9.

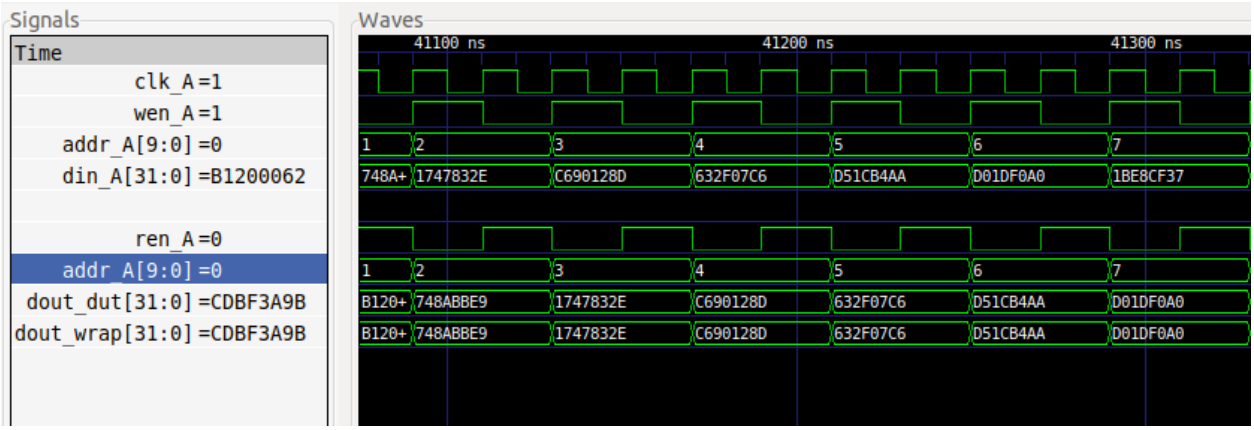

Figure 9: Test Results

Simulation

To run simulation, go to Simulate IP option as shown in Figure 10.

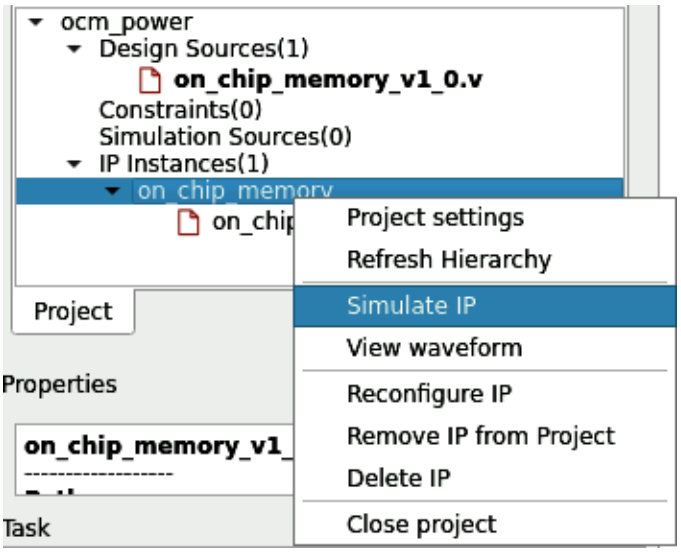

Figure 10: Simulate IP

Waveform

To view waveform, go to View waveform option as shown in Figure 11.

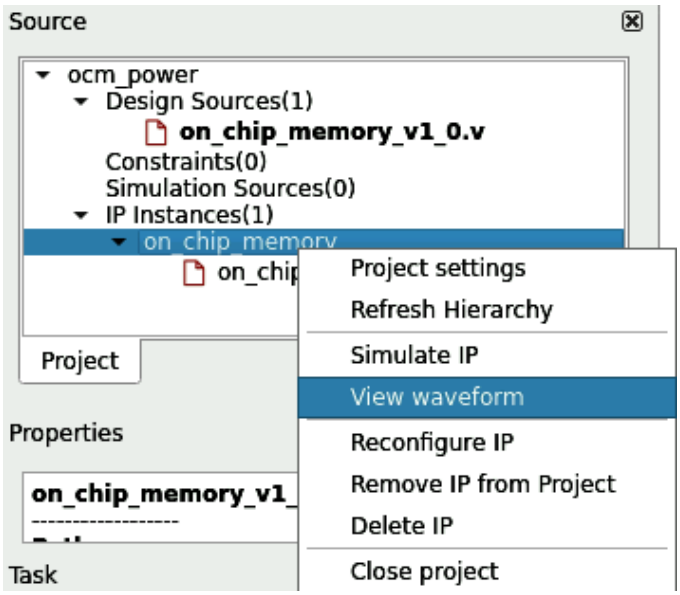

Figure 11: View Waveform

Revision History

Date	Version	Revisions
March 4, 2024	0.1	Initial version On Chip Memory Generator User Guide