Karlsruhe Institute of Technology

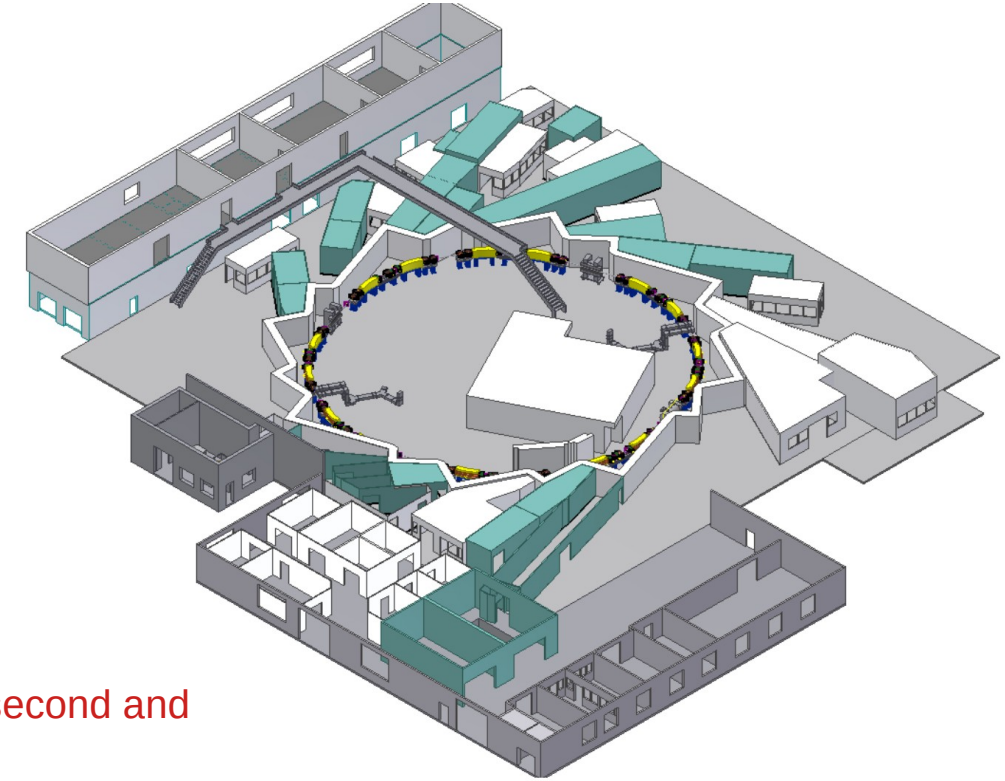
HELMHOLTZ
RESEARCH FOR GRAND CHALLENGES

A Low-Latency Feedback System for Control of Horizontal Betatron Oscillations of an Electron Beam

Luca Scomparin, Edmund Blomley, Michele Caselle, Andrea Santamaria Garcia, Johannes Steinmann, Andreas Kopmann, Erik Bruendermann, Anke-Susanne Mueller, Marc Weber

Karlsruhe Research Accelerator (KARA)

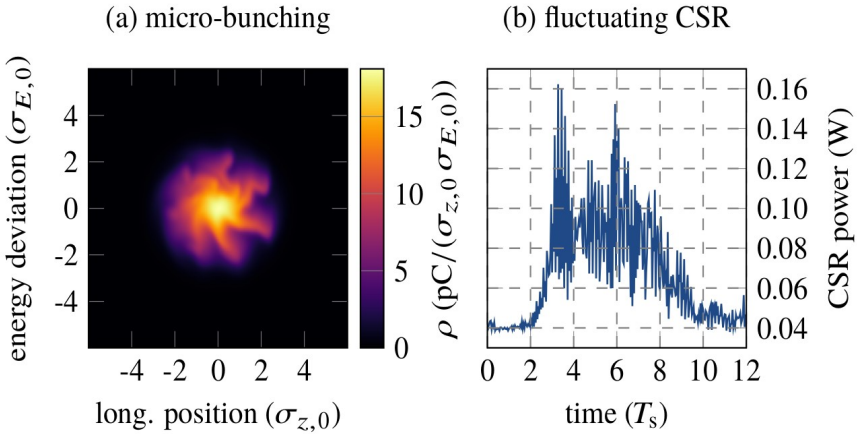
- Synchrotron light source with insertion devices
- Electron storage ring
- 2.5 GeV maximum energy @ 150 mA
- 110 m circumference
- RF frequency of ~500 MHz with 184 buckets
- Bunch length in the few ps range

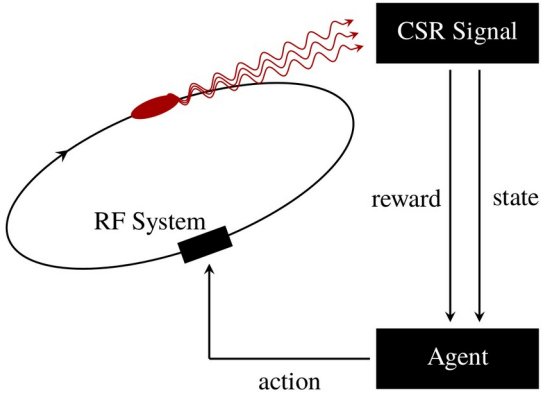


Detector and DAQ need 500M Events per second and ps resolution

Control of Longitudinal Beam dynamics at KARA

Interaction of beam with emitted radiation creates **instabilities** making **power fluctuate** (micro-bunching instability) → **limits user operation**





Control loop to limit this effect, **but** control problem not solved → can we used Reinforcement Learning methods?
In RL an agent learns to maximise the cumulative reward by interacting with an environment
Yes, but inference must be at dynamics timescale O(tens of µs)

Specialized hardware is needed

Xilinx Versal ACAPs

Adaptive Compute Acceleration Platform (ACAP)

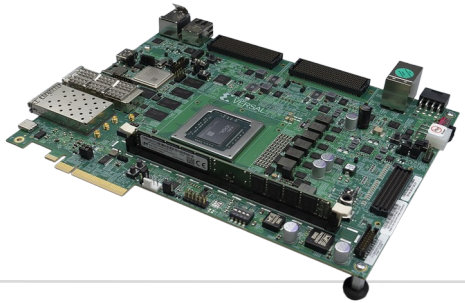
Which combines:

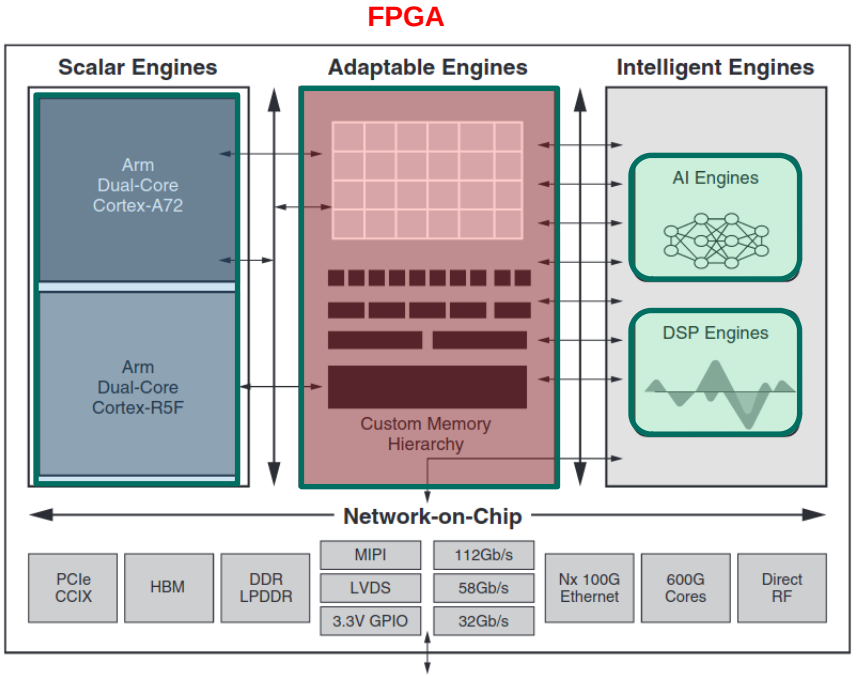
- ARM processors (Scalar Engines)
- FPGA (Adaptable Engines)
- AI Engines
- DSP Engines
- Advance interfaces

All linked by high bandwidth Network on a Chip (NoC)

Allows full customization of the dataflow depending on the application

VCK190 Evaluation Kit



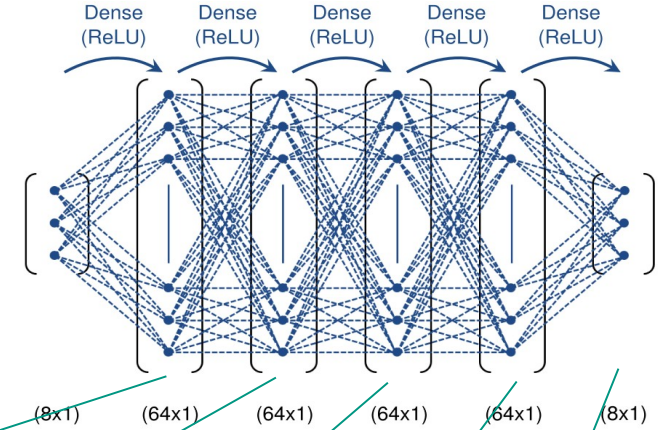


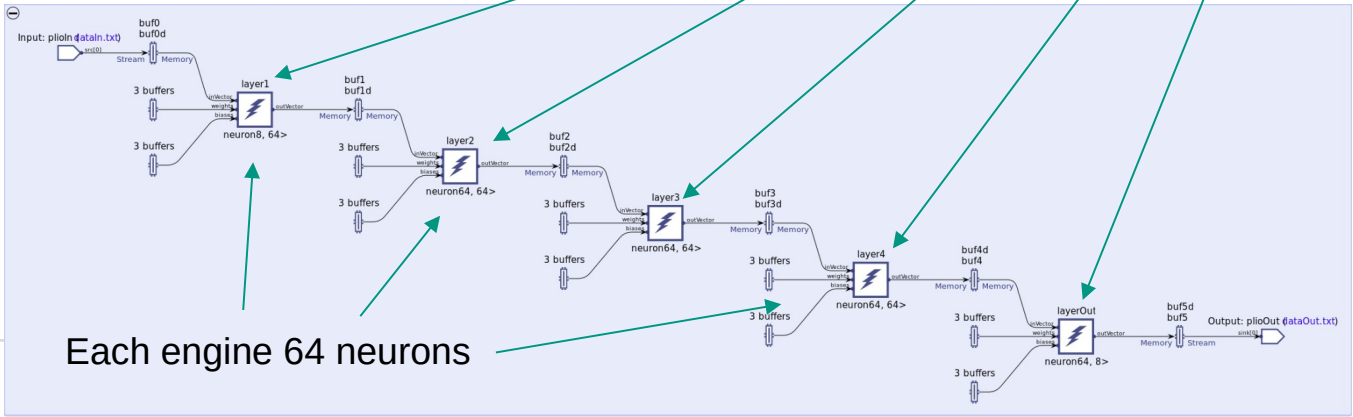
Why Versal: first benchmarks

- 4 layer, 64 neuron fully connected neural network
- 32 bit floating point computation
- AXI4-Stream input/output
- fully reconfigurable at runtime, easy to deploy
- limited hardware requirements (2% of AIE array)
- tested on hardware

Preliminary result (no optimization)

Latency measured ~4.5 µs, 4 times better than previous Ultrascale+ family FPGA (on 8 bit integers)





Each engine 64 neurons

W. Wang, M. Caselle, et al IEEE TNS, https://doi.org/10.1109/TNS.2021.3084515 (2021)

KINGFISHER

Intended as a general platform for fast inference for scientific applications (e.g. autonomous accelerator systems, ...)

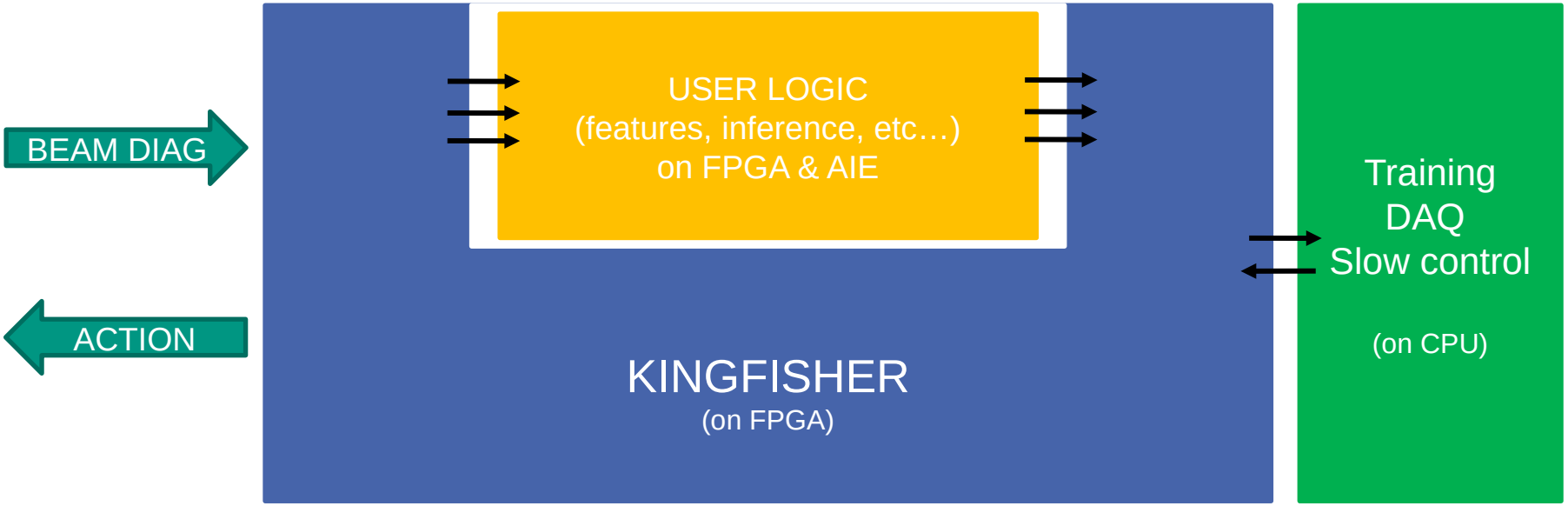
Idea:

Create general platform and hardware interfacing with beam diagnostic (KALYPSO, KAPTURE, …) and action control (interfacing with accelerator control system) providing standard way of accessing data and provide feedback.

Goal:

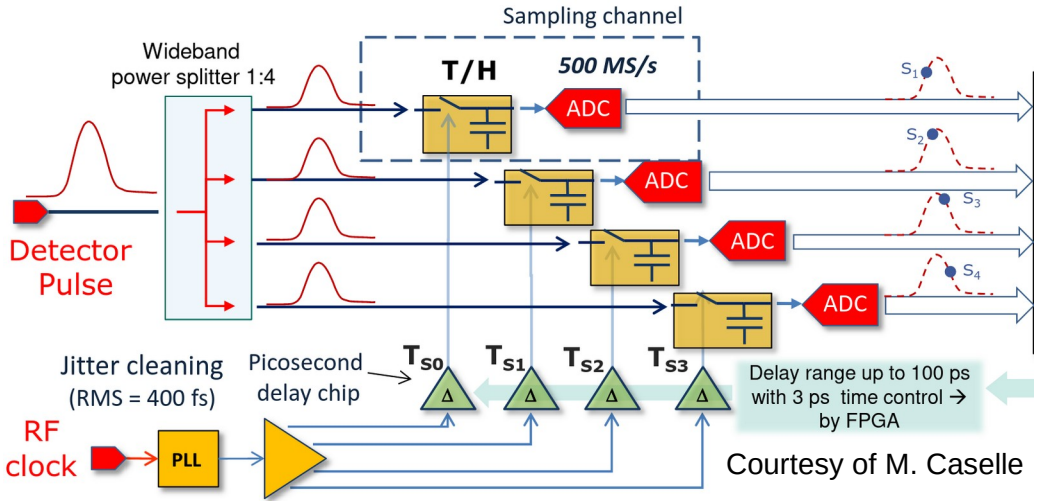
Do the system level part (interfacing to the world) once, let user program in “higher level” languages (C++, Python). Easier access to non- FPGA experts!

KINGFISHER: the structure



KAPTURE

Sampling of four points, local sampling rate over 300 Gsps. Selectable 3 ps delay for pulse shape reconstruction



Courtesy of M. Caselle

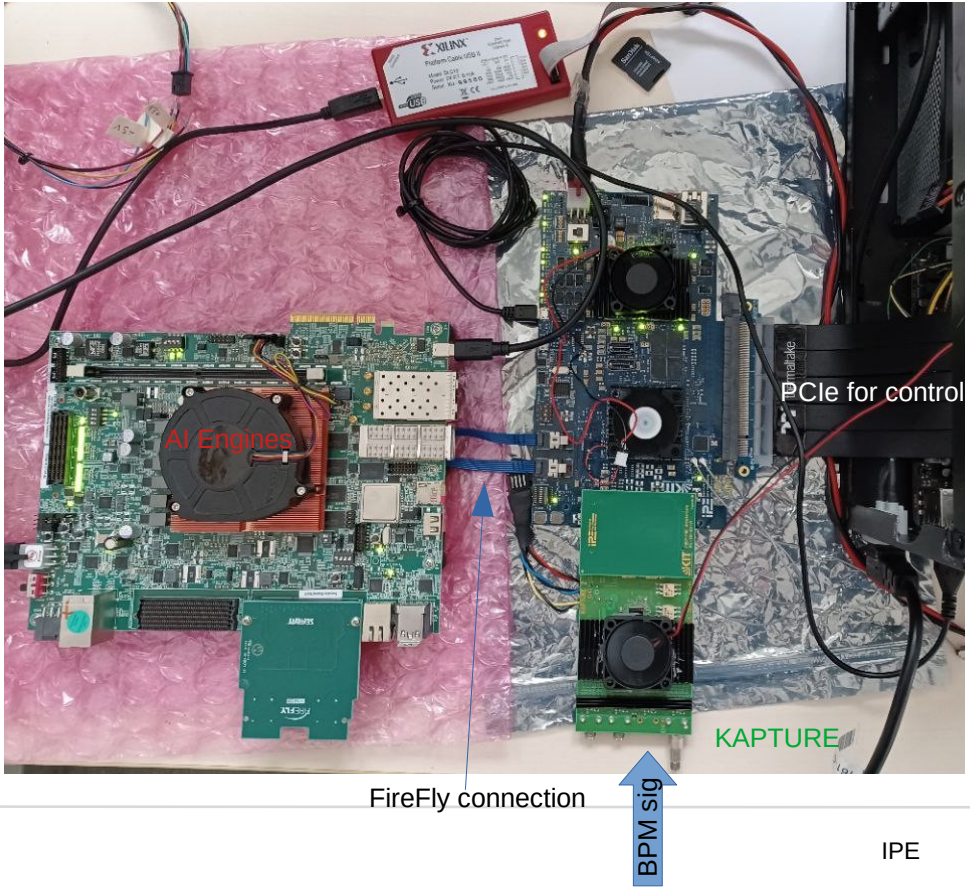
38 Gbps data rate per card

KAPTURE and Versal

Interconnection of Versal and KAPTURE systems requires:

- transfer data on particle accelerator scale (~100 m);
- latency as low as possible;
- high data throughput;

Aurora 64b/66b protocol from Xilinx



KAPTURE and Versal

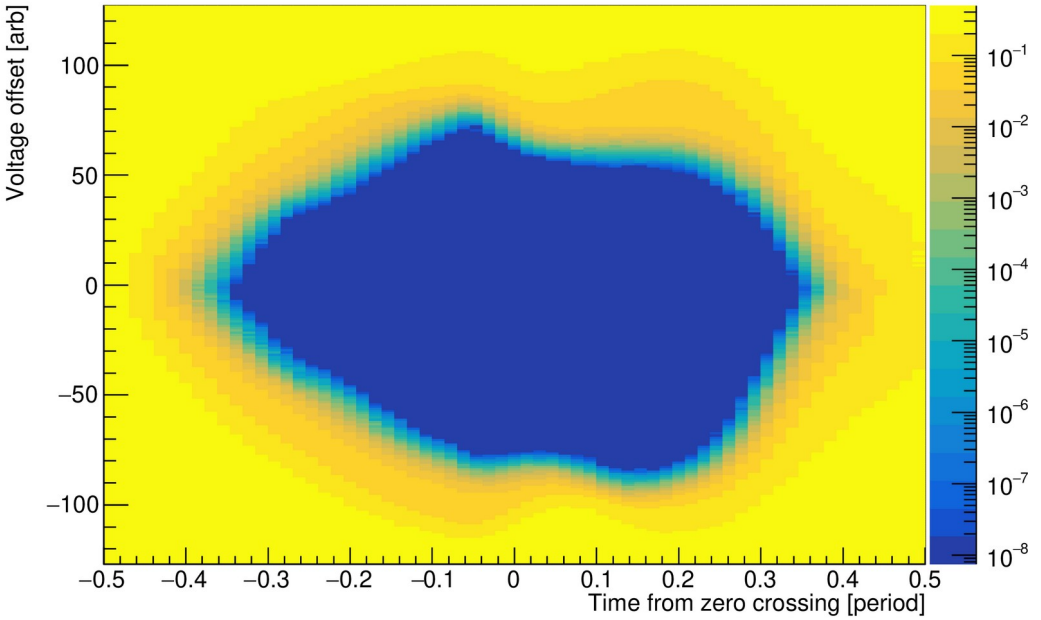
10 Gbps links

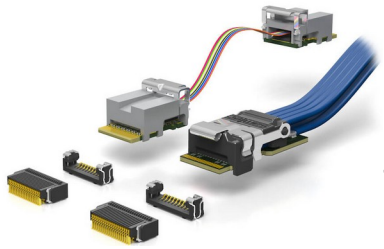
Four parallel channels in full duplex

Low Bit Error Rate (< 10^{-12})

Low latency (~ 300 ns)

Both copper and fiber physical layer, using Samtec FireFly connectors





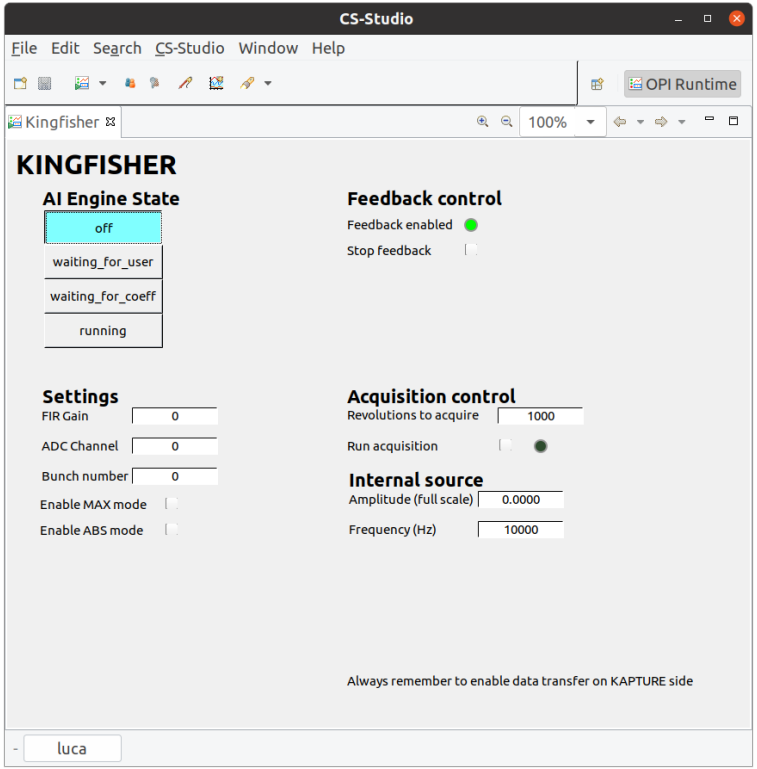
EPICS IOC

EPICS Input/Output Controller running using caproto on the ARM processor

Controls parameters and data acquisition

Fully integrated in KARA control system

Operator control panel in Control System Studio

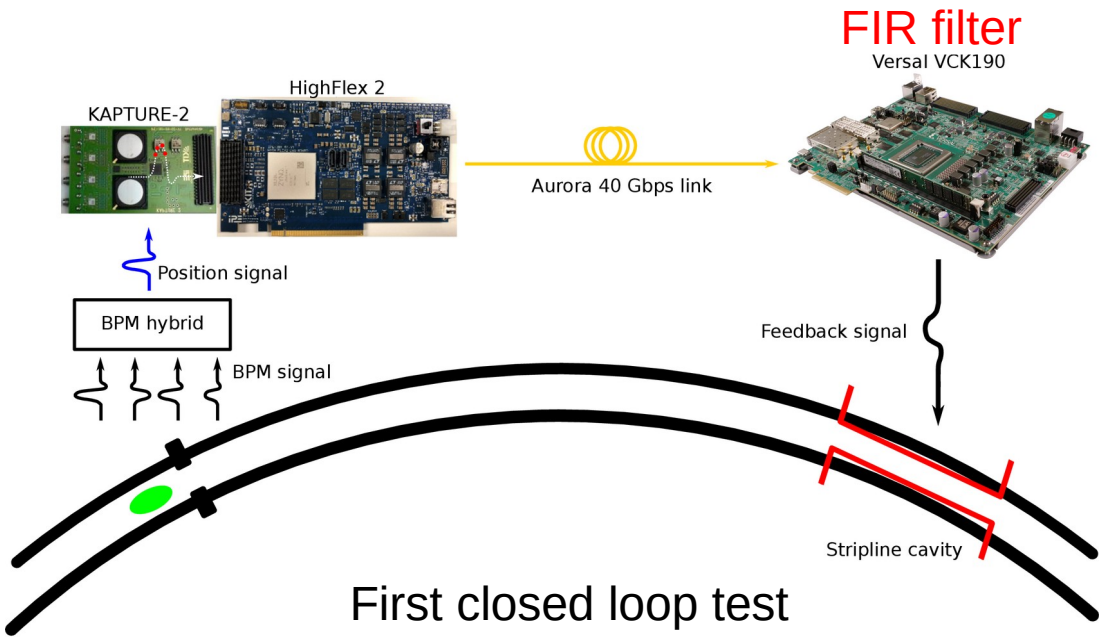


Preliminary horizontal feedback test

Goal is to control horizontal betatron oscillations

Good test as their apparent frequency is around 700 kHz

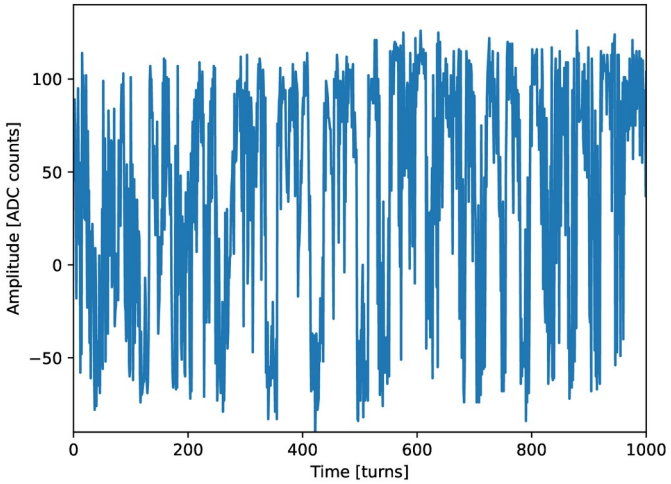
Lab tests showed a total latency of 2.4 µs

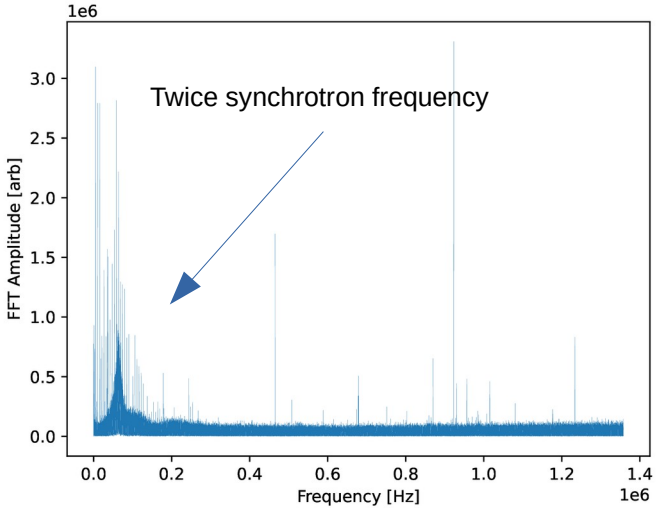


Readout issues

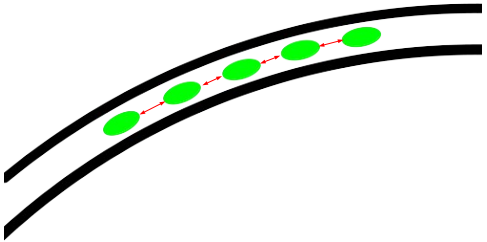
First BPM signal readout attempt failed

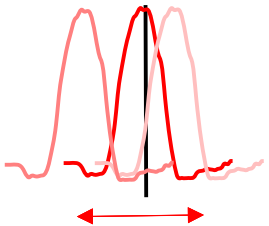
Strong noise dominated the signal





Noise due to coupled bunch instabilities: each bunch produces wakefield, that kicks itself/other bunches



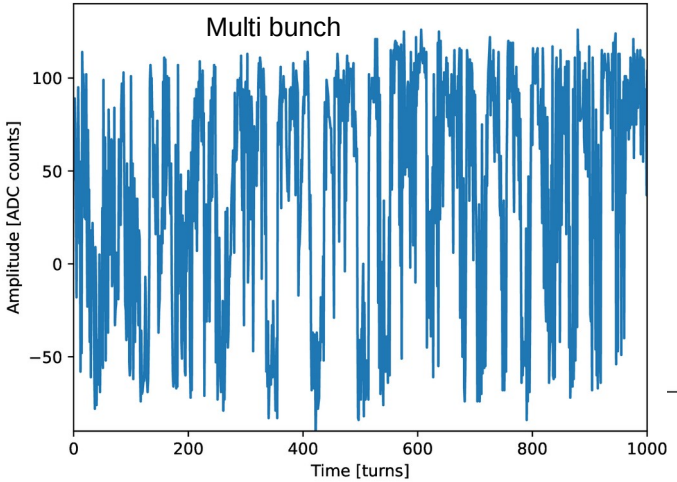


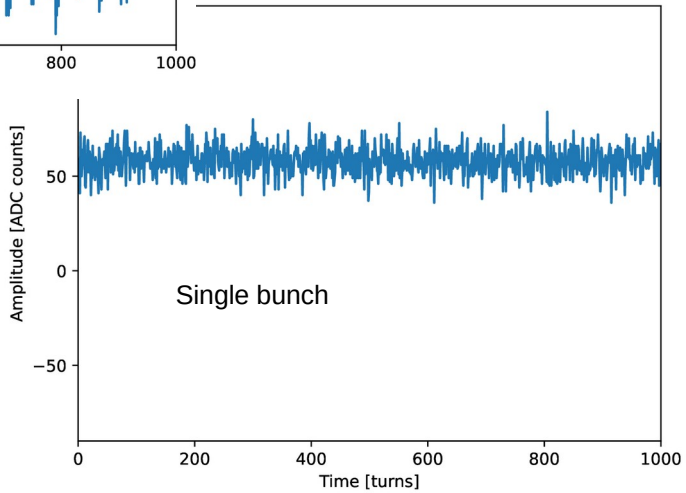
Readout issues: solution

Possible solution is to “fit” the peak using the four points.

An even easier one is to use a single bunch in order to reduce the wake field interactions

This last approach was successful

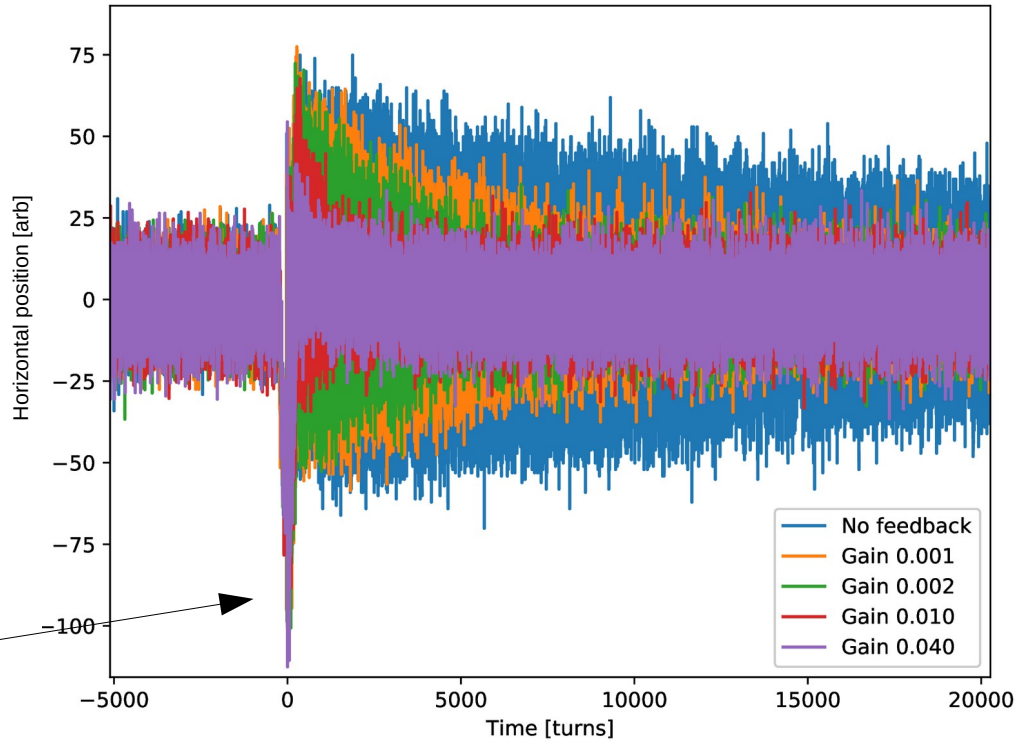




Feedback tests: injection kicker

Injection kicker moves the beam horizontally: it the oscillates at the betatron frequency and is damped by synchrotron radiation

Experimentally demonstrated that the feeback system can effectively improve this behavior



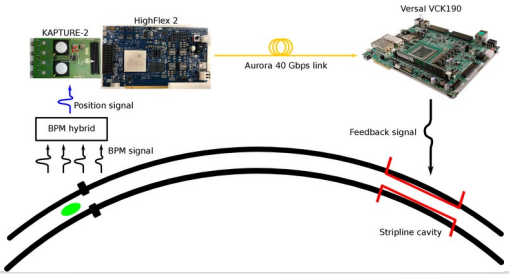
Injection kick

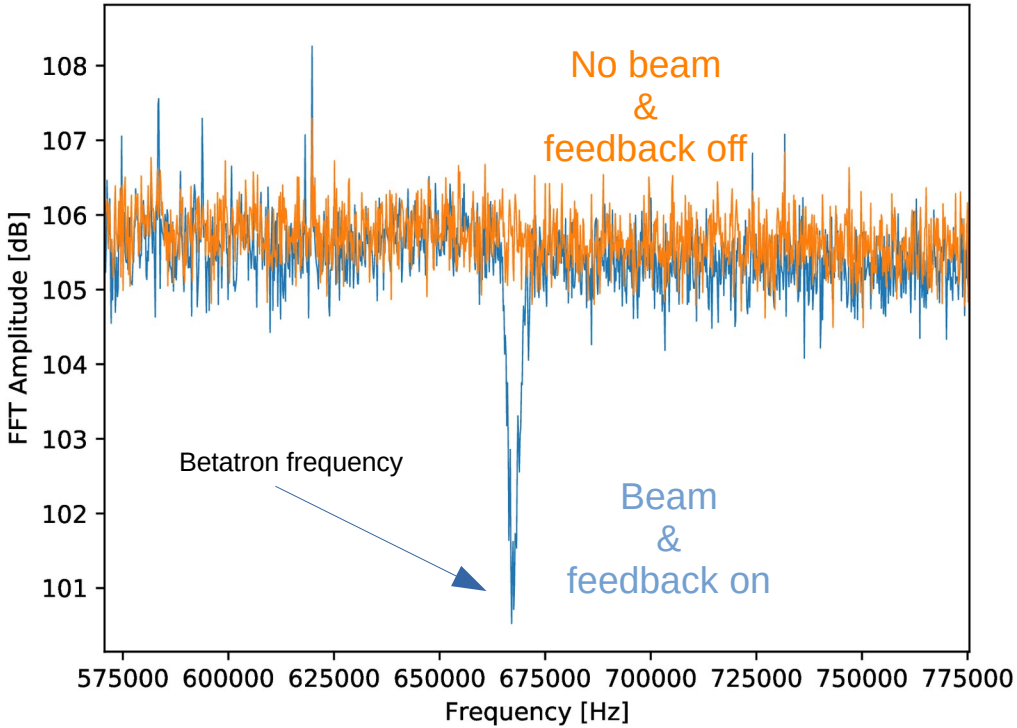
Feedback tests: the notch

Noise of the BPM and front-end input gets propagated, amplified by the beam and then reduces the input noise at betatron frequency

Works just at the betatron resonance

Good indication feedback is working





Conclusions

Complete feedback system tested
Feedback system was working as expected

Outlook:
Soon reinforcement learning tests

Thanks to this tests, new Low Level RF feedback input to be installed soon @ KARA

Thank you for your attention

Backup slides

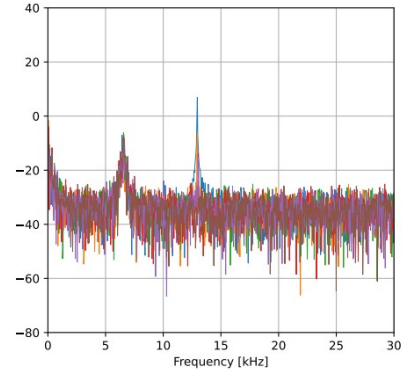
Action choice

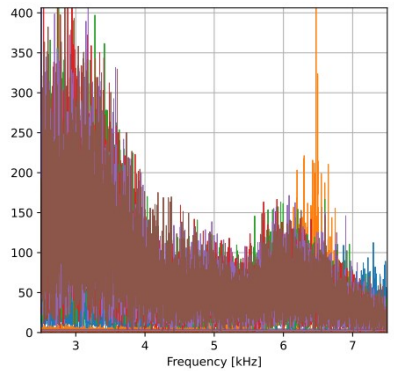
Two different possible longitudinal kicks:

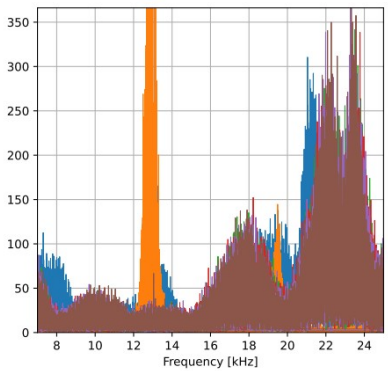
1) Bunch-By-Bunch feedback cavity
2) Low Level RF system

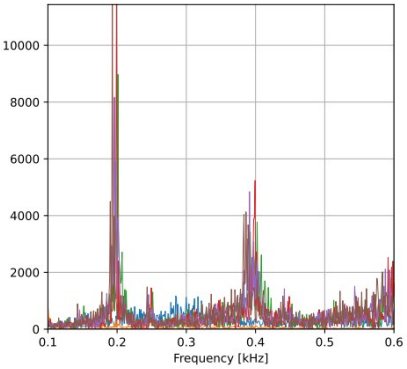
(1) can be performed separately on each bunch, but not very strong

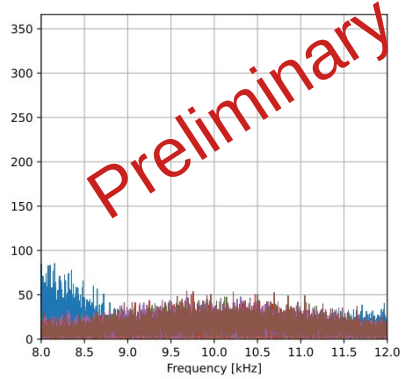
(2) extremely strong, but acts on all bunches











Basic param	Value	Unit
Energy	1.30016	GeV
Current	0.157313	mA
Burst thres.	0.108866	mA
RF voltage	854.279	kV
RF freq.	499.744	MHz
Mom. comp.	0.00836007	
Sync.freq.	6.55591	

LLRF 4	Value	Unit
Voltage	0	kV
Frequency	12.9413	kHz
Phase	2	deg
LLRF 4	**Value**	**Unit**
Voltage	30	kV
Frequency	12.9413	kHz
Phase	0	deg
LLRF 4	**Value**	**Unit**
Voltage	30	kV
Frequency	12.9413	kHz
Phase	0	deg
BBB	**Value**	**Unit**
Voltage	0.75	kV
Frequency	13.0413	kHz
Phase	1300	?
BBB	**Value**	**Unit**
Voltage	0.75	kV
Frequency	13.0413	kHz
Phase	150	?
BBB	**Value**	**Unit**
Voltage	0.75	kV
Frequency	13.0413	kHz
Phase	850	?

Courtesy of A. Santamaria Garcia

Result: (1) seems not to do the job!

Synchrotron radiation 101

Accelerated charges emit radiation

In synchrotron charged particles move in a circle → centripetal acceleration → synchrotron radiation

Very bright, short pulses, extremely broadband → **interesting for user**

In some cases **coherent**

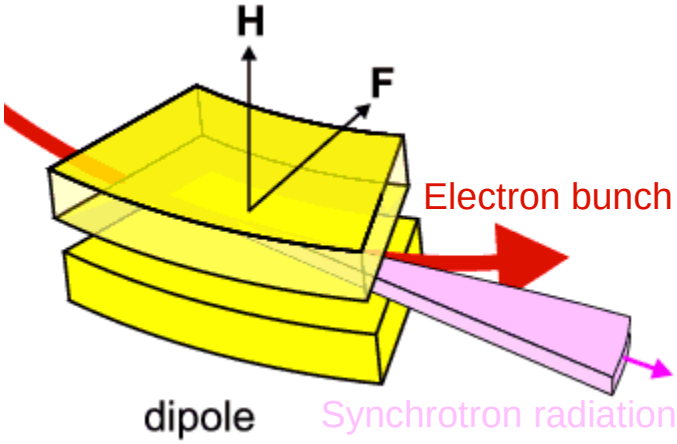
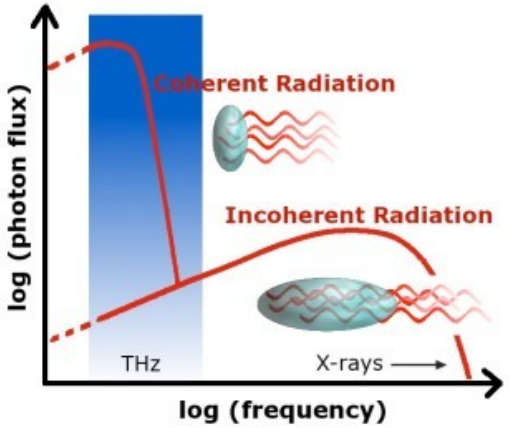


Image source http://pd.chem.ucl.ac.uk/pdnn/inst2/prop.htm



Sampling signals

Signal from synchrotron radiation detector:

- bunch every 2 ns
- would like picosecond granularity, would mean THz sampling rate!

Standard High Speed A/D Converters

Choose Parameters | All | Reset Table | Maximize Filters | Sort by Newest | Save to myAnalog | Download to Excel | Share | Quick Tips | Send Feedback

Compare

517 parts

Part Number	Analog.com Inventory	Channels	Resolution *bits*	Sample Rate *max | SPS*	ADC SNR in dBFS *typ | dBFS*	INL in LSB *typ | LSBs*	Vin Range *typ | V p-p*	Data Output Interface
Filter Parts		1 - 16	6 - 16	7.5M - 10.25(	37 - 84.1	0.1 - 35	0.1 - 5	OR AND 22 Values Selected
	HIDE	HIDE	HIDE	HIDE	HIDE	HIDE	HIDE	HIDE
AD9213	1	1	12	10.25G	57.5	2.4	1.4	JESD204B
AD9213S		1	12	10.25G	57.5	2.4	1.4	JESD204B
AD9207	Check Distributor Inventory	2	12	6G	53.4	1.38	1.475	JESD204B, JESD204C

But they don't sell it yet!

AI Engine Array

Array of AI Engine Tiles, separate block tightly connected to FPGA

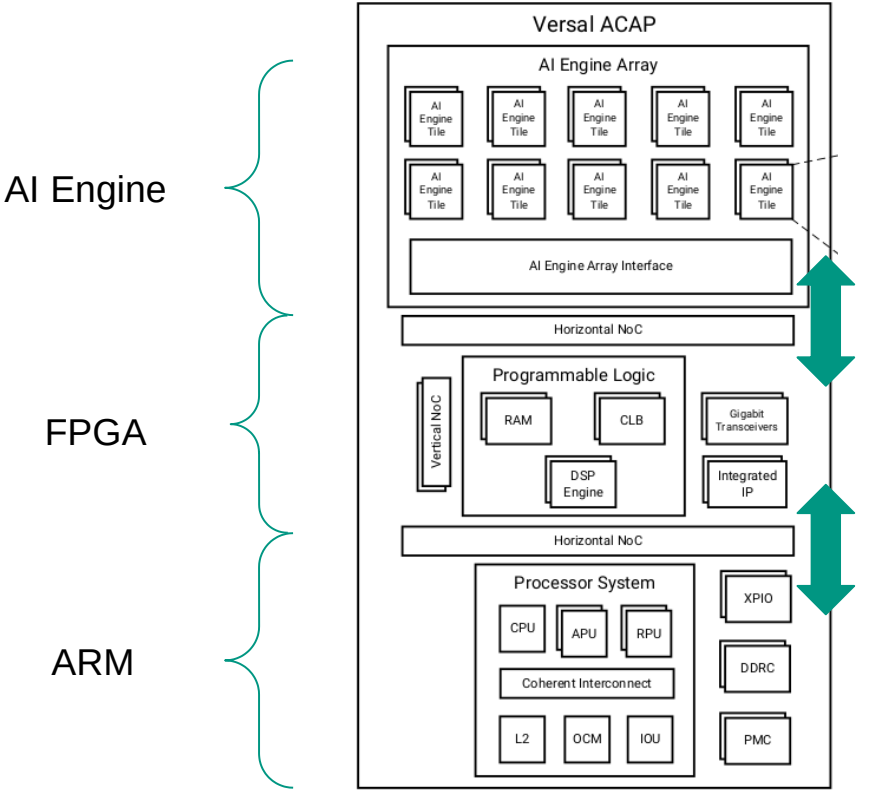
In our board 400 tiles

Each tile is composed of:

- Vector Processing Unit
- 32 kB of memory
- Several interfaces to other tiles

Multiple operations per clock cycle:

- 2 vector load
- 1 vector write
- several multiply and accumulate



AI Engine Array

Array of AI Engine Tiles, separate block tightly connected to FPGA

In our board 400 tiles

Each tile is composed of:

- Vector Processing Unit
- 32 kB of memory
- Several interfaces to other tiles

Multiple operations per clock cycle:

- 2 vector load
- 1 vector write
- several multiply and accumulate

400 tiles x 8 FP multiplications x 0.5 GHz

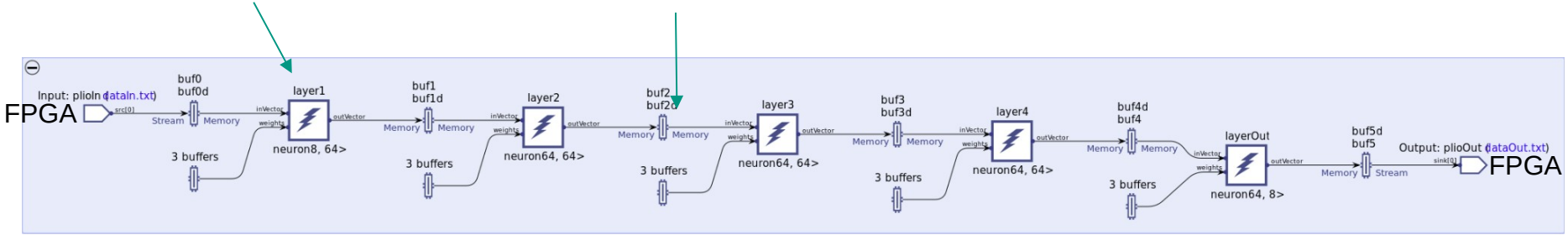
up to 1.6 Tera FP operations per second !!!

Multiplications per clock cycle

	X Operand	Z Operand	Output	Number of MACs
int	8 real	8 real	48 real	128
	16 real	8 real	48 real	64
	16 real	16 real	48 real	32
	16 real	16 complex	48 complex	16
	16 complex	16 real	48 complex	16
	16 complex	16 complex	48 complex	8
	16 real	32 real	48/80 real	16
	16 real	32 complex	48/80 complex	8
	16 complex	32 real	48/80 complex	8
	16 complex	32 complex	48/80 complex	4
	32 real	16 real	48/80 real	16
	32 real	16 complex	48/80 complex	8
	32 complex	16 real	48/80 complex	8
	32 complex	16 complex	48/80 complex	4
	32 real	32 real	80 real	8
	32 real	32 complex	80 complex	4
	32 complex	32 real	80 complex	4
	32 complex	32 complex	80 complex	2
float	32 SPFP	32 SPFP	32 SPFP	8

AI Engine Programming

User defined tile task and how they are connected



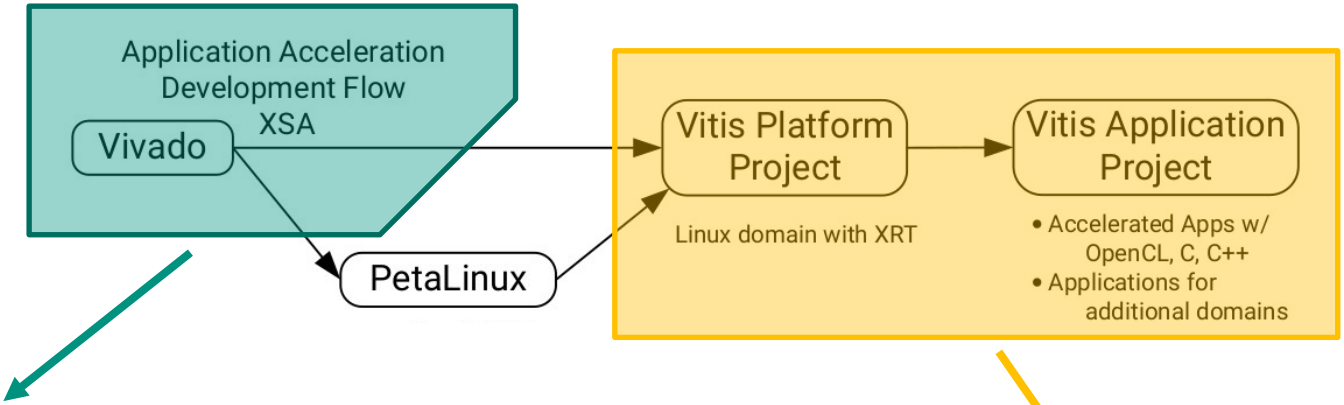
Programming in C++ as an Adaptive Data Flow Graph (ADF)

Capabilities:

- runtime definition of parameters (interesting for Reinforcement Learning applications)
- addition of kernels on FPGA (interesting for Convolutional Neural Networks)
- much more

Versal programming: a job for two people

Use of Xilinx Acceleration Kernel Flow



FPGA I/O
Clock
System level design

Use of IP integrator, VHDL, Verilog, etc..

Write actual programs (Linux of bare metal)
Target processor, AI Engine and FPGA (HLS)

Use of C/C++, in some case Python