FPGA Implementation of Model Predictive Control using High Level Synthesis

Minh Nguyen, Colin Knizek, Cecelia Pham

Overview

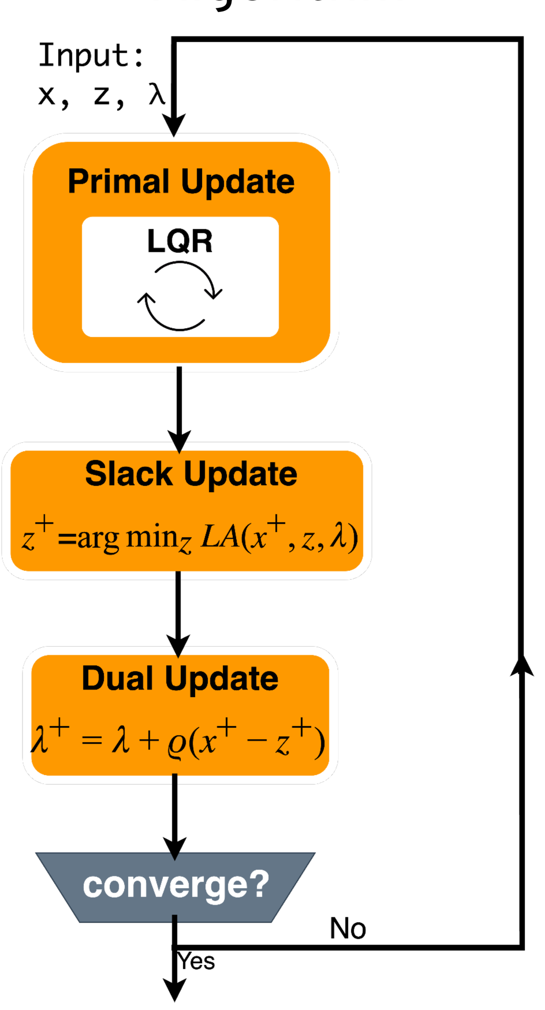
We took an existing MPC algorithm (TinyMPC) and adapted it to HDL using High Level Synthesis (HLS). We then deployed the algorithm on two common FPGAs and tested the performance on a simulated end to end workload

Motivation

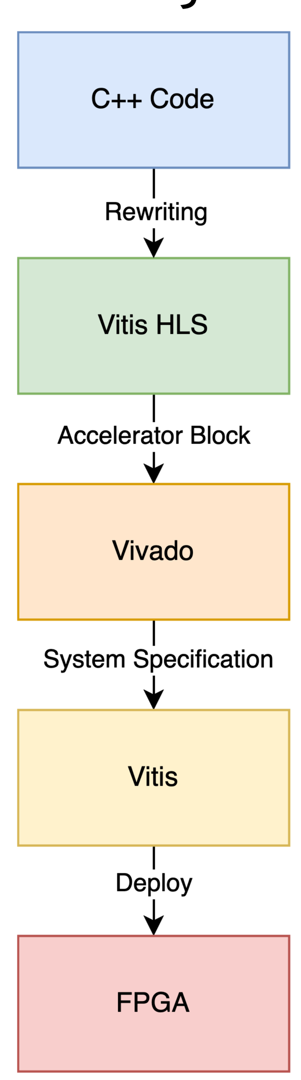
- Resource constrained robotic platforms have difficulty running online control algorithms
- FPGAs offer high performance and flexibility but are difficult to program

Background

TinyMPC ADMM Algorithm



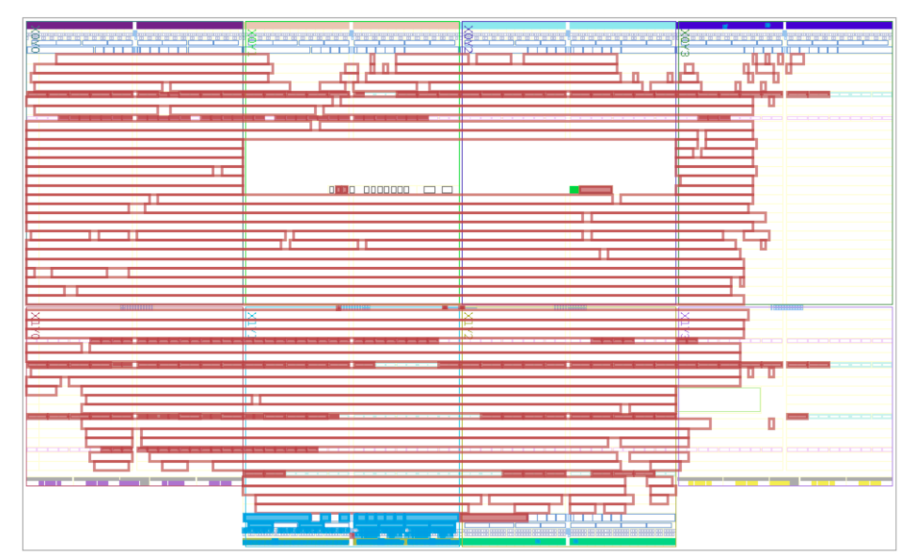
HLS Design Flow

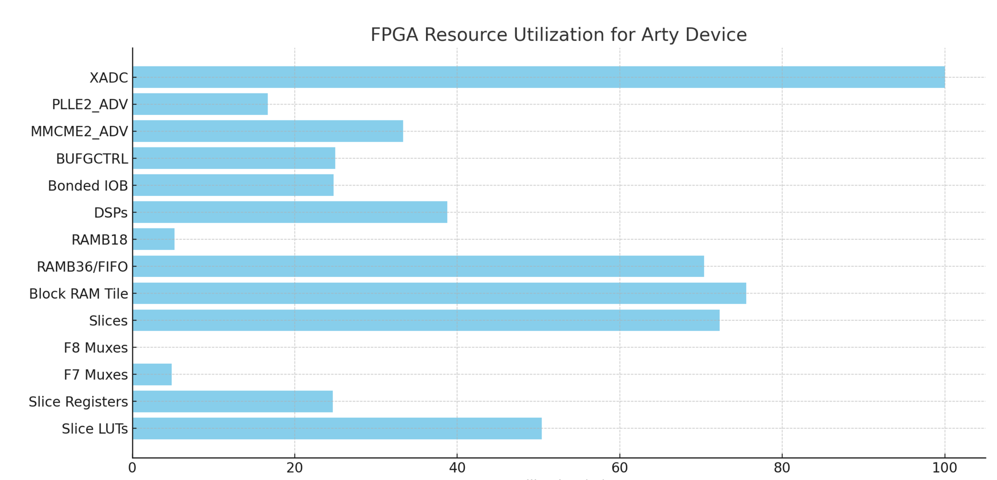


- MPC is a control algorithm that uses a robot's dynamics to calculate an optimal trajectory
- MPC is computationally intensive so want to use custom hardware for speedups
- HLS turns C code to Verilog with appropriate pipelining and parallelism
- Use HLS to take annotated MPC code and turns it into an accelerator block to be incorporated into an SOC

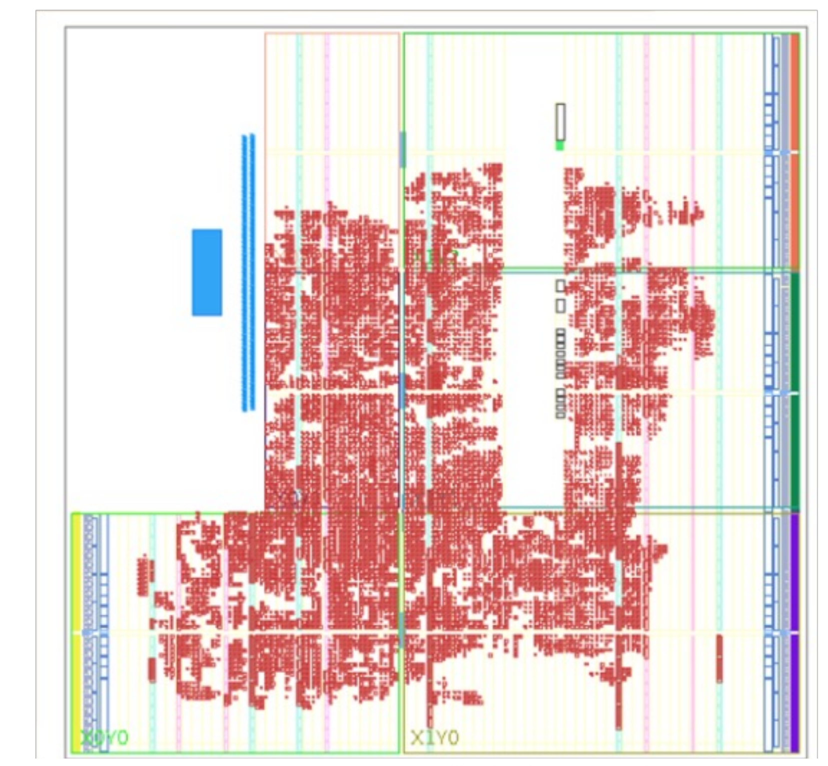
Implementation

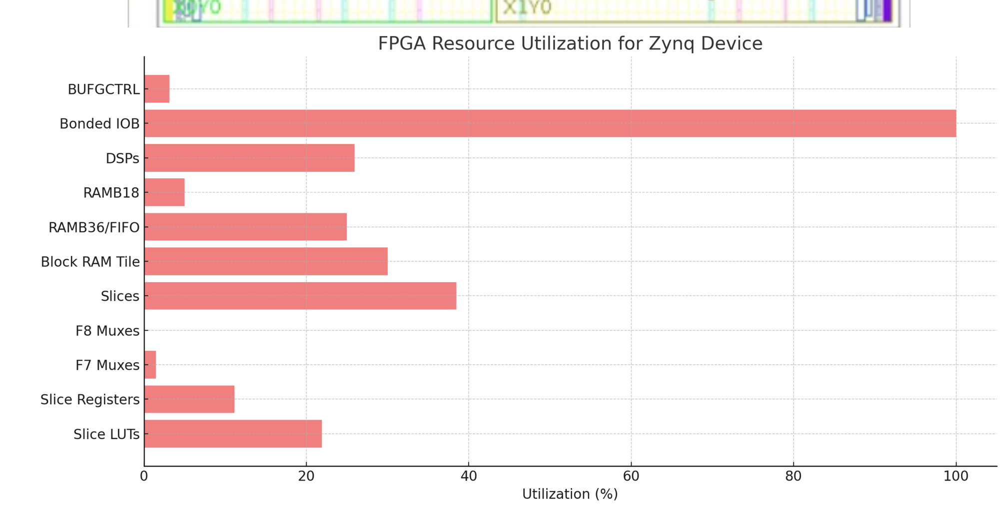
Arty 100T



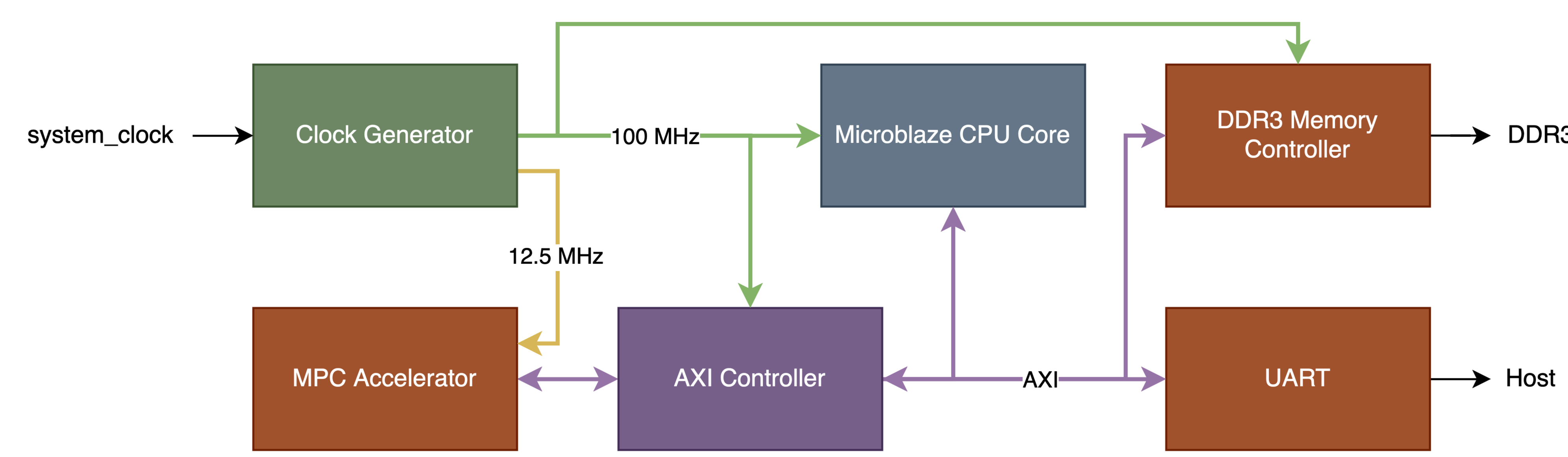


ZYNQ 7 Series

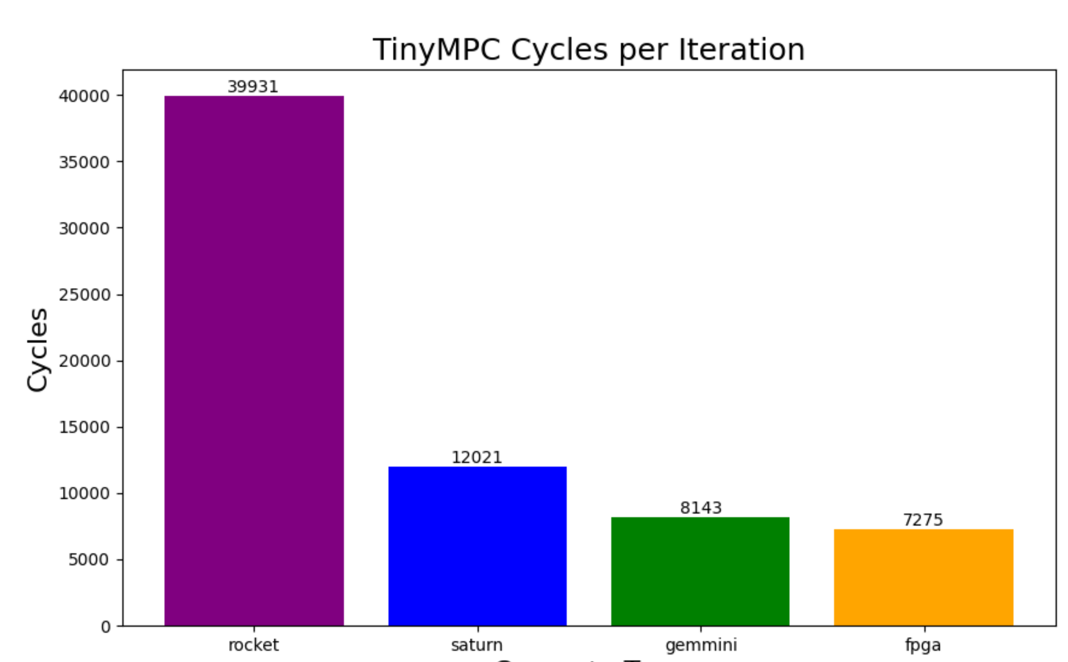


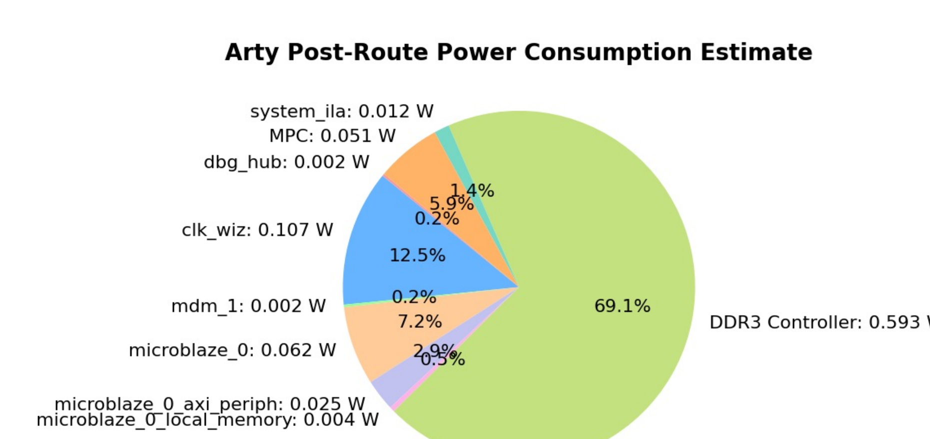


System Block Diagram



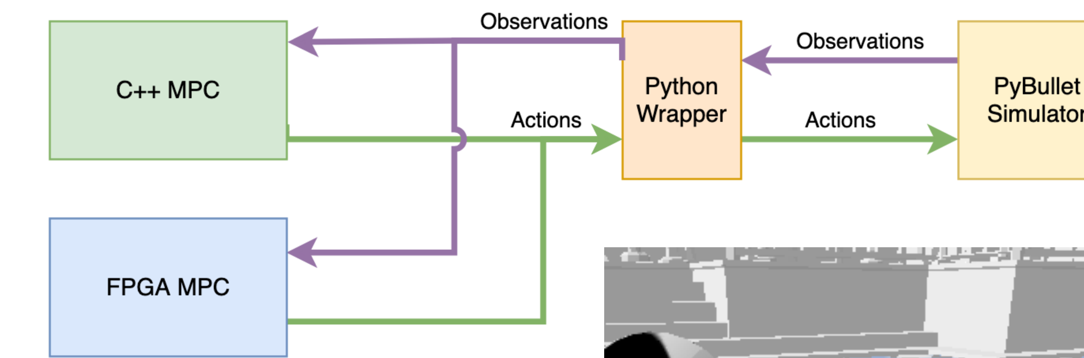
Performance Evaluation



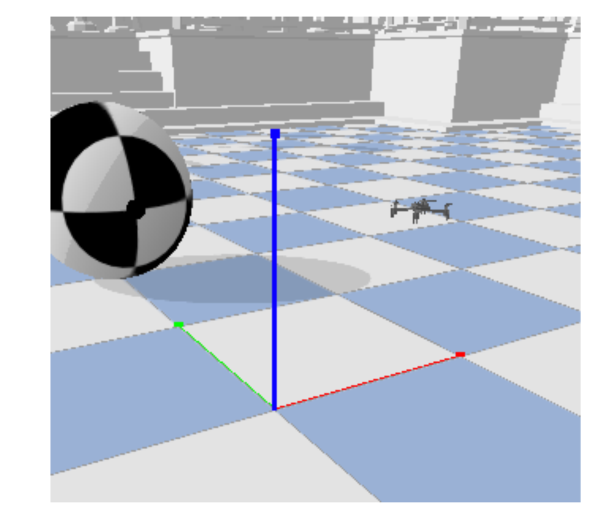


MPC logic takes about 6% of power (0.051W)

Simulation



- ***Gym-Pybullet-Drones*** simulator provides a decoupled interface for the FPGA to interact with



UART Communication Delay

Profiler stopped. Press 'Start' button to restart profiling

Address	% Exclu	% Inclus	Function	File
80018200	19.4		XTracking_IsDone	xtracking.c
80018e8c	67.6		XUartLite_RecvByte	xuartlite_l.c
80018e6c	9.74		XUartLite_SendByte	xuartlite_l.c
80016480	.153		__divdf3	
80016e70	.077		__mulsi3	
80005528	.077		__srefill_r	
80015c28	.077		_fpadd_parts	
80010234	.077		_read_r	
80007578	.077		_strtod_r	
80017e44	2.60		tracking	helloworld.c

Conclusion

- Successfully deployed FPGA implementation of MPC with UART communication
- 5.49x speedup over Rocket and 1.2x speedup over Gemmini with no hand tuning

Future Work

- Hand optimization of current code with HLS pragmas to increase parallelism/performance
- Add on to existing algorithm with nonlinear control
- Use research work in parallelized LQR to fully utilize FPGA parallelism

Contact Information

Minh Nguyen	minh02@berkeley.edu
Colin Knizek	knizek@berkeley.edu
Cecelia Pham	ceciliapham@berkeley.edu

May 2024