OPAL-RT TECHNOLOGIES
FROM IMAGINATION TO REAL-TIME

Finite Element Based, FPGA-Implemented Electric Machine Model for Hardware-in-the-Loop (HIL) Simulation

Presentation Agenda

1. HIL Simulation by OPAL-RT – Introduction & Context
2. E-drive simulation - Why FPGA?
3. PMSM solver on FPGA
4. Integration of Maxwell FEA models and eDRIVEsim
5. Conclusion
6. Q&A

OPAL-RT Introduction

We supply **real-time digital simulators** to industry, research labs and educational institutions for hardware-in-the-loop (HIL), rapid control prototyping and accelerated non-realtime (number crunching) applications

OPAL-RT: Turnkey HIL Simulators

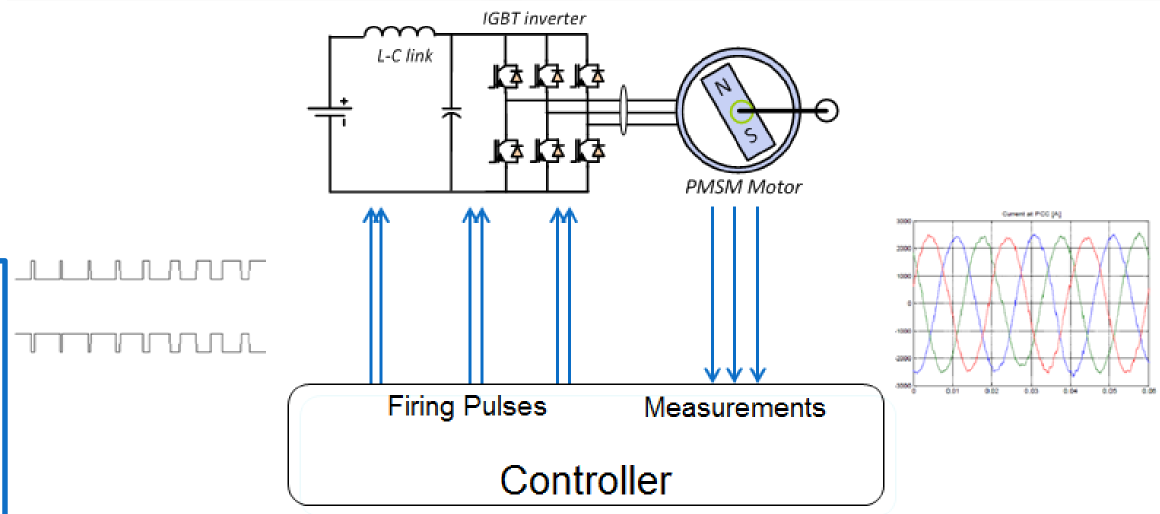
Hardware, software and integration for real-time simulation and testing

- ✓ We **program** sophisticated solvers and interfaces for real-time applications
- ✓ We **design** full range I/O signal processing peripherals (modular mapping boxes, FIU, break out boxes)
- ✓ We **develop/integrate** application models and solutions for various industries (automotive, aerospace, military, power utilities)

The Challenge of Electric Motor Control Testing

Faster time to market with parallel development and accelerated test:
a proven approach with HIL simulation

1. scoping
2. build business case
3. development
4. testing & validation
5. market launch



The Challenge of Electric Motor Control Testing

Motor control engineers want :

- ✓ To test the motor controller with non-ideal behavior.
- ✓ To test the motor controller with different points of operation, such a saturated states
- ✓ To insert fault conditions
- ✓ To rapidly simulate different types of motors

➡ **High-fidelity and flexible motor simulation**

The Challenge of Electric Motor Control Testing

Managers want to:

- increase test case coverage
- reduce costs
- accelerate time to market

✓ By reducing testing time on real dynamometer

✓ By detecting errors at earlier stages of the design

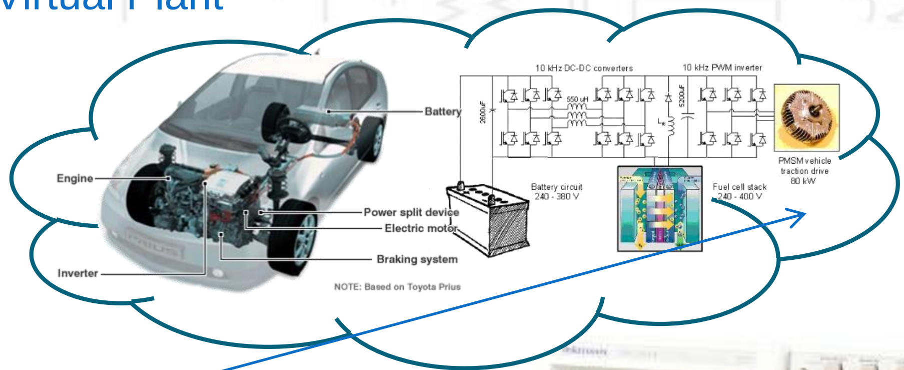
✓ Faster improvement of complex control strategies

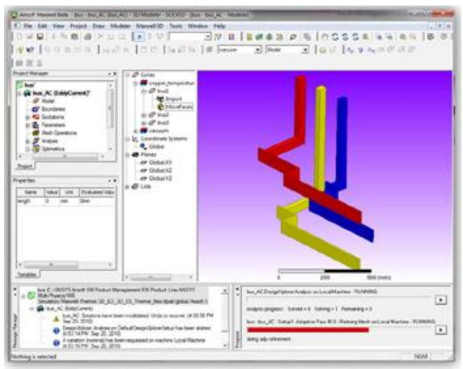
Creation of a technical link between motor designer and control engineer – HIL model IS the design

OPAL-RT

RT-LAB for ECU Testing and Validation

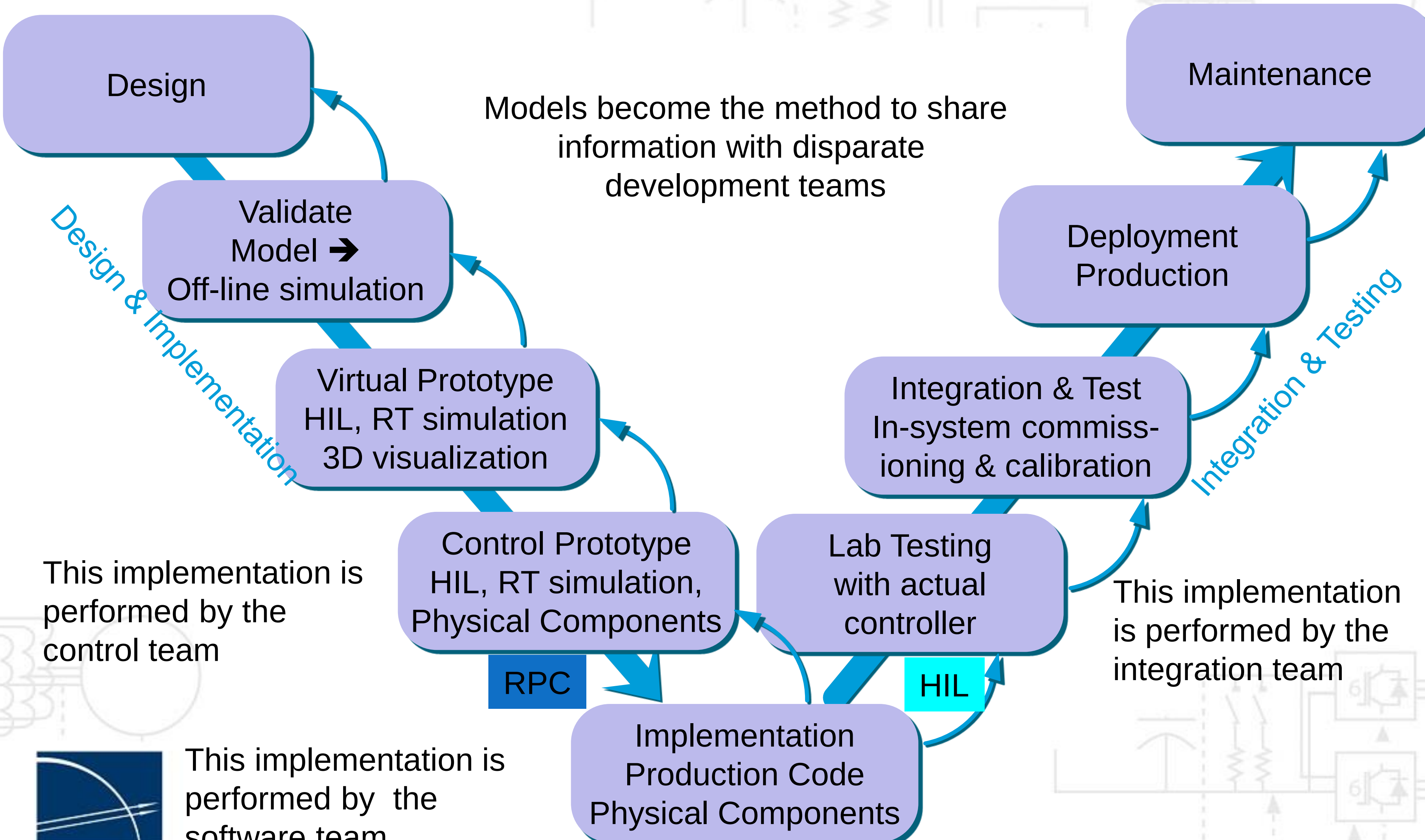
Virtual Plant





Electronic Control Unit (ECU) Under Test

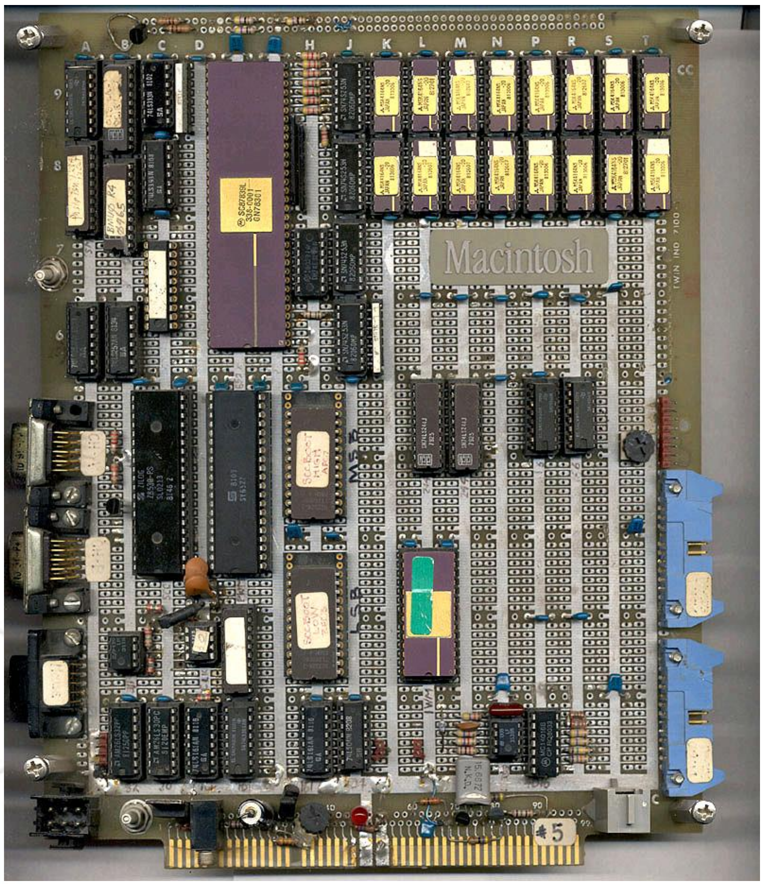
Model-based Design (MBD) & Hardware-in-the-Loop (HIL)

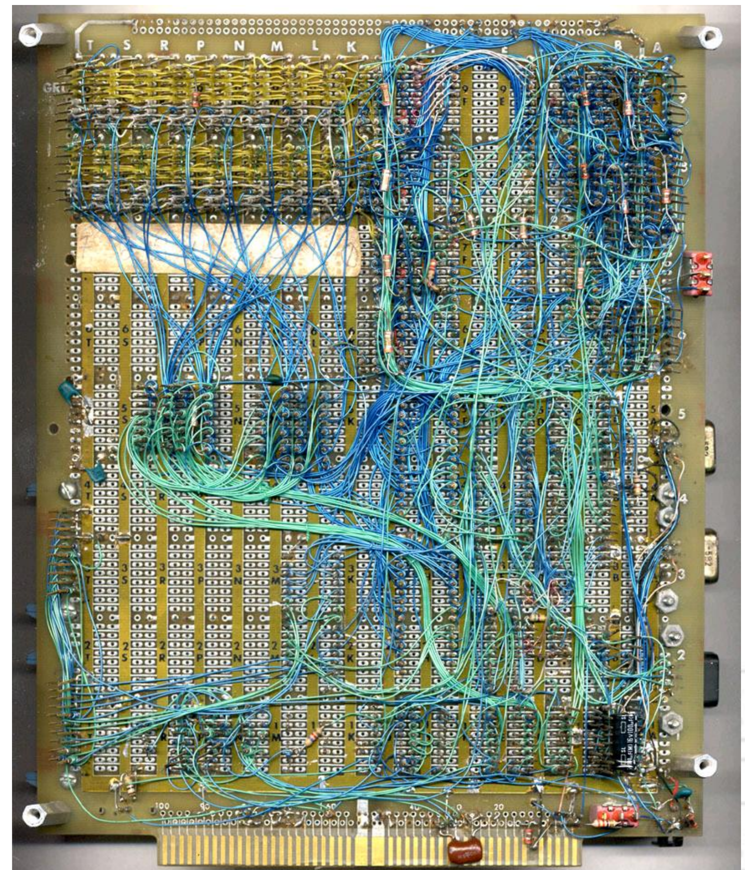


Why use FPGA?

Hang on… First, what's an FPGA???

Well, Before FPGAs, years ago…

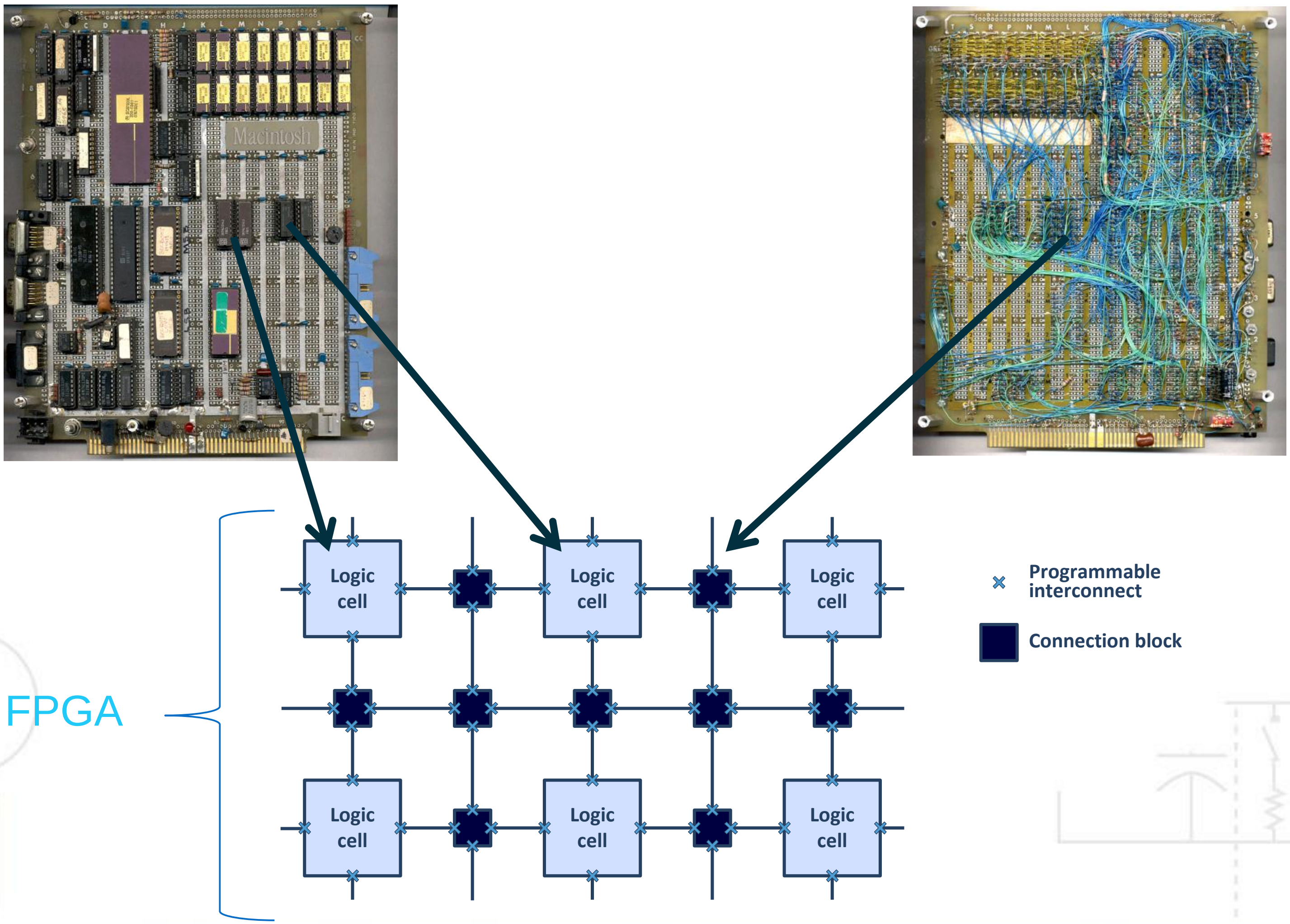




Macintosh prototype, 1980

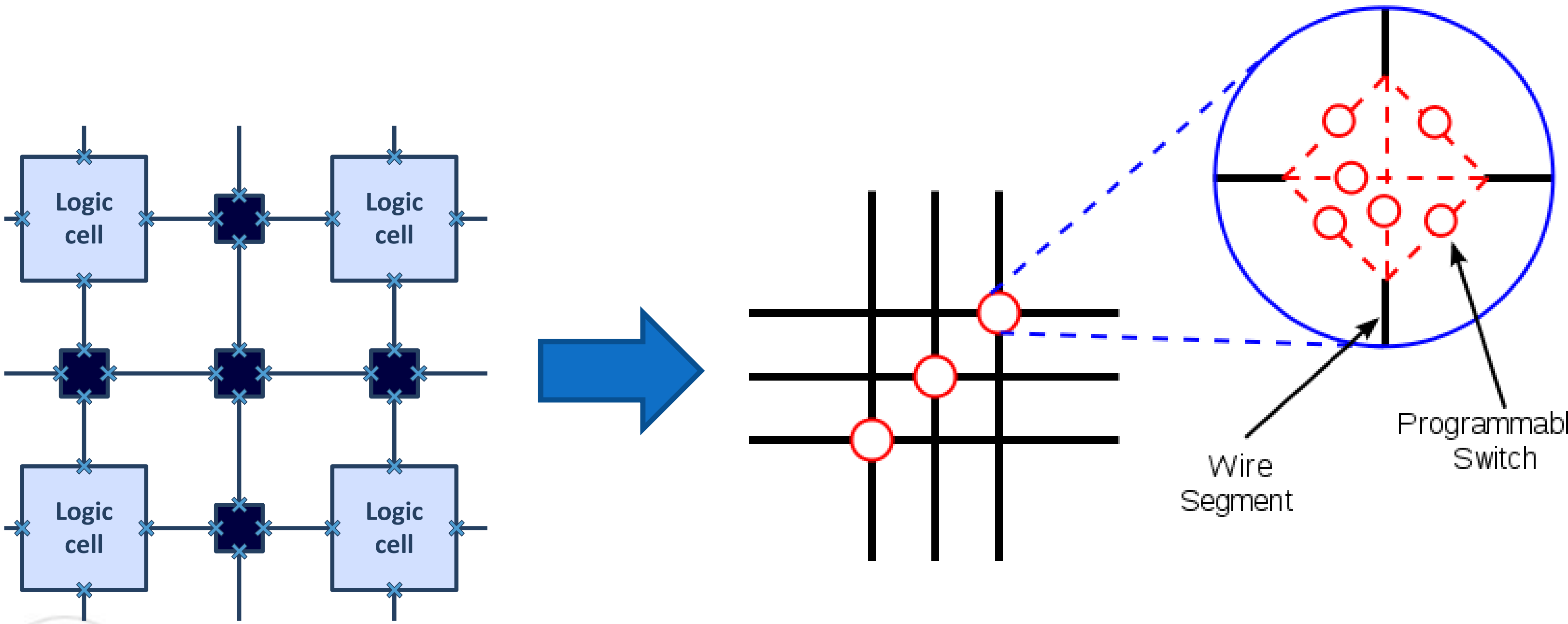
What is an FPGA?

Now in one integrated cirduit



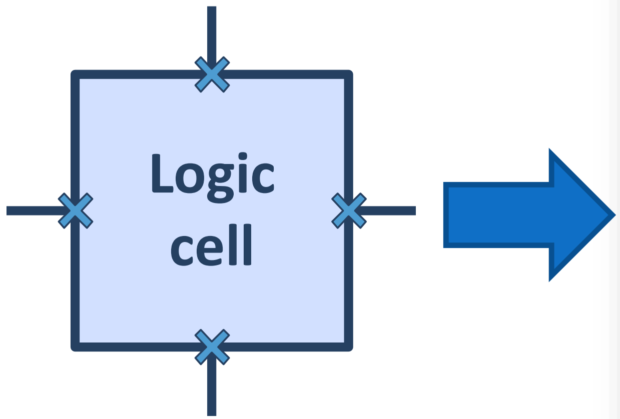
What is an FPGA?

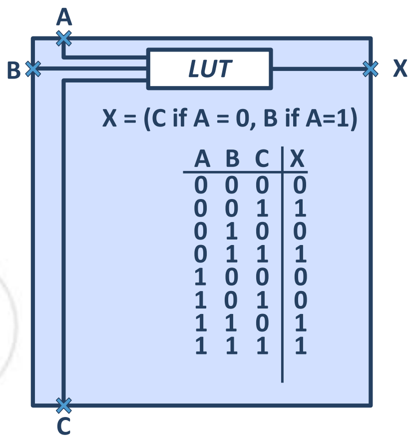
Yeah, I've heard of Integrated Circuits! What's an FPGA!?!?
Cells interconnection – Field programmable Gate Array



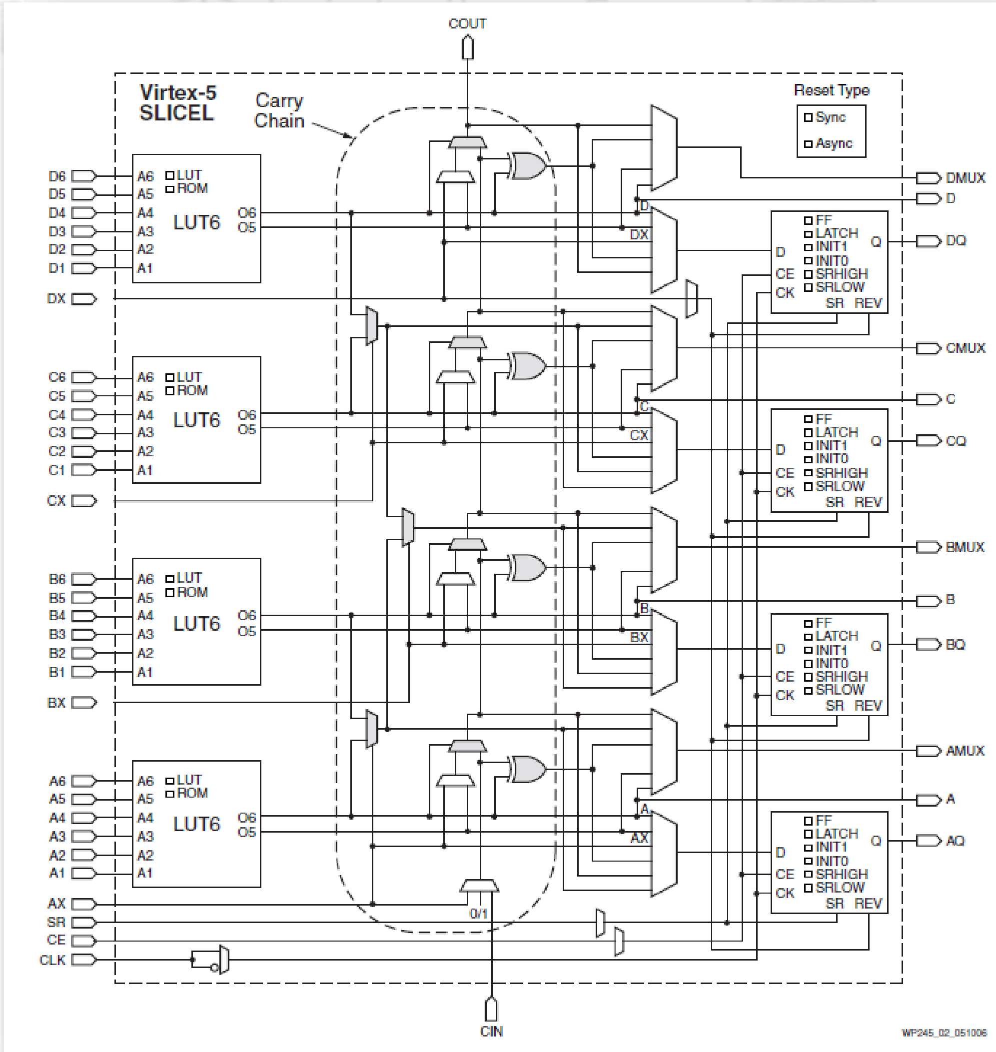
What is an FPGA?

Inside a Virtex 5 Logic Cell



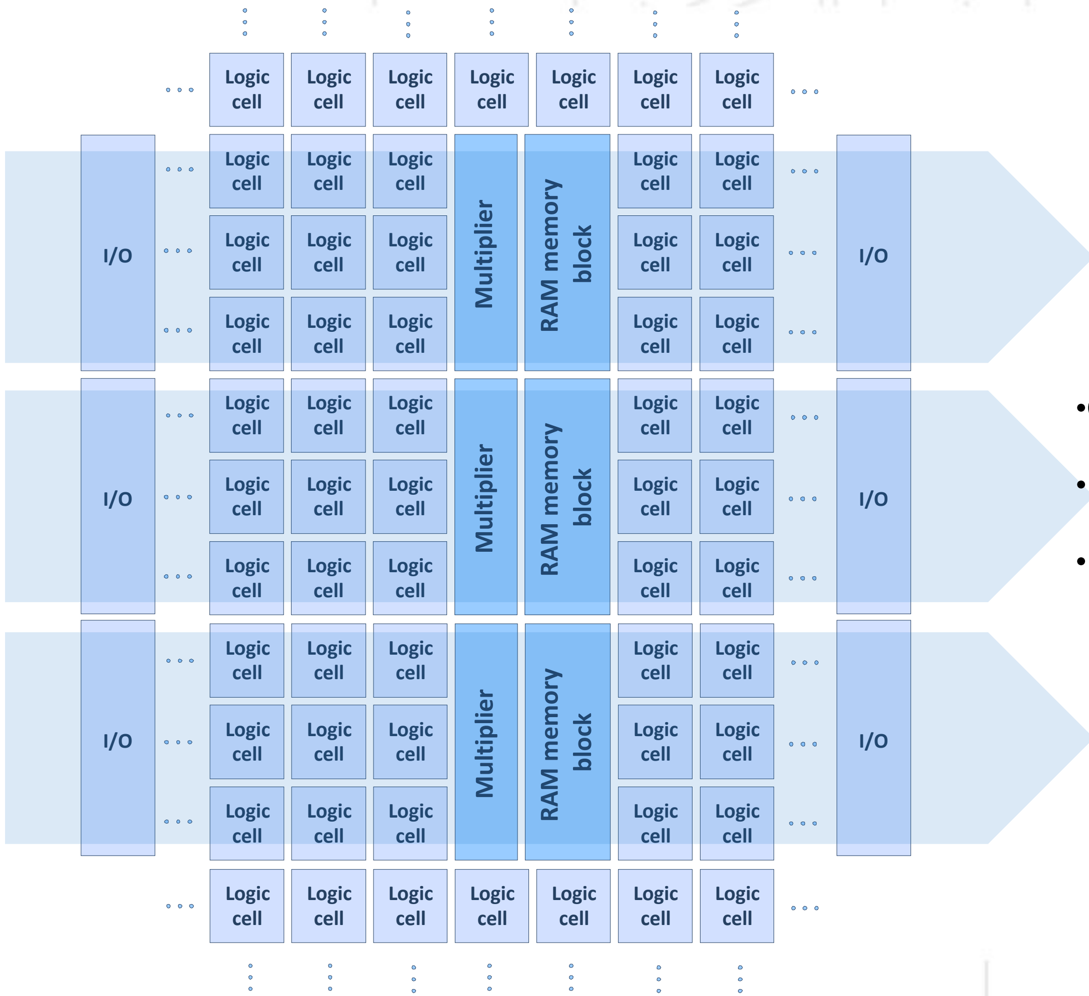


A	B	C	X
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



FPGA for Parallel Computing

- OPAL-RT Analog inputs
- RT-LAB Digital inputs
- RT-LAB CPU Model



- OPAL-RT Analog Outputs
- RT-LAB Digital Outputs
- RT-LAB CPU Model

Constraints

- Limited Ressources
  - Logic Cells
  - Memory
  - LUTs
  - Etc.
- Might not be possible to route the design
- Propagation delays
- Fixed Point calculation

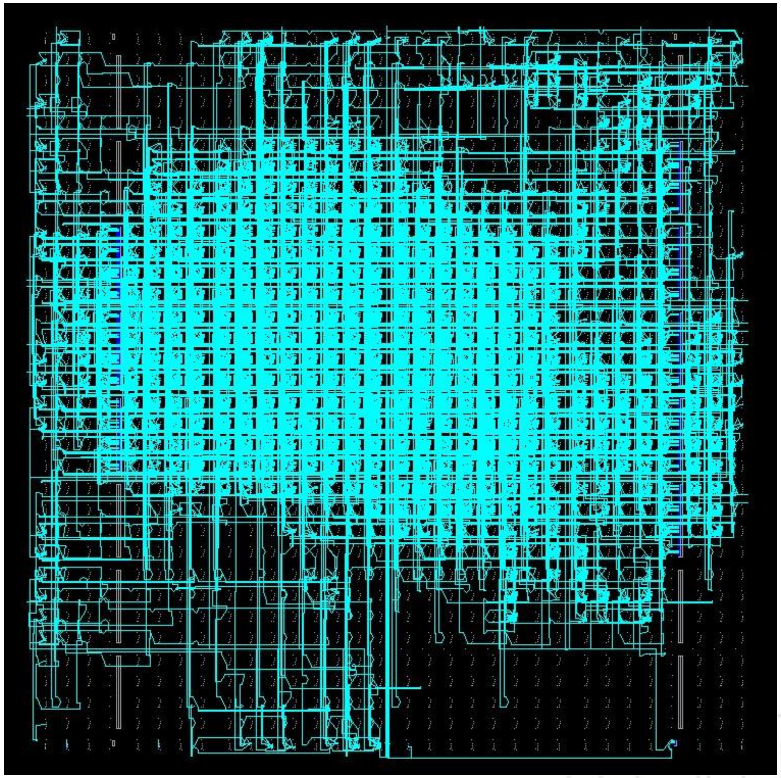
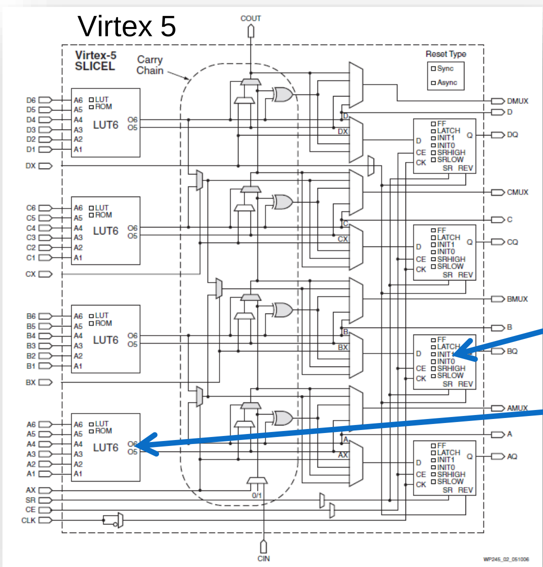


Image Sourcre: http://www.student.uni-kl.de/~alles/fpga/pics/fpga-layout.jpg

FPGAs in Numbers…

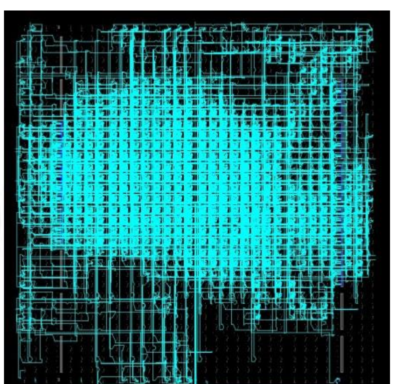


	Virtex II Pro (XC2VP7)	Virtex 5 (XC5VSX50T) ML506	Virtex 6 (XC6VLX240T) ML605	Spartan III (XC3S500)
Arrays (Row x Col)	40x34	120x34	46x34	46x34
Slices (Logic Cells)	4,928	7,200	37,680	4,656
Flip-Flops	9,856	28,800	301,440	9,312
LUTs	9,856	28,800	150,720	9,312
Multipliers	44 (18x18)	48 (25x18)	768 (25x18)	20 (18x18)
Block RAM	792kb	4,752kb	14,976kb	360kb

Recap

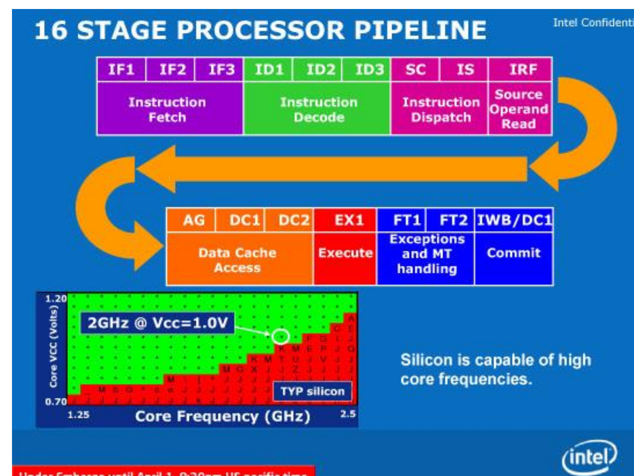
FPGA

- Typically 200 MHz clock
- No instruction, everything “executes” at the same time
- Logic blocks are connected together
- Floating point is more challenging (requires a lot of resources)
- Routing & dealing with delays are challenging



CPU

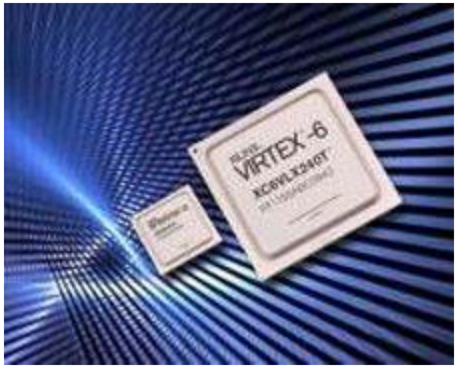
- Typically 3.3 GHz clock
- Operations are executed sequentially
- Floating point engine is embedded inside the chip



Ok! Why FPGA for HIL then?

Advantages compared to CPU-based model processing:

- ✓ Physically near I/O
- ✓ Low latency
- ✓ Parallel signal processing
- ✓ Rapidly improving capacity



Why (not) FPGA ?

For most engineers, FPGAs:

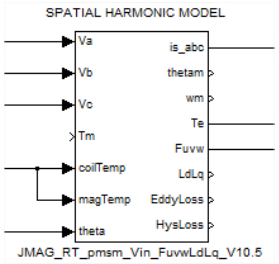
OPAL-RT's answer

- ✓ Are complex to use → HIL turnkey solution
  fixed vs. floating point
- ✓ Lack flexibility → Generic approach to FPGA
- ✓ Have low fidelity → Implementation of ANSYS Maxwell FEA motor models

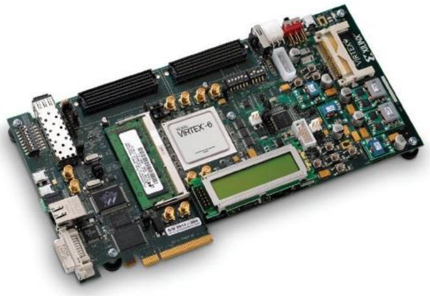
PMSM Solver on FPGA

OPAL-RT latest developments:

- ✓ CPU equivalent Step Time : 5-20us
- ✓ Model total Latency: 15-40us



- ✓ Step Time : 100-450ns
- ✓ Model total Latency: Below 2.5us



OPAL-RT PMSM FPGA Solver

High fidelity modeling + High speed I/Os

PMSM Solver on FPGA

OPAL-RT latest developments:

- ✓ Upgrade of motor solver to latest FEA software levels
  - ➡ Solver compatible with PMSM spatial harmonics and VarDQ approach
- ✓ Streamlining of integration steps
  - ➡ Solver configuration ready in a few clicks with online reconfiguration of I/O mapping
- ✓ Improvement of solver accuracy
  - ➡ Model entirely computed in floating point

PMSM Solver on FPGA

Implementation:

✓ Export Netlist from ANSYS Maxwell Software

➡ Precalculation of multiple operating points

✓ Import Netlist into RT-LAB environment

➡ Use RT-LAB to build your realtime simulation

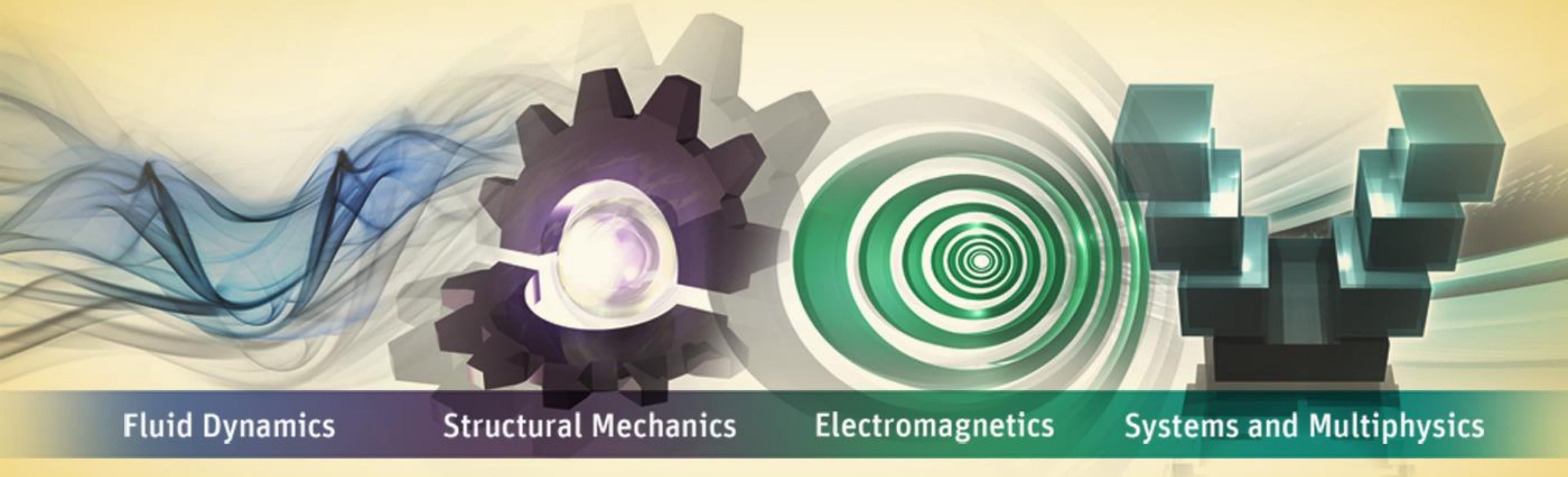
✓ Ready-for-Realtime

➡ Integrate I/O and any other required application peripherals

ANSYS

Realize Your Product Promise™

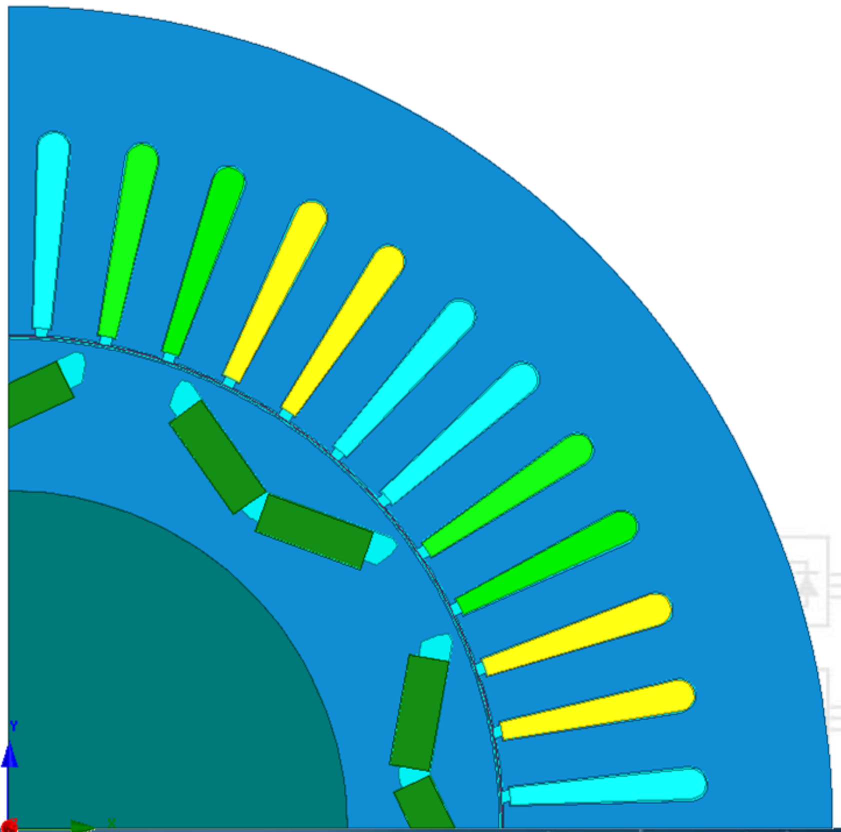
OPAL-RT **Benchmark Simulation**



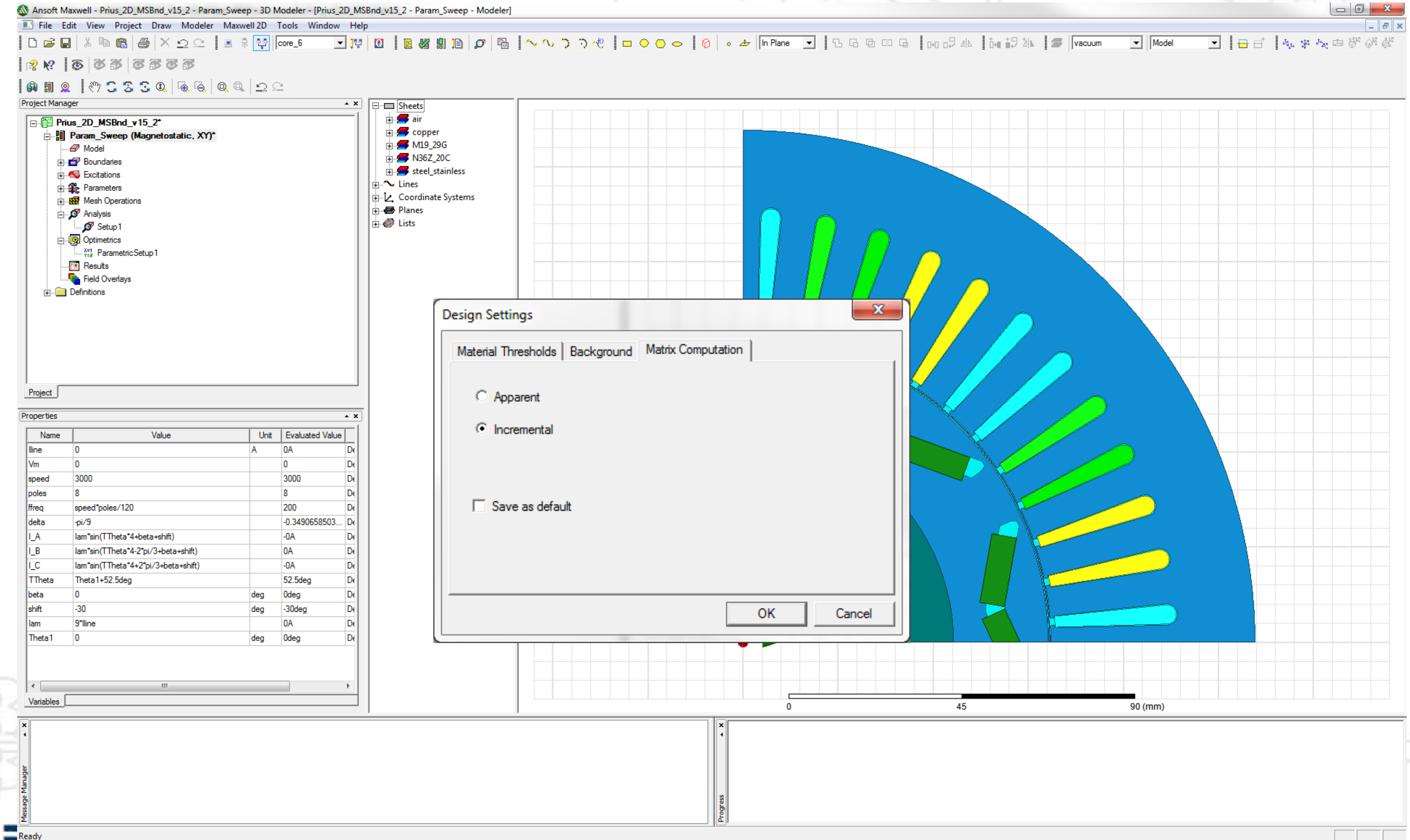
Maxwell 16.0

IPM Motor Simulations on LS-DSO

- **OS Linux cluster specifications:**
  - **Total CPUs (cores): 48**
  - **Total hosts (nodes): 4**
- **Large Scale Distributed Solve Option (LS-DSO)**
- **Prius motor project**
- **17195 variations**
- **Cores used 48**



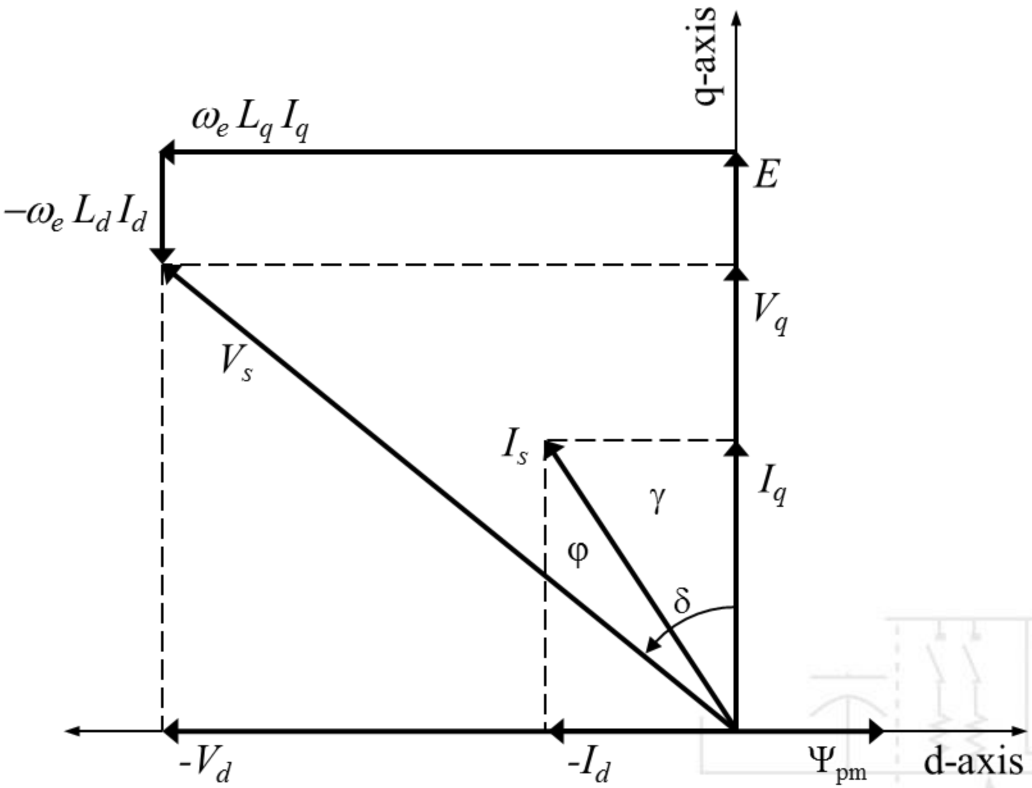
Maxwell Setup



O P A L · R T

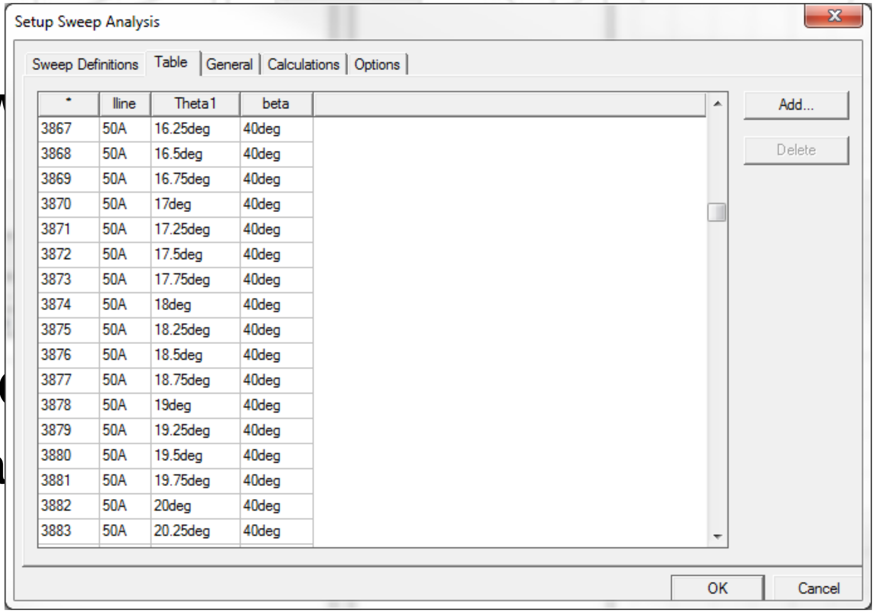
Maxwell Setup

- **Alignment of the initial rotor position is done**
- **Flux linkage is maximum when theta = 0 and Iamp =0**



IPM Motor Simulations on LS-DSO

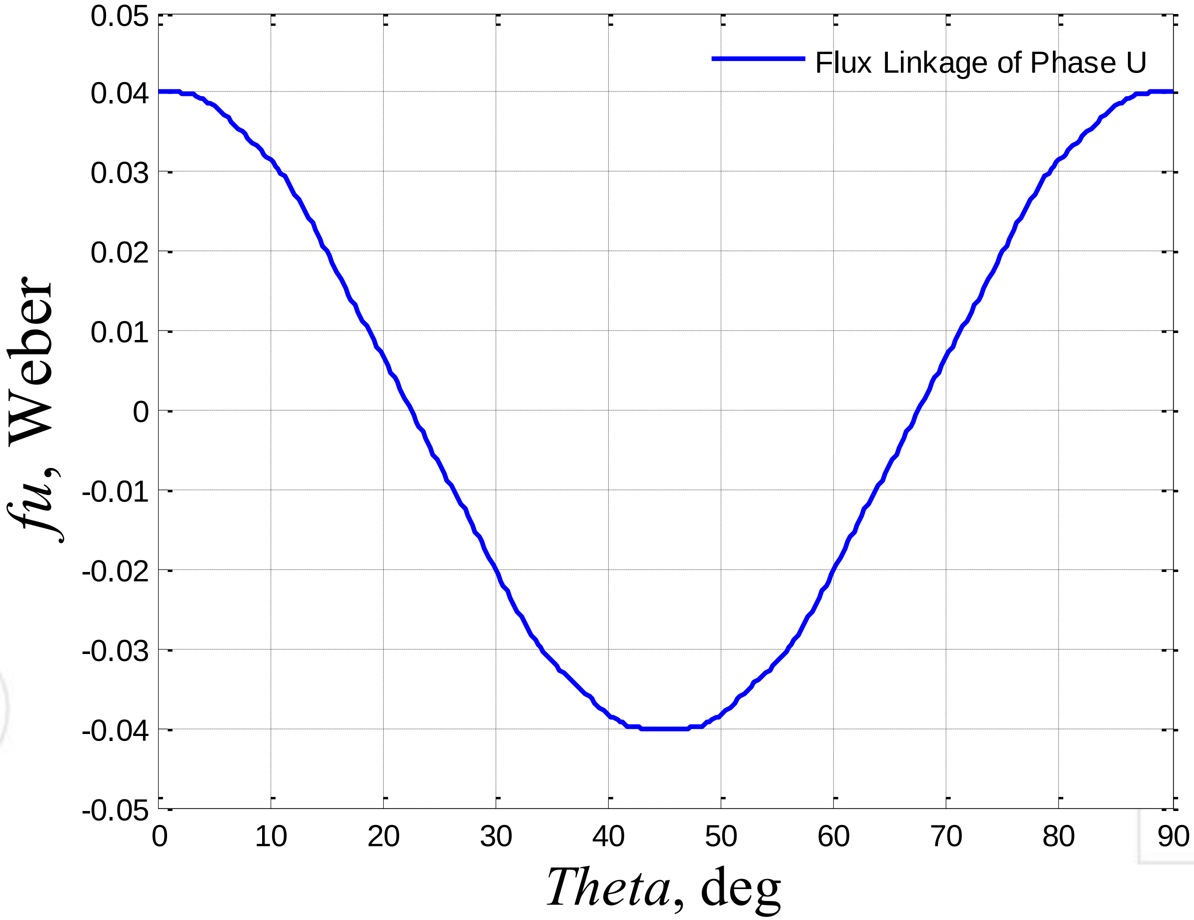
- **Parametric sweep table of 17195 row**
  - **Beta = 0:20:360**
  - **Theta = 0:0.25:45**
  - **Iamp = 0:50:200**
- **Parametric table was run on (LS-DS**
- **Results post-processed using Matla**
- **Final Table:**
  - **Beta = 0:5:360**
  - **Theta = 0:0.25:45**
  - **Iamp =** [0,2.5,5,8,11,18,25,37.50,50,75,100,125,150,175,20]
- **Note: Results were post-processed using spline interpolation in Matlab**



*	Iline	Theta1	beta
3867	50A	16.25deg	40deg
3868	50A	16.5deg	40deg
3869	50A	16.75deg	40deg
3870	50A	17deg	40deg
3871	50A	17.25deg	40deg
3872	50A	17.5deg	40deg
3873	50A	17.75deg	40deg
3874	50A	18deg	40deg
3875	50A	18.25deg	40deg
3876	50A	18.5deg	40deg
3877	50A	18.75deg	40deg
3878	50A	19deg	40deg
3879	50A	19.25deg	40deg
3880	50A	19.5deg	40deg
3881	50A	19.75deg	40deg
3882	50A	20deg	40deg
3883	50A	20.25deg	40deg

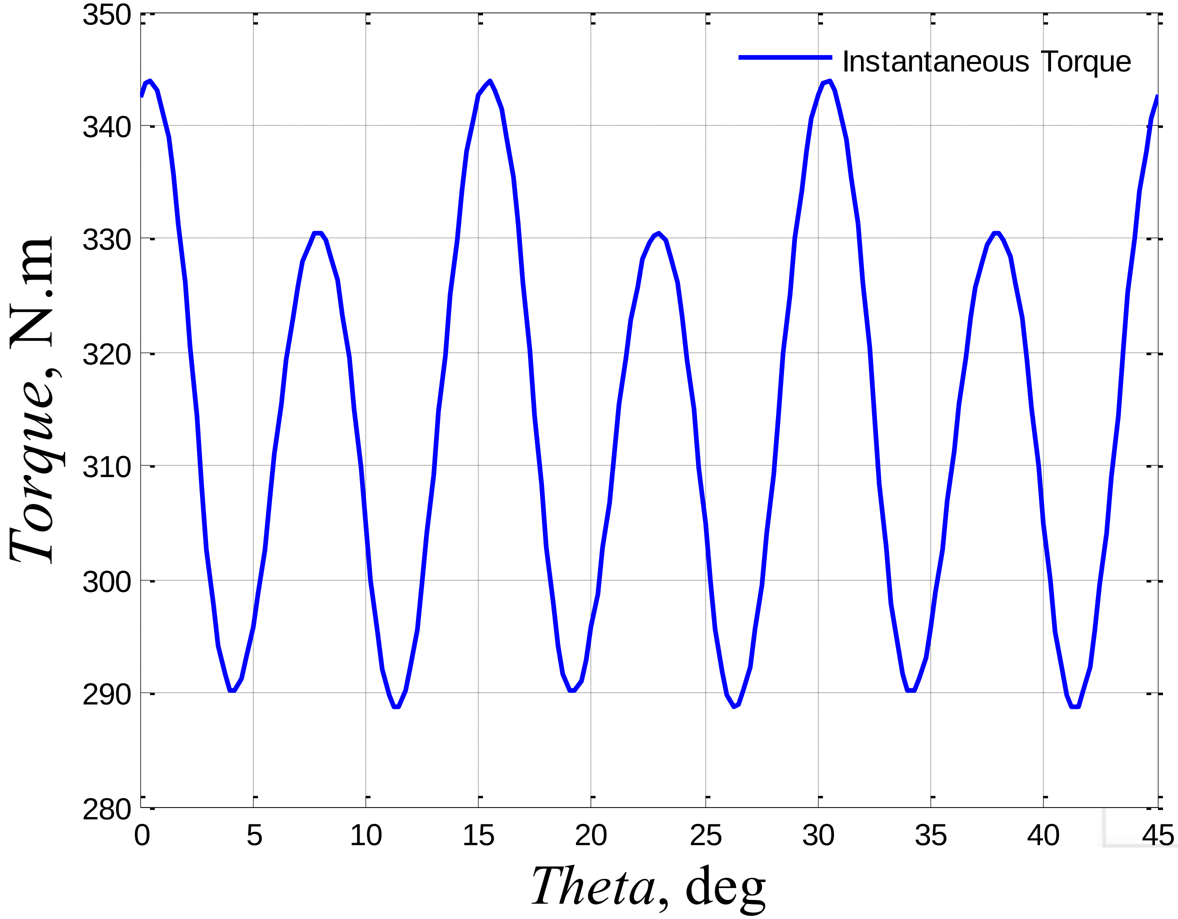
Results – Flux Linkage

- **Flux linkage of phase U when Beta = 0 deg and Iamp = 0A**



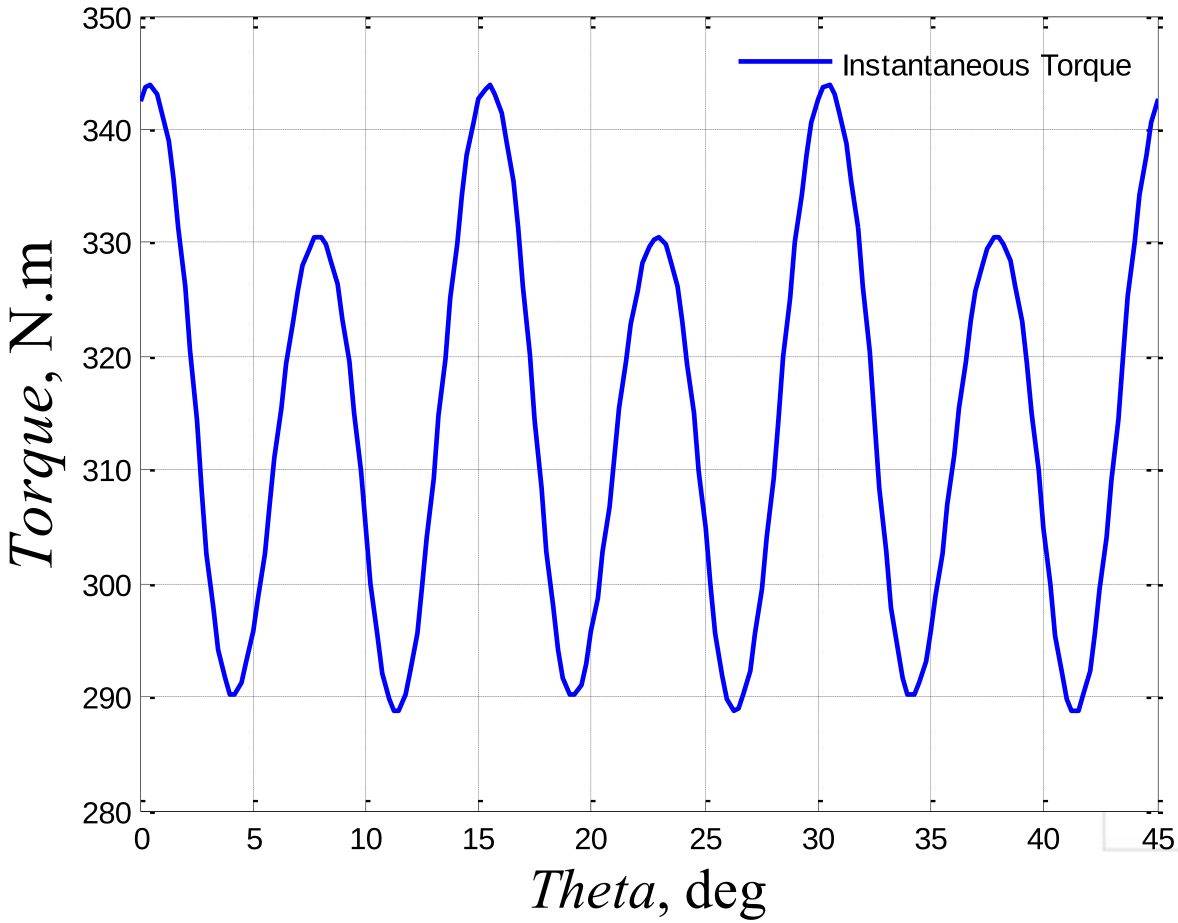
Results – Instantaneous Torque

- **Instantaneous torque of phase U when Beta = 45 deg and Iamp = 200A**



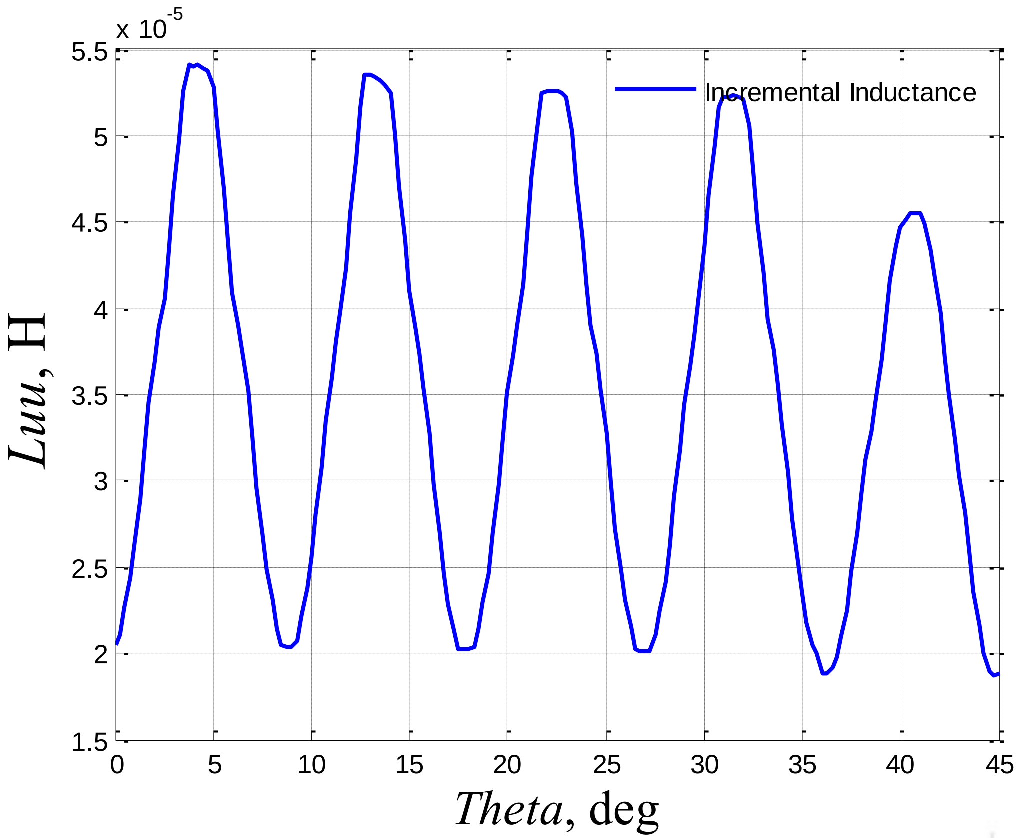
Results – Instantaneous Torque

- **Instantaneous torque of phase U when Beta = 45 deg and Iamp = 200A**



Results – Incremental Inductance

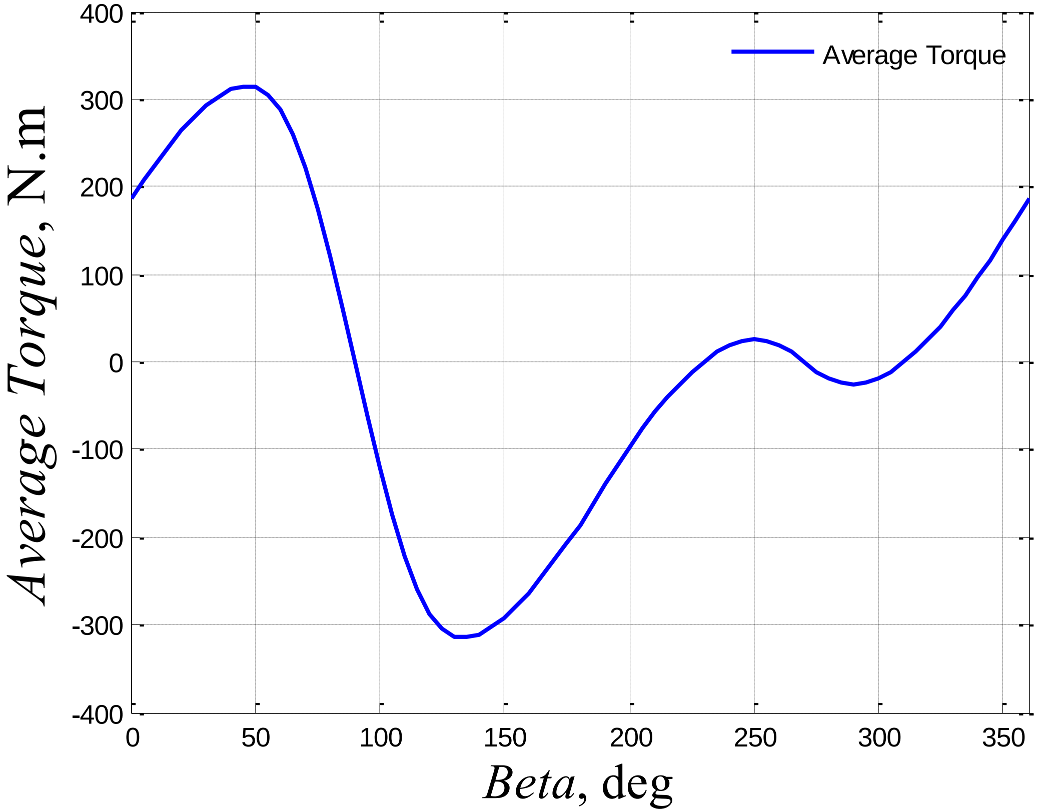
- **Incremental inductance of phase U when Beta = 45 deg and Iamp = 200A**



- **Note: Maxwell also can compute the incremental inductance when Iamp = 0**

Results – Average Torque

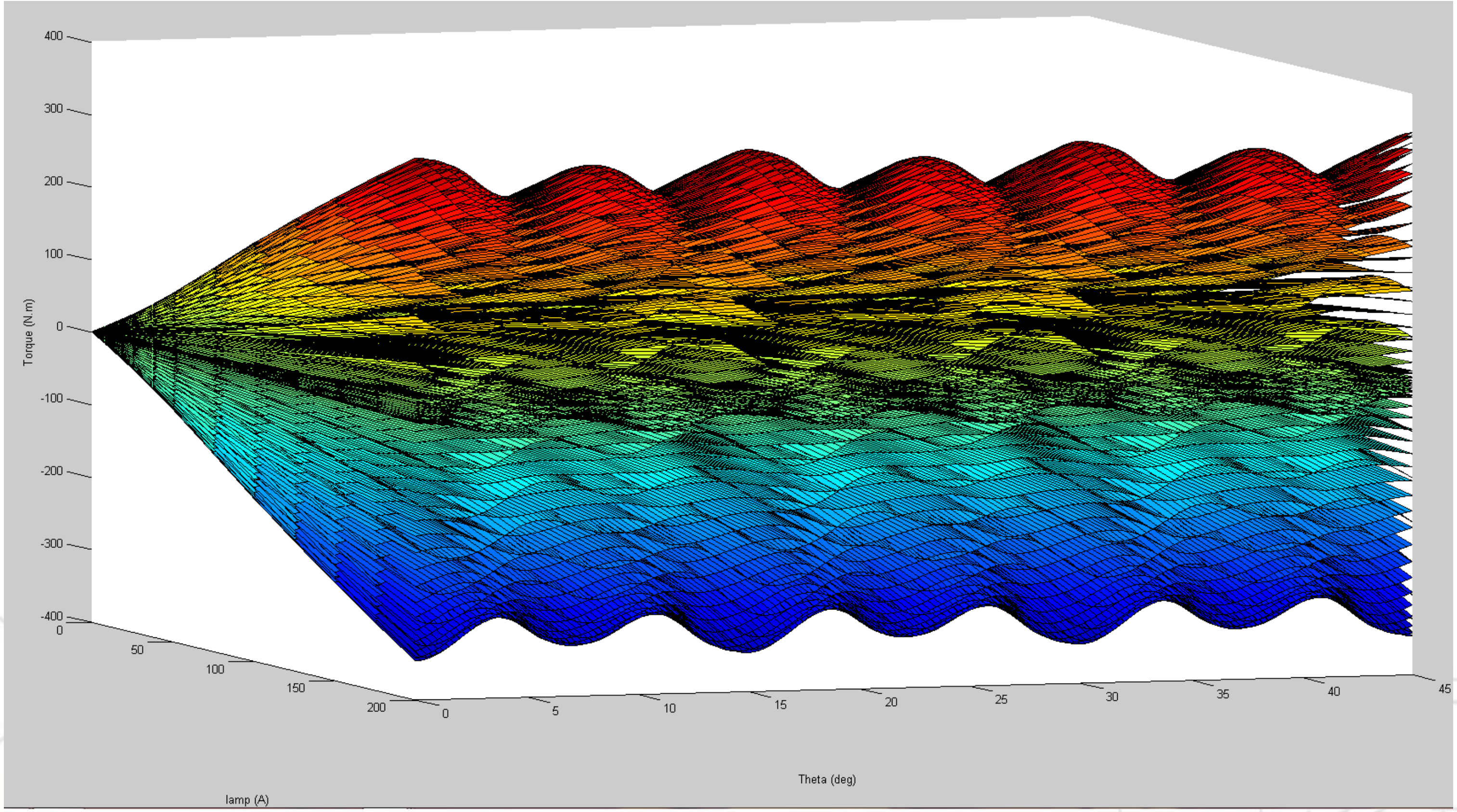
- **Average torque of phase U when Beta = 45 deg and Iamp = 200A**



- **Note: For motor mode operation, Beta ranges from 0 deg to 90 deg which adheres to the alignment criteria shown in the phasor diagram**

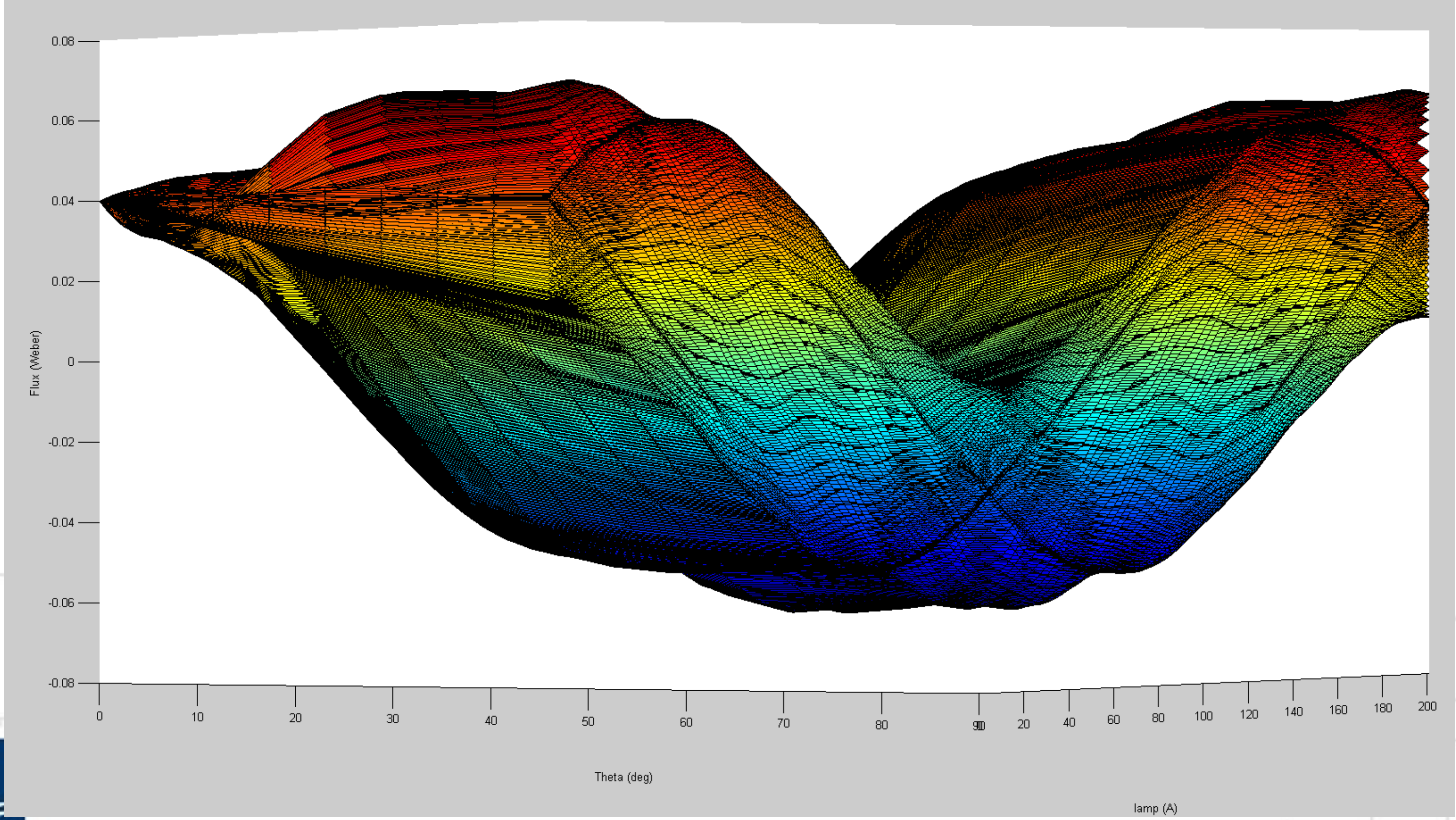
Results – Instantaneous Torque

- **Instantaneous Torque**



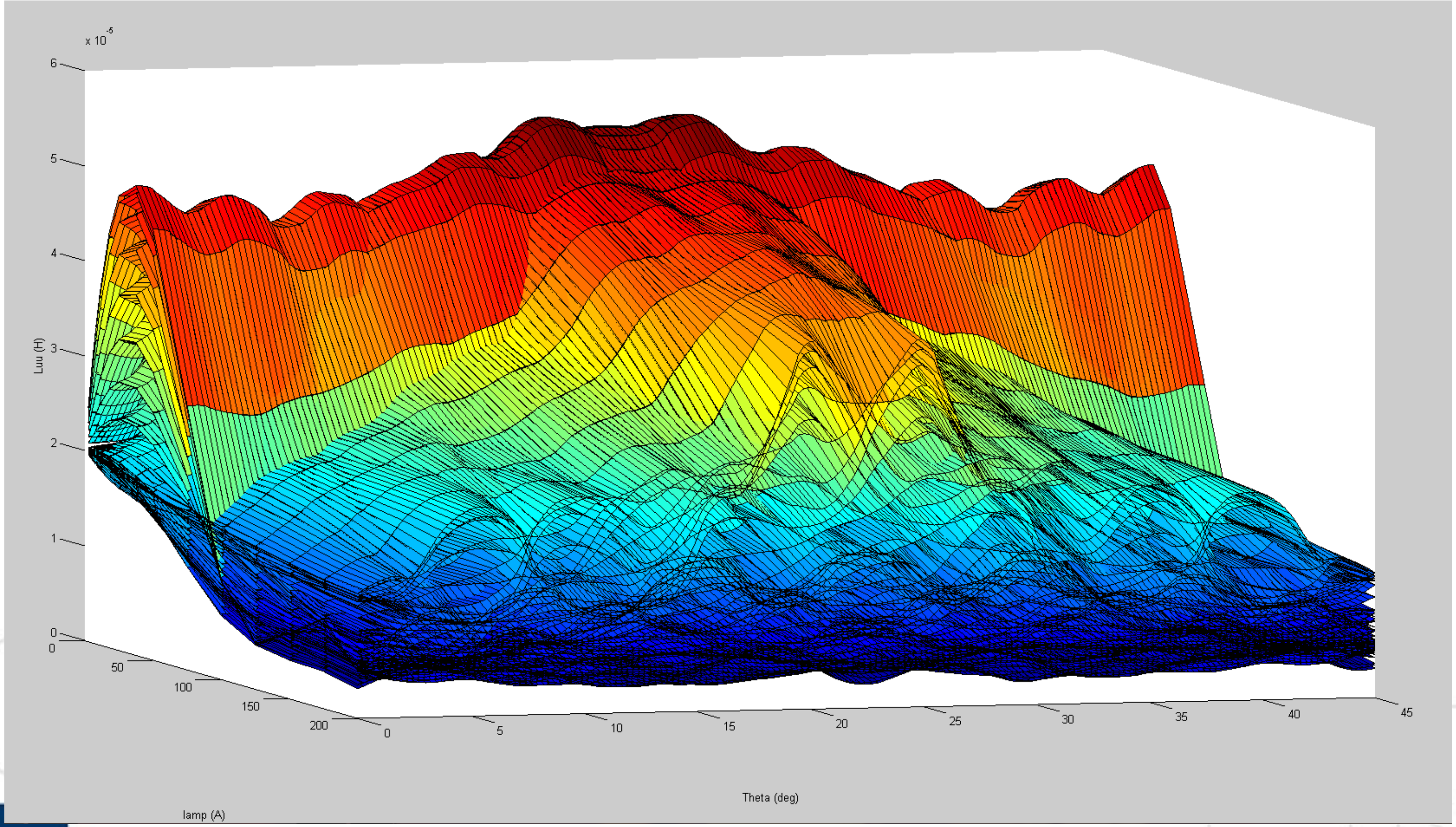
Results – Instantaneous Flux Linkage

- **Instantaneous Flux Linkage**



Results – Instantaneous Flux Linkage

- **Instantaneous Flux Linkage**



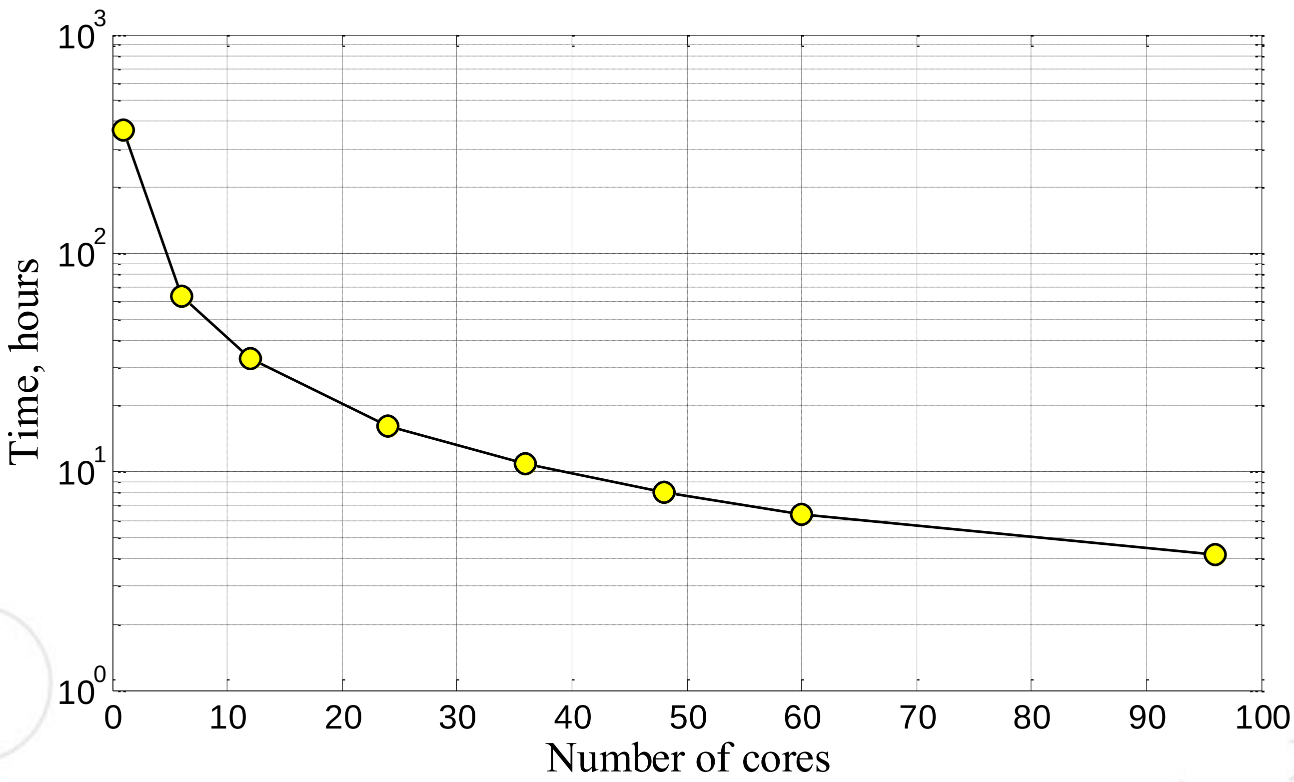
IPM Motor Simulations on LS-DSO

- **Speed-up factor and cores utilizations**:

Number of cores	Simulation time (hours)	Speed-up factor	Cores utilization %
1	368.3	1	100%
6	63.7	5.7	95%
12	32.9	11.2	94%
24	16.2	22.8	95%
36	10.9	33.4	94%
48	8.0	46.1	96%
60	6.4	57.6	96%
96	4.2	90.0	94%

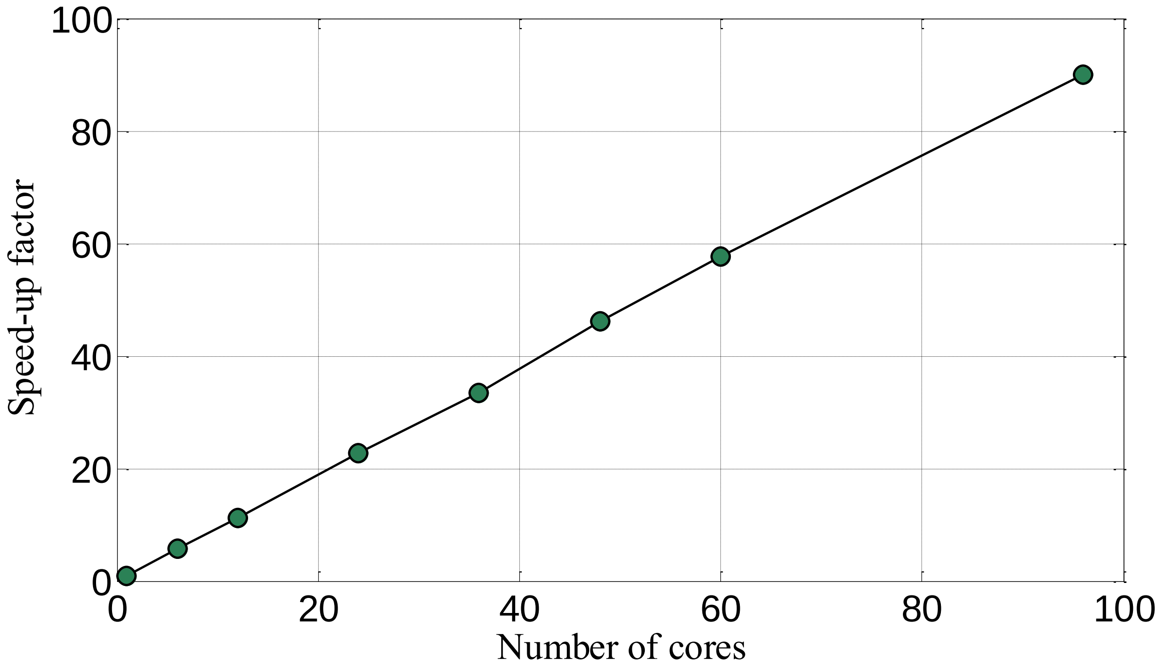
IPM Motor Simulations on LS-DSO

- **Simulation time in *log scale***:



IPM Motor Simulations on LS-DSO

- **Speed-up factor**:

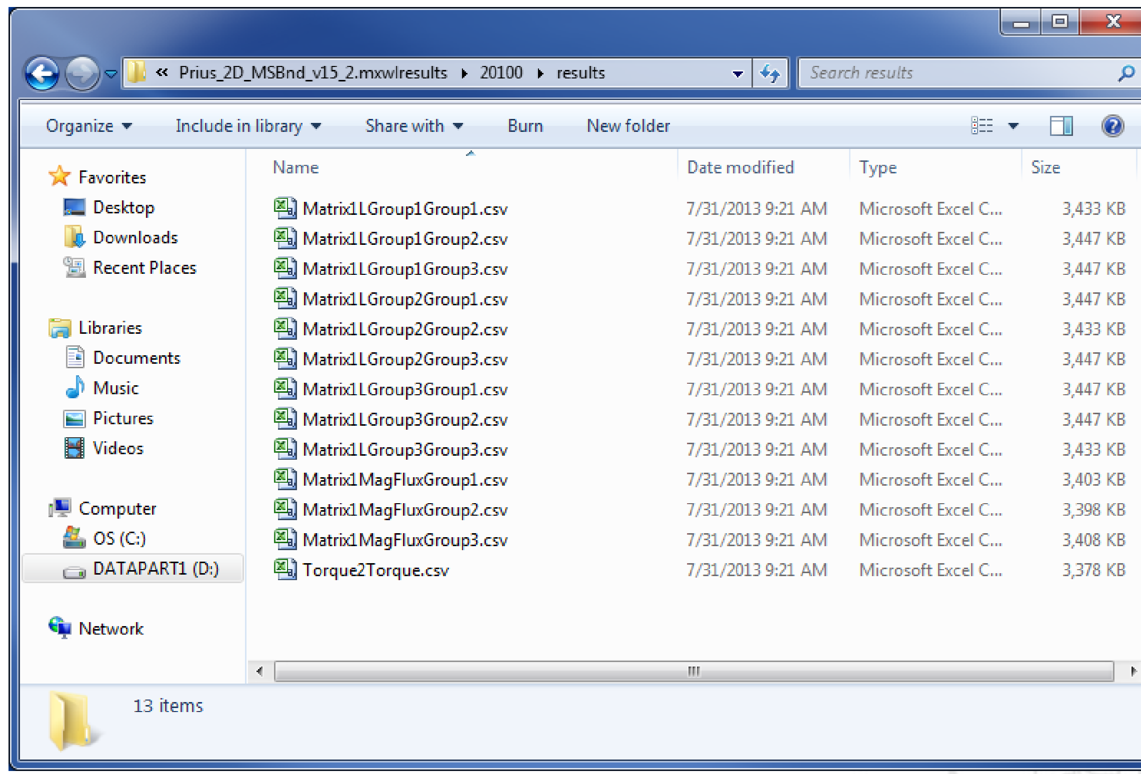


This graph shows that the simulation time is reduced linearly with the increase of number of cores

Extraction of Results on LS-DSO

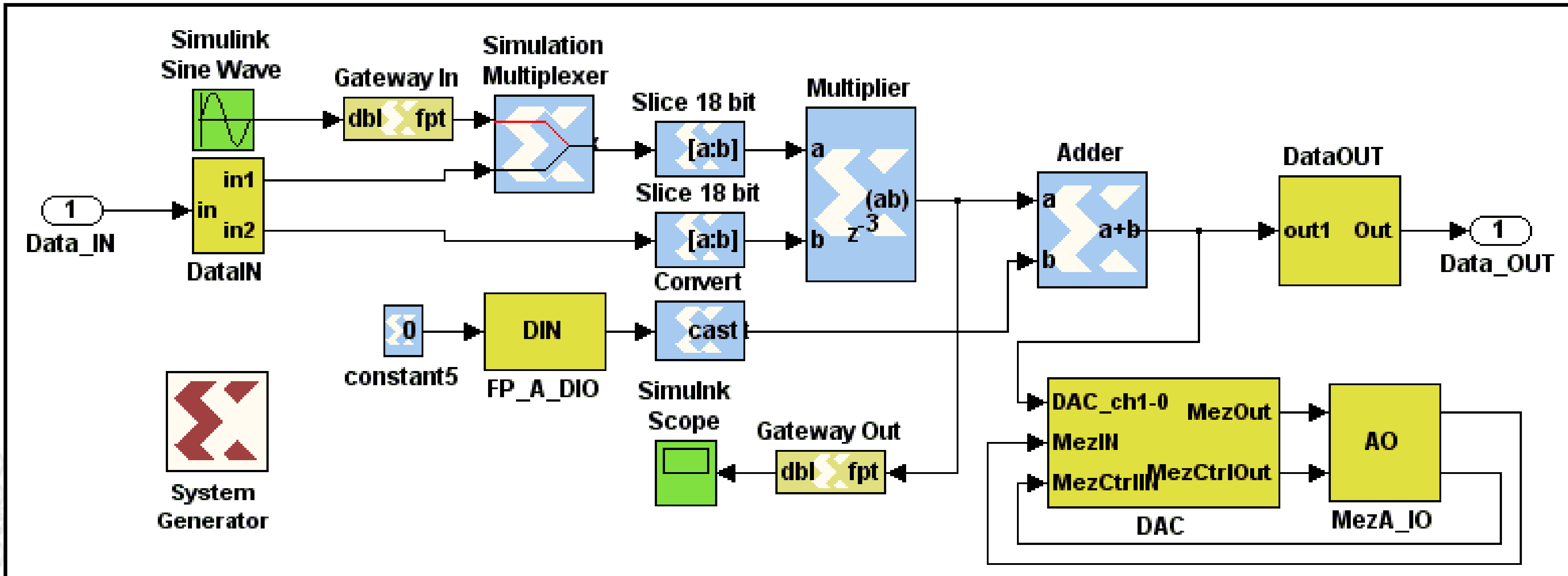
- **Results for all variations extracted using LSDSO extractor with –mergecsv option**

All results of the variations are combined in a single CSV file



RT-LAB I/Os are fully programmable with Xilinx System Generator

- Plant model exported from Maxwell, is integrated with I/O & any peripheral plant model components in Simulink to be compiled for real-time.
- Xilinx System Generator is a FPGA Simulink blockset
  - No need to know VHDL language
- User can customize the I/O for complex applications



A typical XSG model in RT-LAB

Conclusion

- ✓ FPGA will soon be the reference for HIL testing

- ✓ High-fidelity HIL model on FPGA is a reality

Large scale parametric analysis of (example) Prius Motor was done to prepare data for OPAL-RT software using ANSYS Maxwell software

- ✓ Motor prototyping is ready

- ✓ Enhanced control algorithm validation is now possible on HIL

- ✓ Faster test means lower cost

- ✓ Motor and controller designer can work closely together – The exported Maxwell model (Design) IS the HIL plant model

OPAL-RT