

# FPGA Runtime Benchmark

Optimized server configuration for  
Vivado

liux@njupt.edu.cn

# Desire to decrease P&R runtime



FPGAs are bigger with more resources

Designs are more complicated

Designs are Not verified  
exhaustively like ASIC

Time to Market  
requirement

Minimize  
iteration time

# Vivado runtime is a complicated topic

## Design complexity

Small/huge design

Small/huge device part

Low/high design complexity

## Server workload

No additional workload

Small/high additional workload

## Vivado configuration

GUI/Batch mode

Project/No-project mode

## Server configuration

Number of Thread

High Freq/Lower Freq of core

More or less physical cores

DDR4  
1866/2100/2400 bps

# Fix other variables and focus on server configuration

## Design complexity

- Fixed design
- without timing constraints

## Server workload

- Dedicate server
- Minimize workload for other tasks

## Vivado configuration

- Non-project mode
- Without additional time consumption introduced write\_checkpoint, report\_timing\_summary
- Without hard drive related activities

# Benchmark Environment

## CPU- Intel Core i5-6500

- Clock Speed:3.20GHz
- PHY Cores:4
- Cach:6M

## Kingston DDR4 memory 8Gx2

- Max Perf: 2400MHz

## Motherboard

- ASUS 170-A
- Intel Z170

## Software

- Centos Release 6.8 (Final)
- Vivado 2016.4



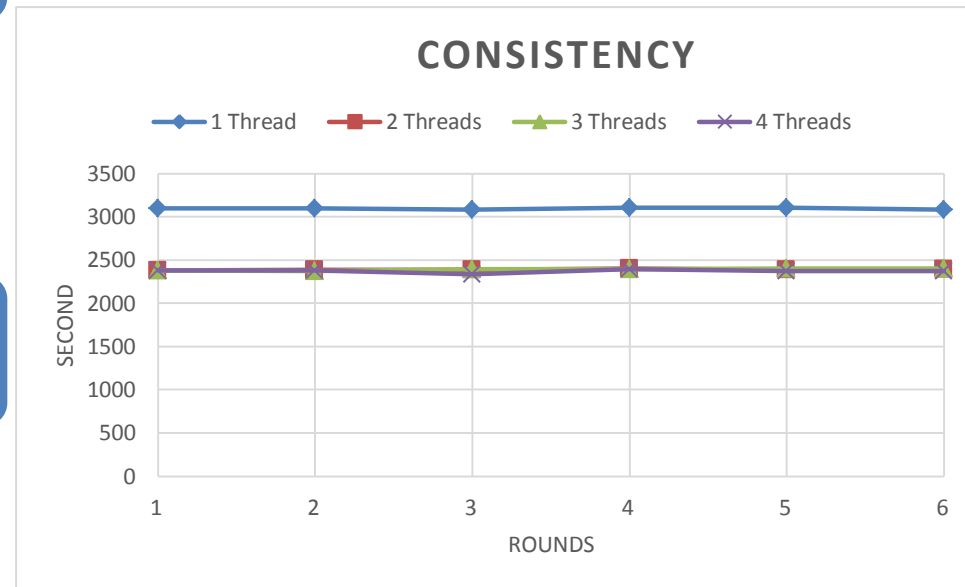
# Runtime consistency

## Tests

- Launched 6 rounds tests
- Each round with 1/2/3/4 threads 4 tests

## Results

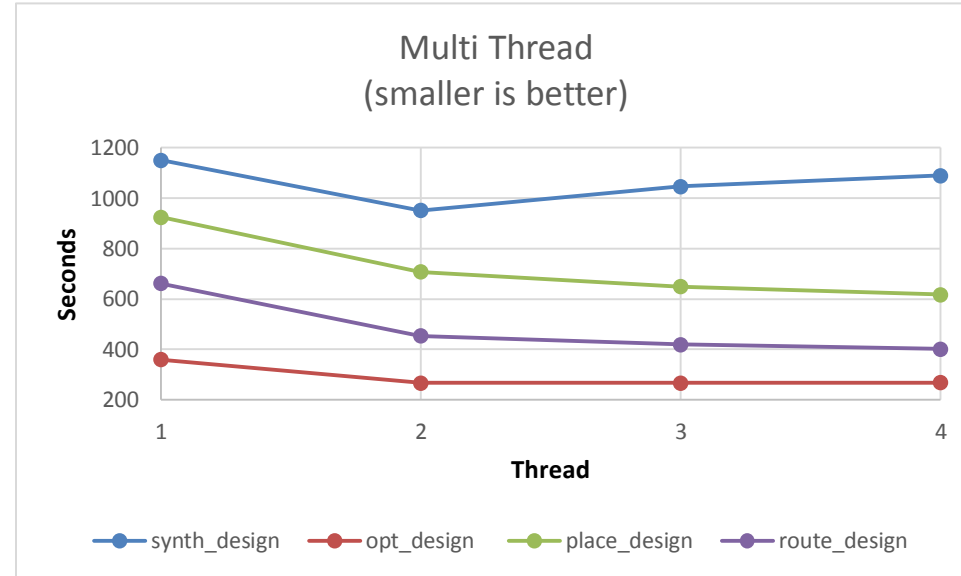
- Each round runtime has limited variation
- Don't need repeat test for further test under certain configuration



# CPU threads (without hyper-thread)

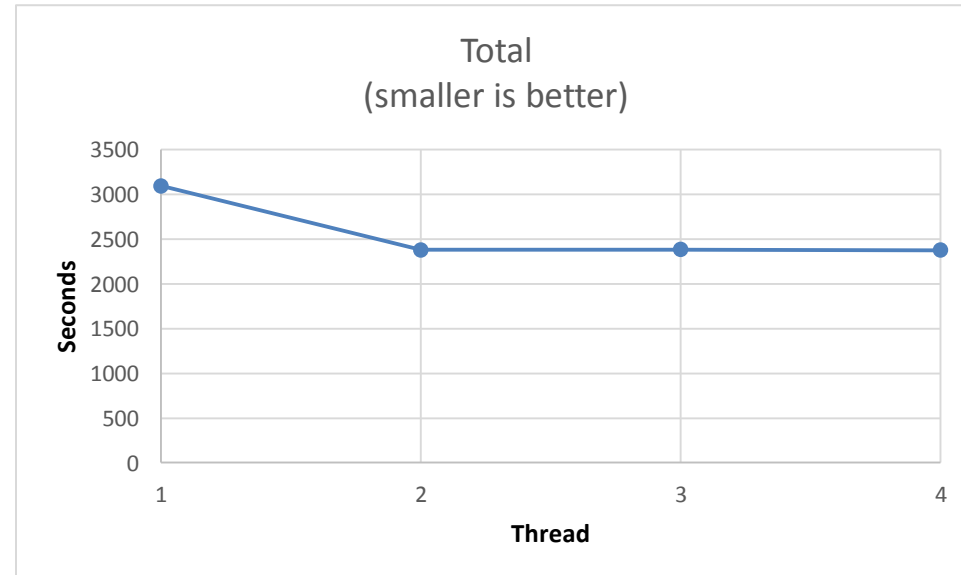
## Break down time

- Synthesis time decreased significantly with 2 threads
- Synthesis time rebound with 3 and 4 threads
- Opt, Place and Route time decreased constantly with more thread enabled



## Total time

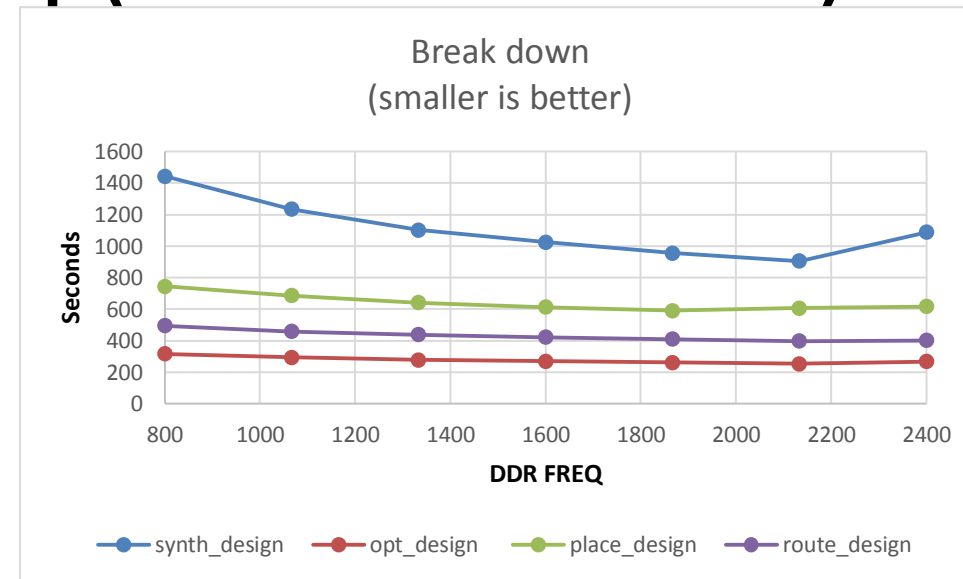
- 23.2% runtime decreased from 1 to 2 thread enabled
- No significant runtime decreased with 3 and 4 threads enabled



# DDR memory Freq (with 4 threads)

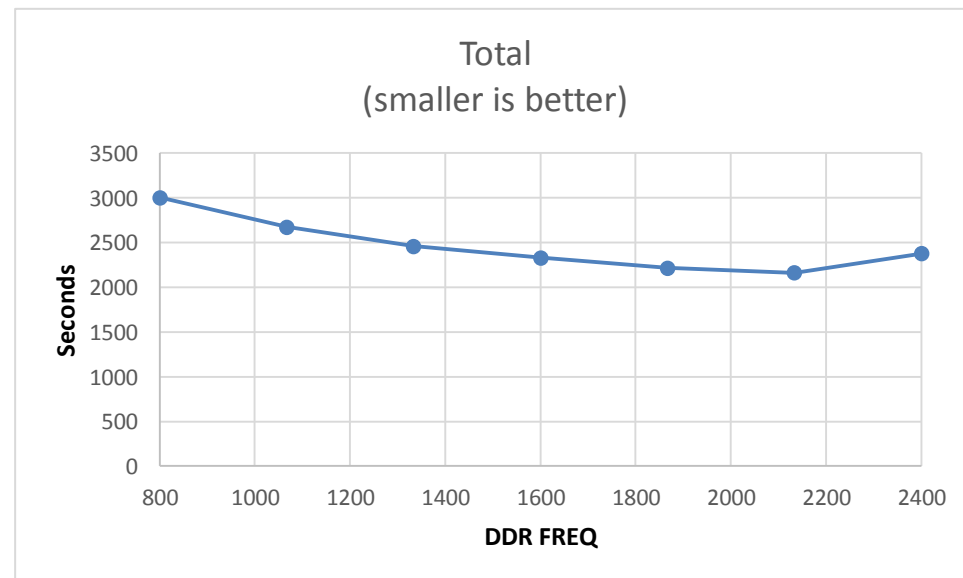
## Break down time

- Synthesis runtime gain is 37% with DDR Freq from 800MHz to 2200MHz
- Opt, Place and Route runtime gains are 19%, 18%, 19% respectively



## Total time

- Total runtime gain is 28% with DDR memory Freq 800MHz to 2200MHz
- Don't understand why runtime rebound with DDR memory Freq from 2200MHz to 2400Mhz





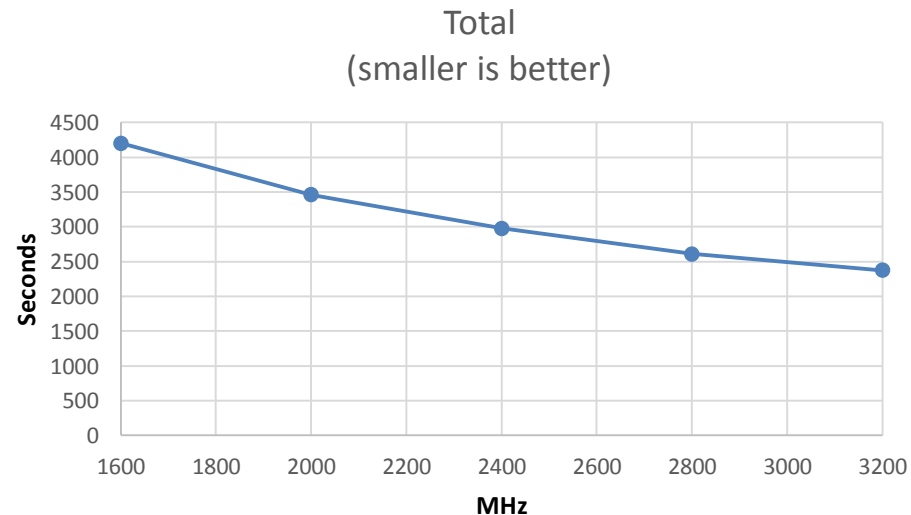
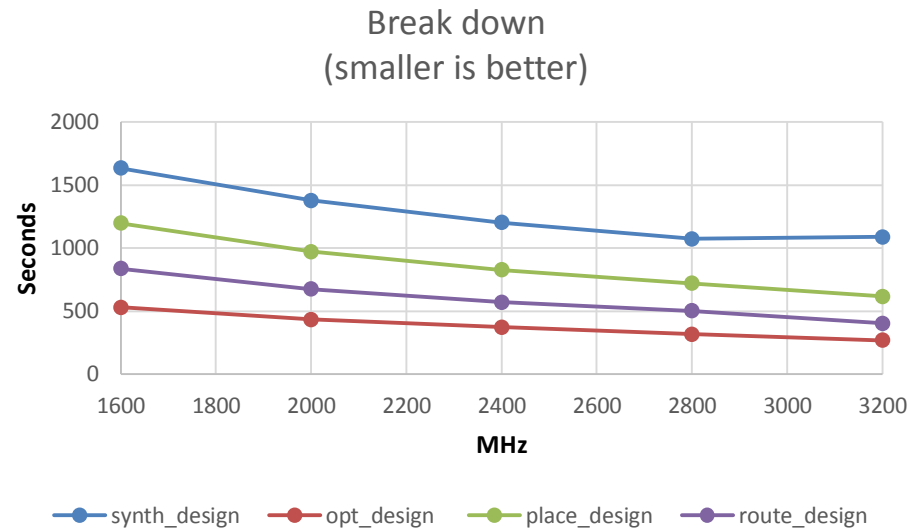
# CPU Freq (with 4 threads)

## Break down time

- Synthesis runtime gain is 36% with CPU Freq from 1600MHz to 2800MHz
- Opt, Place and Route runtime gains are 49%, 48%, 52% respectively with CPU Freq from 1600MHz to 3200MHz

## Total time

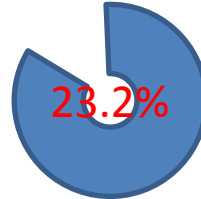
- Total runtime gain is 43.4% with with CPU Freq from 1600MHz to 3200MHz
- Don't understand why runtime rebound with CPU Freq from 2800MHz to 3200MHz on Synth\_design



# Executive summary

## CPU threads

- 23.2% runtime gain from 1 to 2 threads enabled

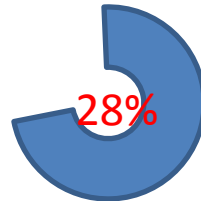


## CPU threads

- Enable thread from 1 – 2 get significant runtime gain
- Enable > 2 threads will help but with limited gains

## DDR Memory Freq

- Total runtime gain is 28% with DDR memory Freq 800MHz to 2200MHz



## DDR Memory Freq

- Increase DDR memory bandwidth will have constant runtime gain from 800MHz to 2200Hz

## CPU Freq

- Total runtime gain is 43.4% with with CPU Freq from 1600MHz to 3200MHz



## CPU Freq

- Total runtime gain is 43.4% with with CPU Freq from 1600MHz to 3200MHz
- Runtime gain is almost linear with CPU Freq increase

# Server configuration suggestion

## CPU selection

- Single thread Performance is the key factor
- Intel Core i3-7345K has highest Perf/Cost efficiency
- Intel Core i7-7700K is the best for performance

## DDR memory selection

- Choose the faster part

### CPU Mark | Price Performance

(Click to select desired chart)

### PassMark - CPU Mark

Single Thread Performance - Updated 14th of April 2017

