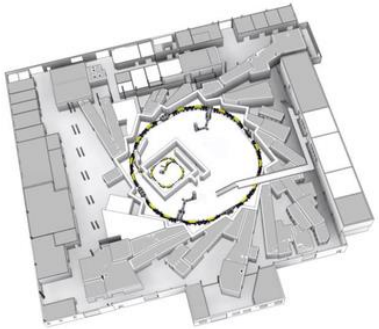
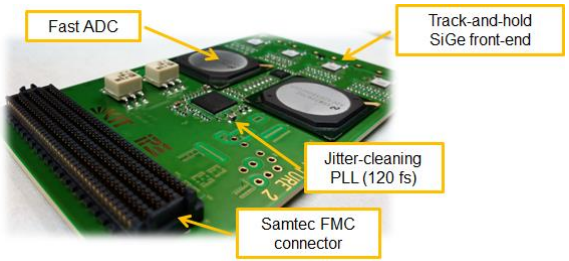
Karlsruhe Institute of Technology

*22nd Virtual IEEE Real Time Conference, 12-23 October 2020*

# Accelerated Deep Reinforcement Learning for Fast Feedback of Synchrotron Beam Dynamics

Weijia Wang, Michele Caselle, Tobias Boltz, Edmund Blomley, Miram Brosi, Timo Dritschler, Andreas Kopmann, Andrea Santamaría García, Erik Bründermann, Marc Weber, Anke-Susanne Müller, Yangwang Fang



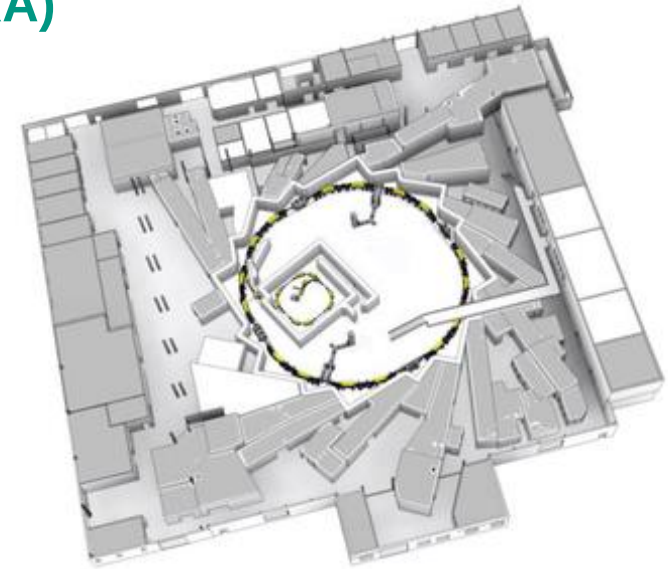


# Outline

- Motivations
- Requirements
- Hardware development
  - Front-end electronics for fast THz detectors
  - Fast feedback hardware implementation
- Real-time data processing by AI
  - Introduction of Reinforcement Learning
  - Real-time implementation
- Preliminary results
- Conclusions and what's next

# Accelerator Facilities at KIT

## Karlsruhe Research Accelerator (KARA)



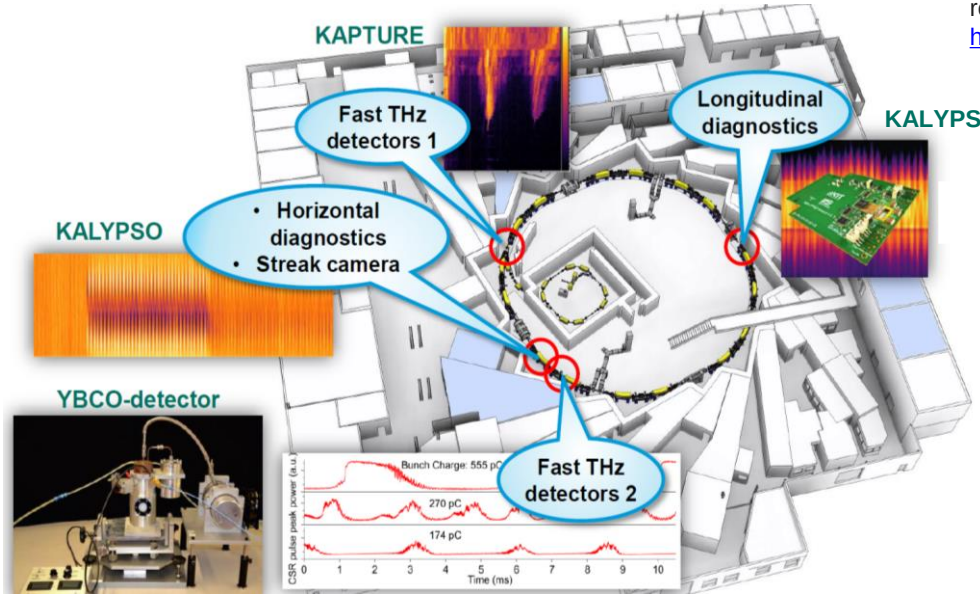
### Electron storage ring and Synchrotron light source

- Circumference: 110.4 meters
- Energy range: **0.5 - 2.5 GeV**
- Platform for development and testing of new beam and acceleration technologies
- Pooling research of new accelerator concepts
- Development of new detectors

# Karlsruhe Research Accelerator (KARA)

Caselle M., et al. KAPTURE-2. A picosecond sampling system for individual THz pulses with high repetition rate. JINST (2017), DOI: 10.1088/1748-0221/12/01/C01040

Rota, L., Caselle M., et al. KALYPSO: Linear array detector for high-repetition rate and real-time beam diagnostics. NIM A (2018), DOI: https://doi.org/10.1016/j.nima



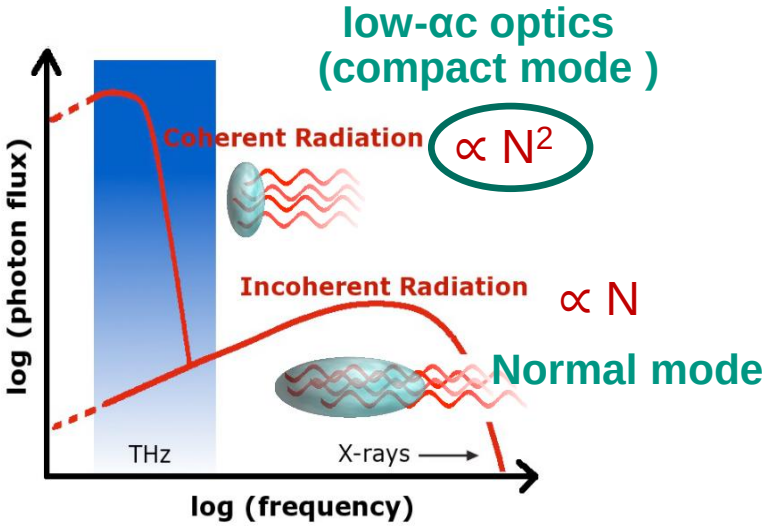

- KARA: one of the 1st to offer short bunch
- (low-$\alpha_c$ optics) bunch length down to few ps
- Enable to study the microbunching instability
- Custom detectors developed for a real-time resolved phase space tomography

Kehrer B., et al. Synchronous detection of longitudinal and transverse bunch signals at a storage ring. *Phys. Rev. Accel. Beams (2018),* 102803

# Coherent Synchrotron Radiation (CSR)

## Generation of CSR

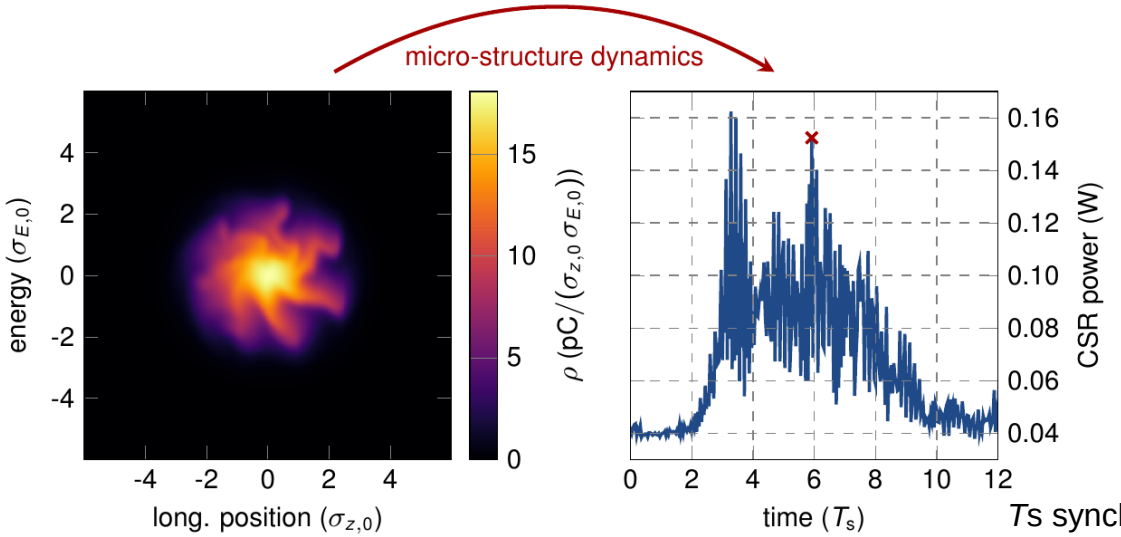
- When the wavelength of the emitted radiation is considerably shorter than the bunch length ($\lambda \ll \sigma_z$), the synchrotron radiation is emitted coherently
- CSR = high radiation power
  - → interesting for users!
  - → operation in short bunch mode (low $\alpha_c$ optics)
- **Short bunch mode → microstructures appear, making the CSR power fluctuate**
- Goal: CSR stabilization with high average (power) and low variance




A.-S. Müller, et al. “Experimental Aspects of CSR in the ANKA Storage Ring” ICFA Beam Dynamics Newsletter (2012), No. 57, 154–165

# KARA and Micro-bunching instability

- High intensity of CSR power
- Complex dynamics of the bunch makes the control challenging
- Real-time control with latency down to few hundred of μs




**GOAL: Control the Micro-bunching instability by Machine Learning**

$T$s synchrotron oscillation period of 7 kHz

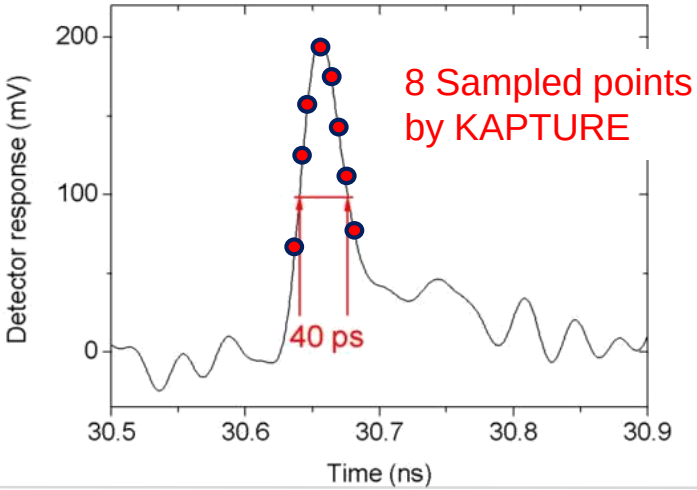
# KAPTURE - Measurement of bunch properties

## Karlsruhe Pulse Taking Ultra-fast Readout Electronics (KAPTURE)

KAPTURE is an ultra-wideband front-end electronics for ultra-fast Terahertz detectors

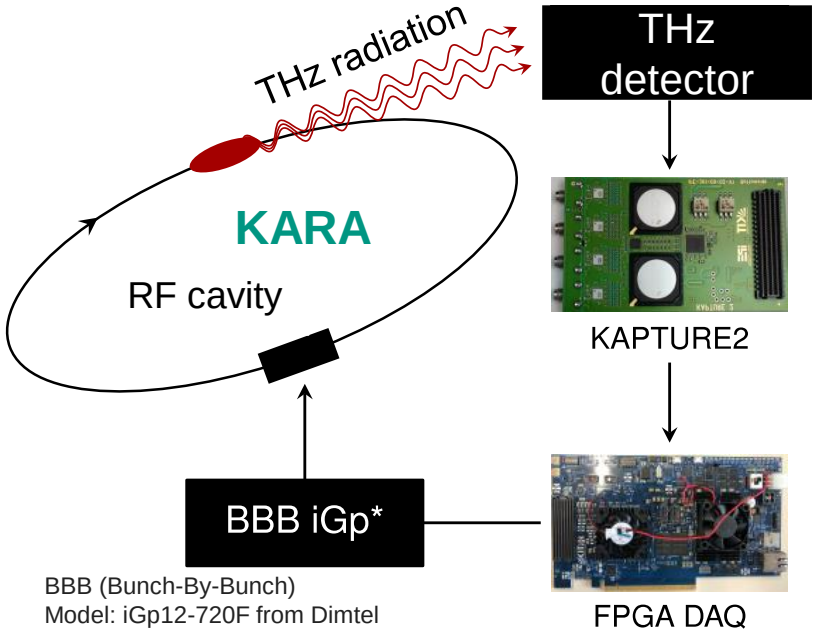
**KAPTURE board**

**Terahertz detector pulse**

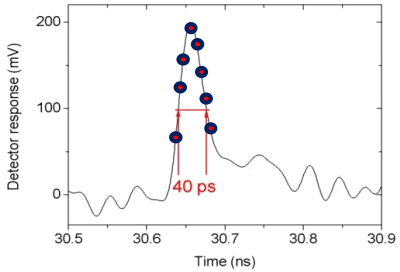



- Sampling time of 3 ps, local sampling frequency > 300 GS/s
- Up to 1 GHz pulse repetition rate
- Pulse amplitude (mV) and arrival time (ps) accuracy
- Real-time pulse sampling for long observation time
- Real-time FFT and measurement of CSR fluctuation by GPU

# Hardware Design of Feedback Loop




KAPTURE




- Sampling the CSR
- Extraction of the frequency of the instability

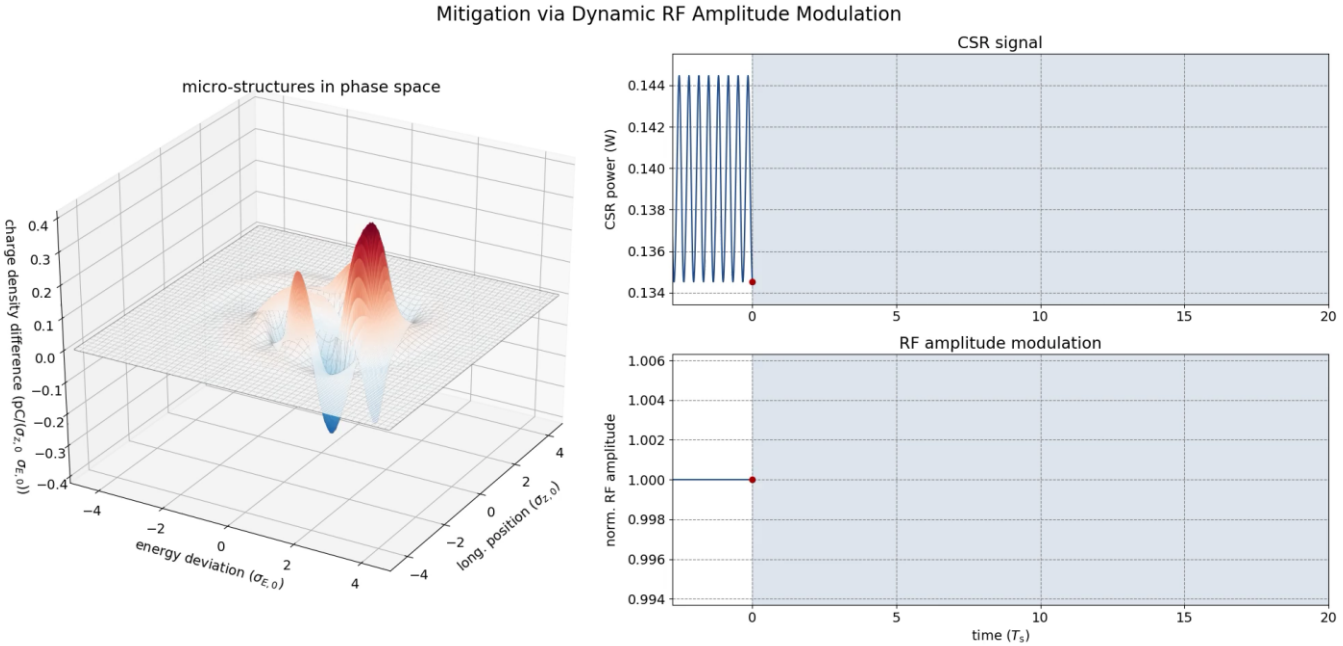
FPGA DAQ:

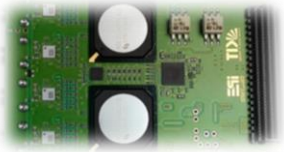
- Real-time process of CSR signal
- RL algorithm running on the FPGA receives the measurements of CSR and decides the action (RF modulation)

The action is processed and, by the Bunch-By-Bunch feedback system, sent to the kicker cavity

T. Boltz, W. Wang et al, Accelerating Machine Learning for Machine Physics (an AMALEA-project at KIT), ICALEPCS (2019), doi:10.18429/JACoW-ICALEPCS2019-TUCPL06

# Simulation of Feedback Control



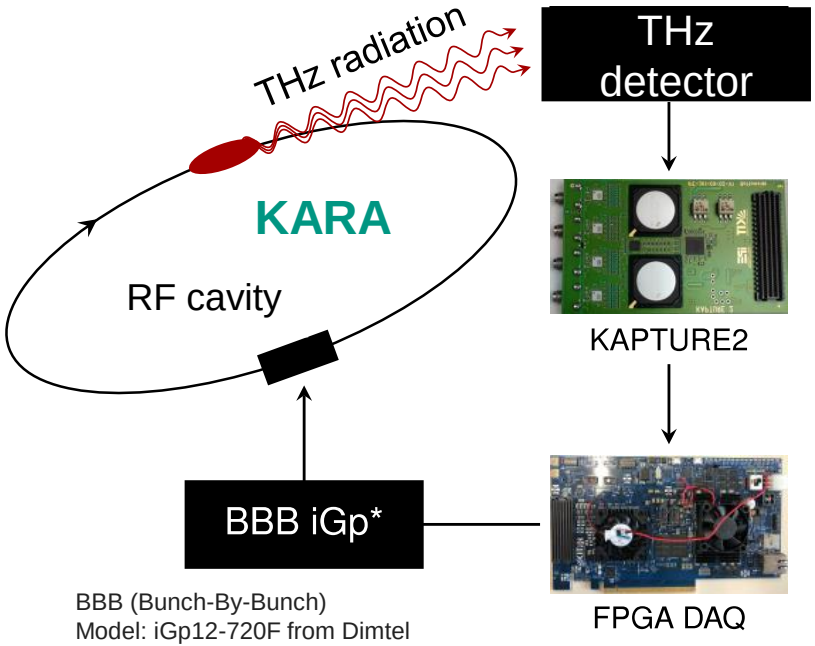



THz CSR acquired by KAPTURE

Example of RF cavity control (real-time)

→ To be implemented by AI

# Time Requirement




BBB (Bunch-By-Bunch)
Model: iGp12-720F from Dimtel

**Which Algorithm?**
**How to implement on Hardware?**

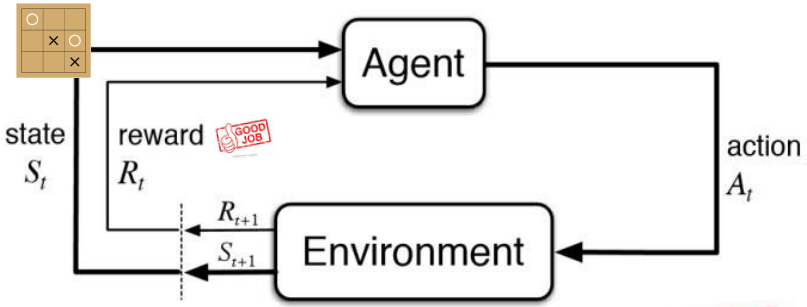
## Timing Contributions

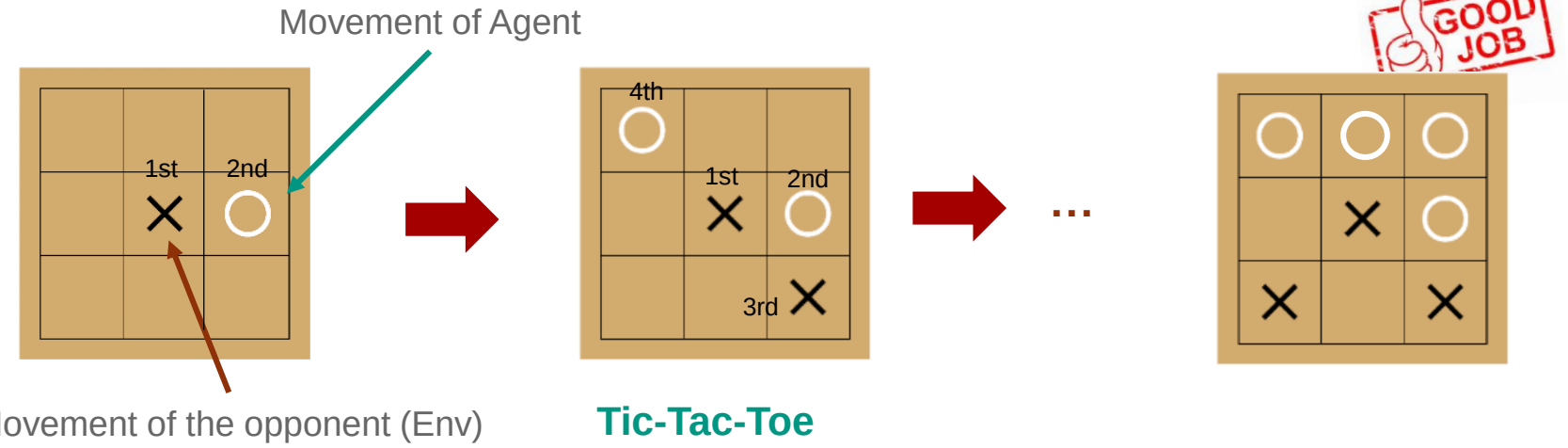
+ 1. THz detector/s
+ 2. Front-End Electronics
+ 3. Measurement of CSR signal (amplitude, time, frequency)
+ 4. Feature extraction from CSR
+ 5. Data processing based on AI processing
+ 6. Fast feedback to BBB

$$\sum_i t_i < \textbf{Hundreds of µs}$$

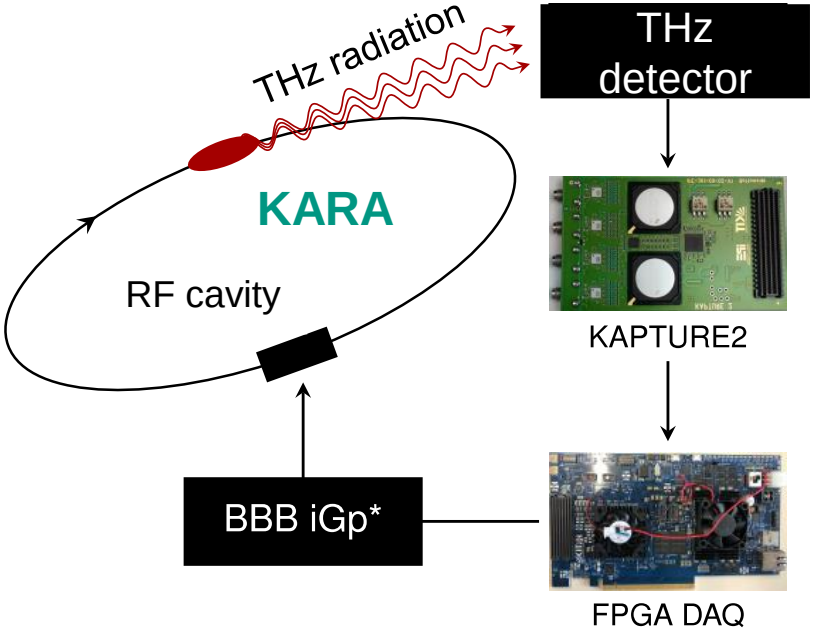
# The Reinforcement Learning Approach

- Goal-directed learning from interaction with an environment
- No pre-existing data set, trial-and-error learning → reinforce “good” behavior
- Markov Decision Process (MDP)



# Defining Reinforcement Learning




The action is processed and, by the Bunch-By-Bunch feedback system, sent to the kicker cavity

## Observable (state definition): CSR signal

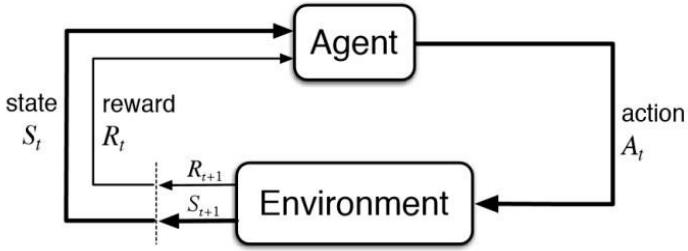
We can consistently and reliably measure the THz emission thanks to KAPTURE II. Provides information about the micro-bunching dynamics

## Theoretical observable (state definition): charge distribution

Ideal, we use it in simulations.
Vlasov-Fokker- Planck solver Inovesa

## Reward

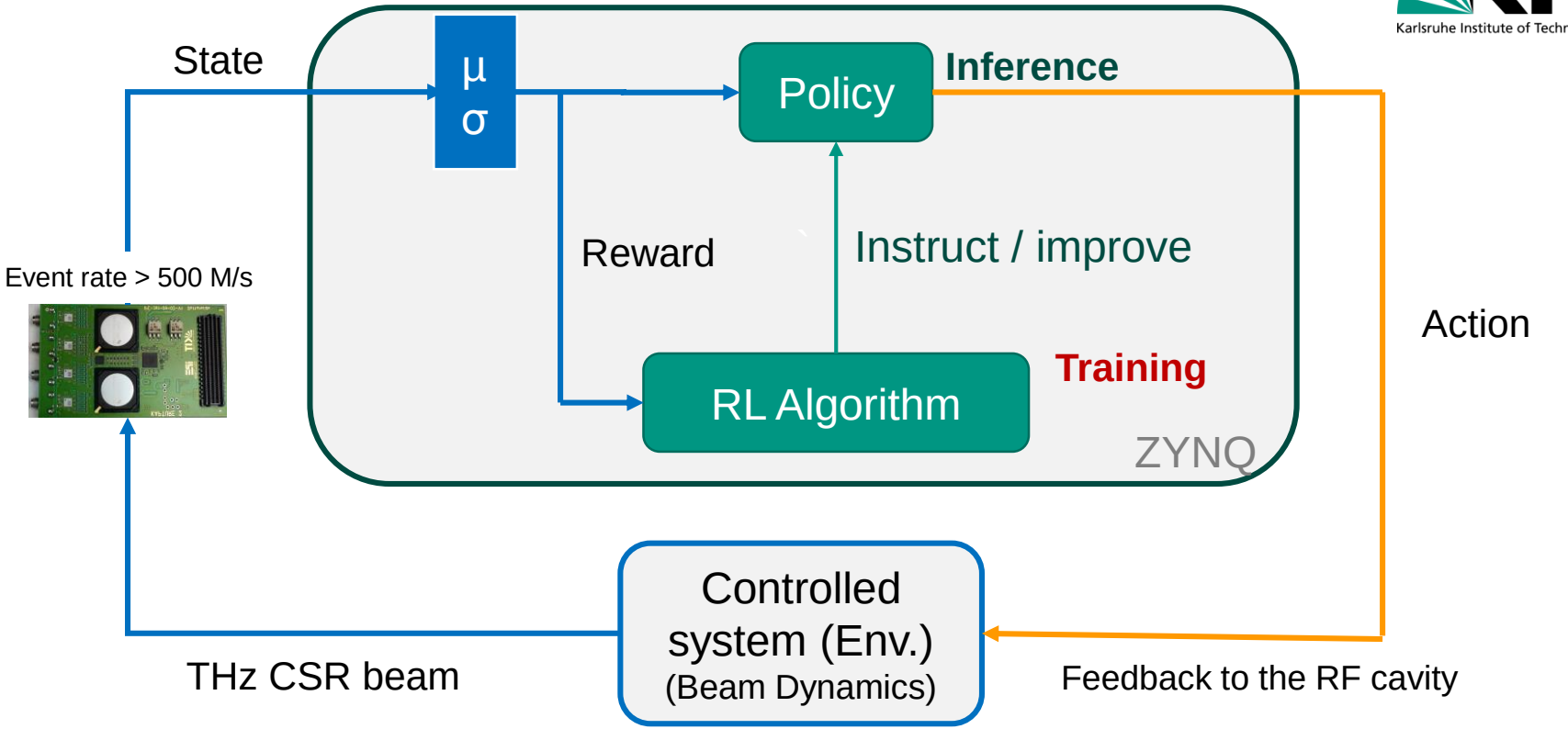
Based on average and STD



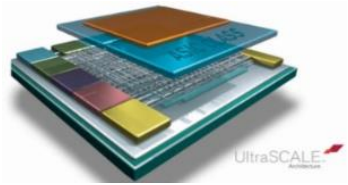

## Action

Modulation of the RF amplitude

# Hardware Deployment of RL Agent



# FPGA - AI platform: HighFlex 2




- Processor System (ARM): User Applications
- Programmable Logic (FPGA): fast and low latency application

Front-End:
KAPTURE
KALYPSO

- 12 lanes @ 28 Gbps
- Data throughput (full-diplex) up to 336 Gbps

- Compatible with standard FMC

PCI EXPRESS

- PCIe Gen 4 (x8 or 16 lanes)
- Data rate up to 240 Gbps

- Two SSD raid Local data storage

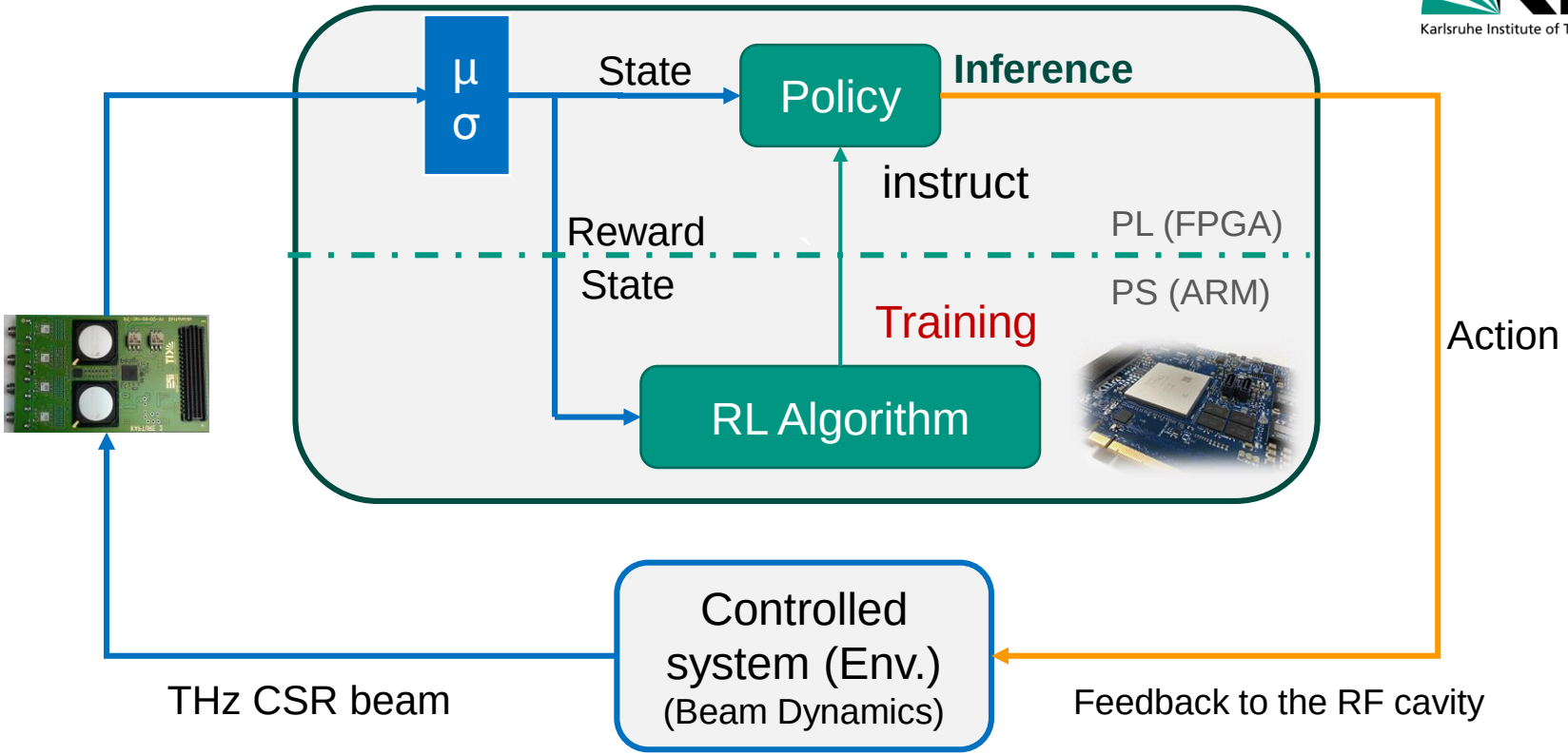
# HighFlex 2 FPGA: ZYNQ Ultrascale plus

- System Logic Cell: 653 K
- Flip-Flops: 597 K
- LUT: 298 K
- Distributed RAM: 9,1 Mb
- Block RAM: 21.1 Mb
- UltraRAM: 22.5 Mb
- DSP Slices: 2928
- PL-DDR4: 2 GB

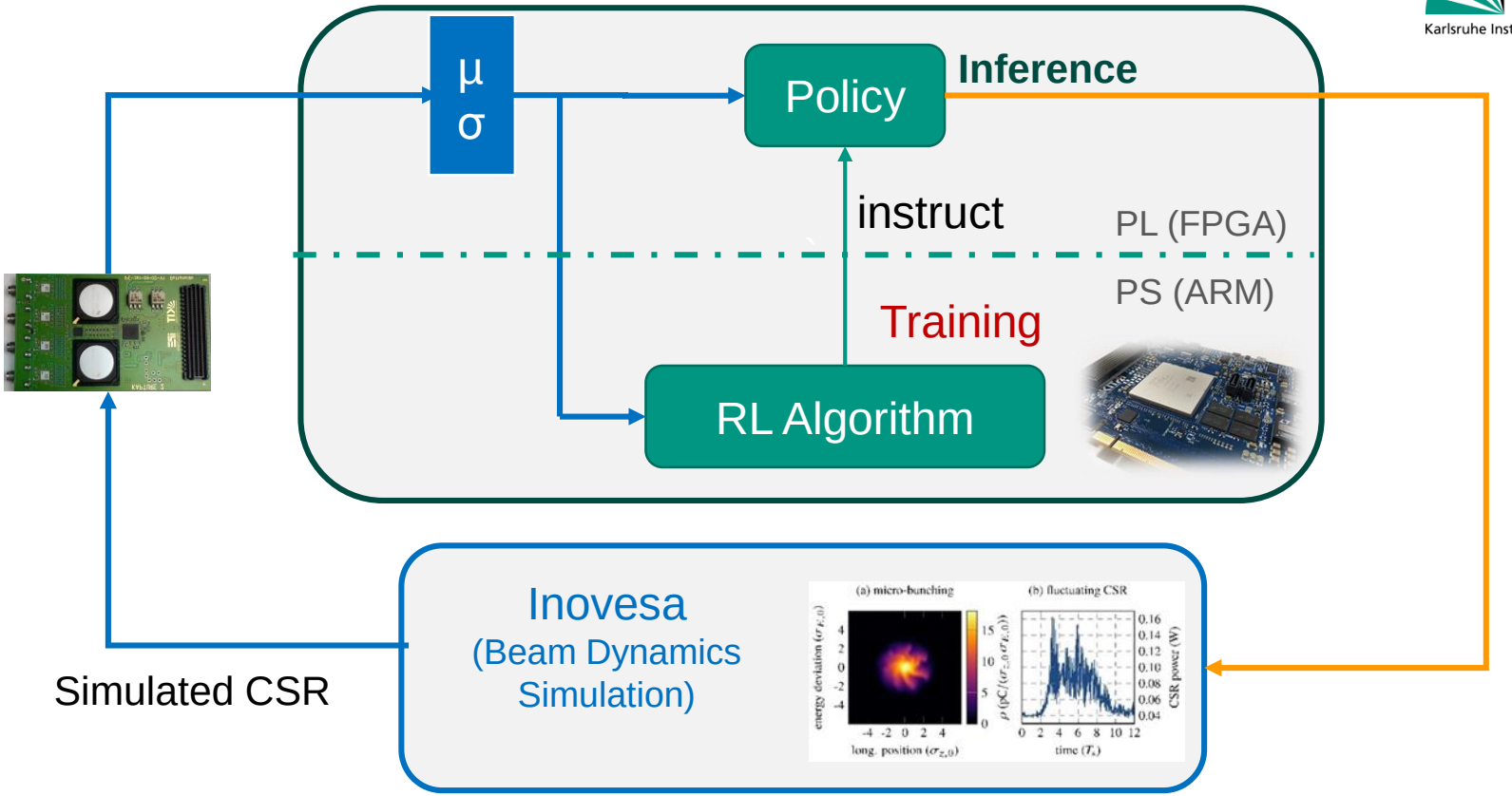
**AMBA AXI4 interfaces for primary data communication**

- Quad-core Arm Cortex-A53
- NEON & Single/Double Precision Floating
- PS-DDR4: 4 GB

# Hardware Deployment of RL Agent

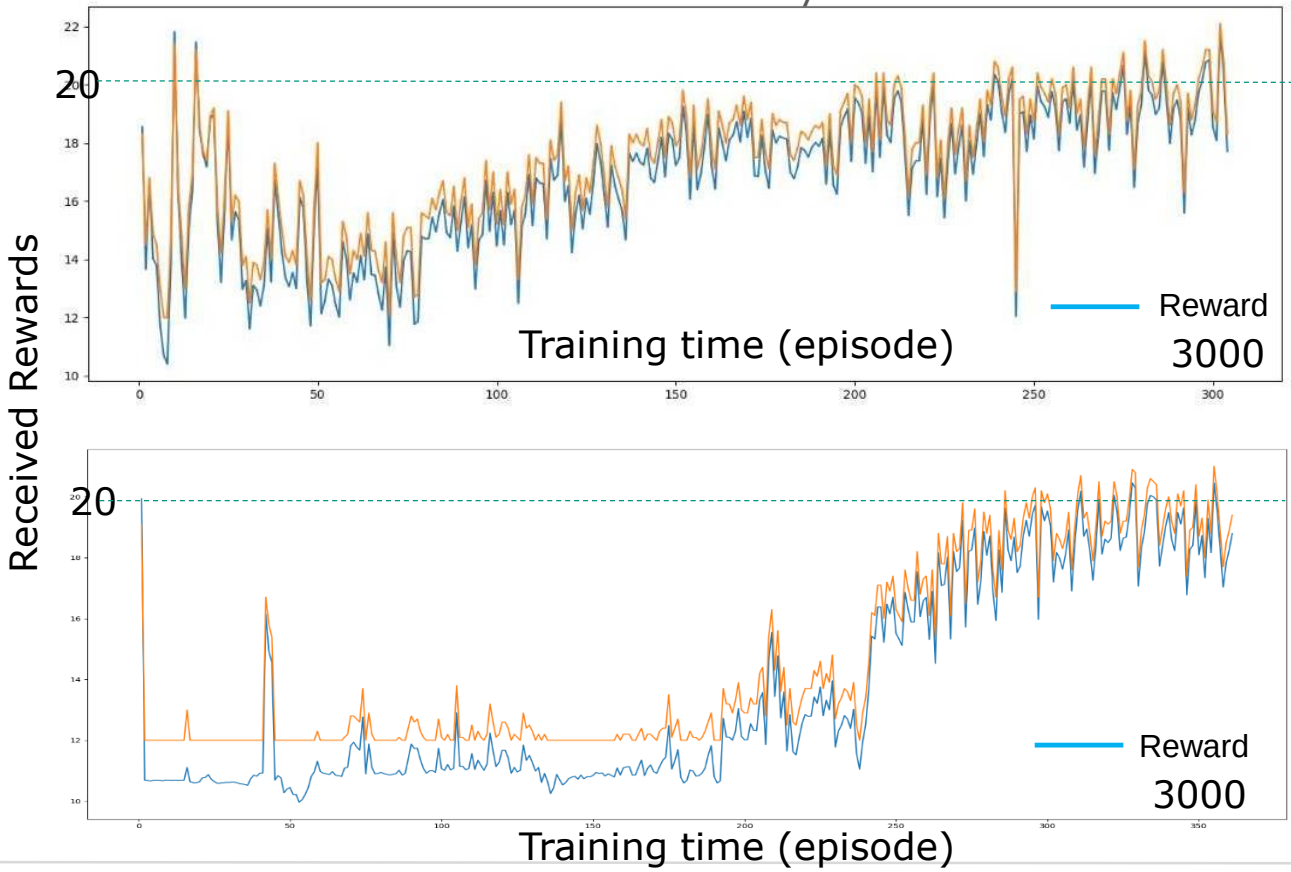


# Proof of Concept



# Comparison between FPGA and GPU (Keras-RL)

Simulation results by Inovesa



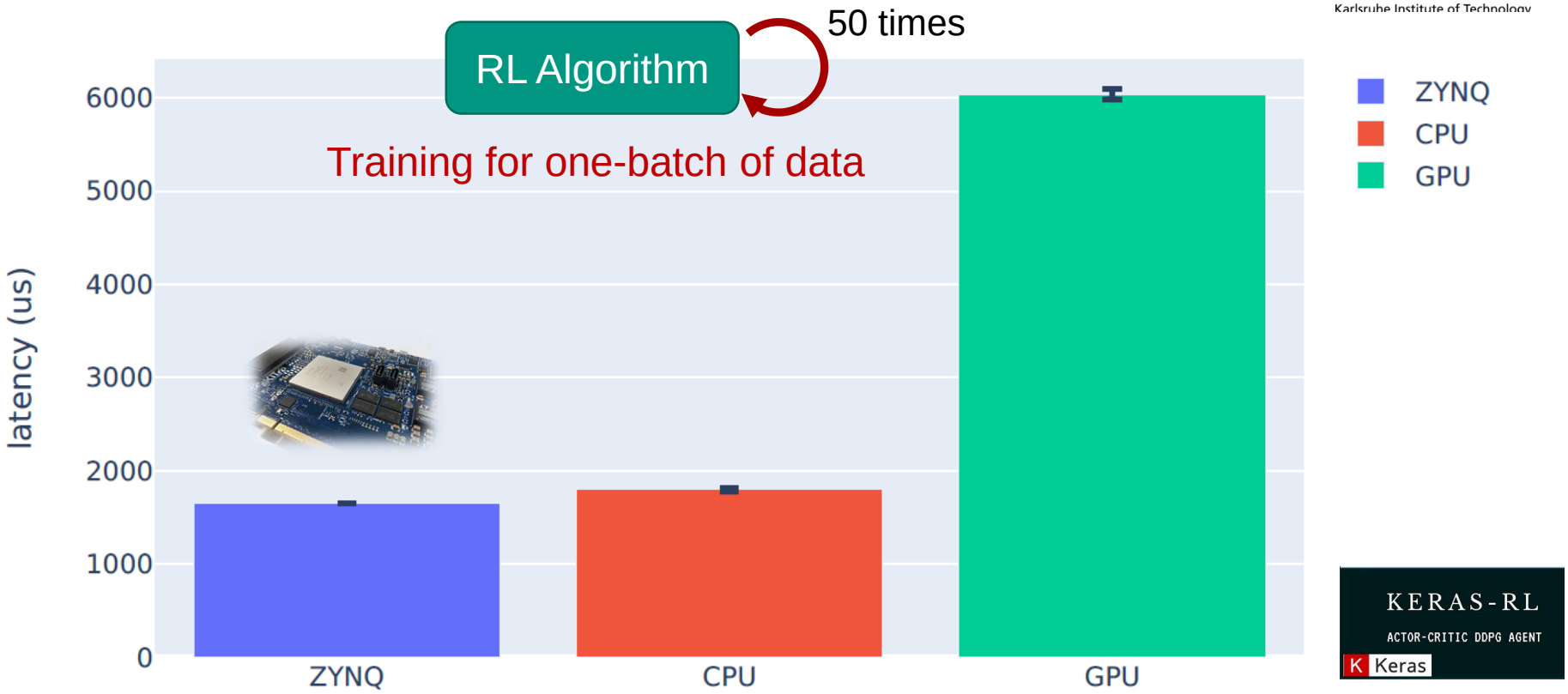
RL on FPGA

- Episode: agent behave badly and interaction stops
- The hardware achieve the similar learning curve compared to Keras-RL

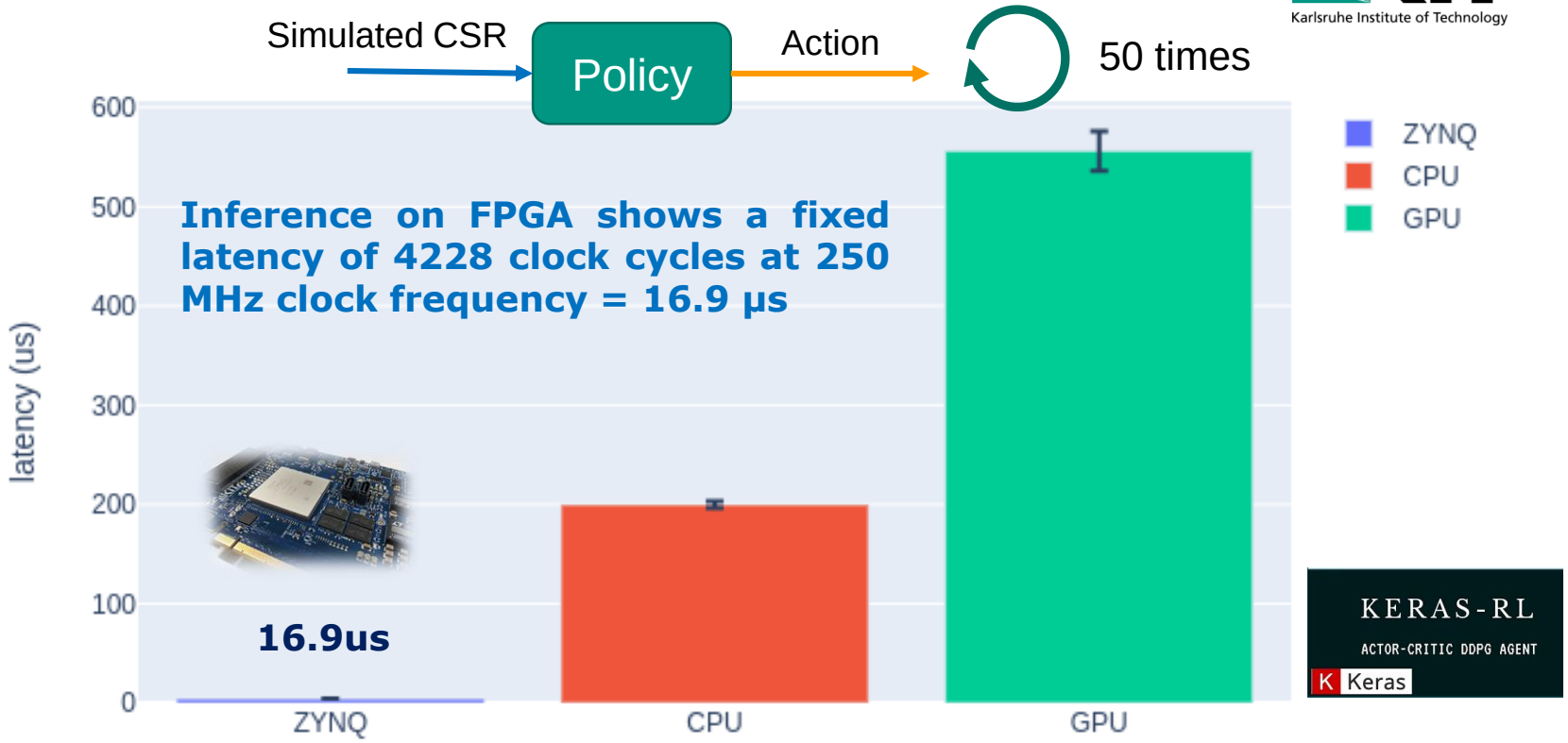
KERAS-RL
ACTOR-CRITIC DDPG AGENT
Keras

RL on GPU

# RL training performance



# RL inference performance



# Conclusion & what‘s next

- Microbunching instability study at KARA
- Fast detector and beam diagnostics fully developed
- Hardware implementation of the feedback loop control → tested

- Developed a framework for the implementation of an RL algorithm on ZYNQ-FPGA
- Preiminary results show a high inference performance of about 17 µs
- Training performance in the order of > 1 ms, *to be improved*

*What’s next*

- Test the feedback system including the RL at KARA
- Further investigation of potential support of different RL algorithms
- Further investigation of different FPGA and strategies to reduce the training time

Karlsruhe Institute of Technology

# Thank you for your attention

# Backup Slides

# Applying Reinforcement Learning to our case

Karlsruhe Institute of Technology

## Observable (state definition): CSR signal

We can consistently and reliably measure the THz emission thanks to KAPTURE II. Provides information about the micro-bunching dynamics

## Theoretical observable (state definition): charge distribution

Ideal, we use it in simulations.

Difficult to integrate in our feedback loop?

## Action

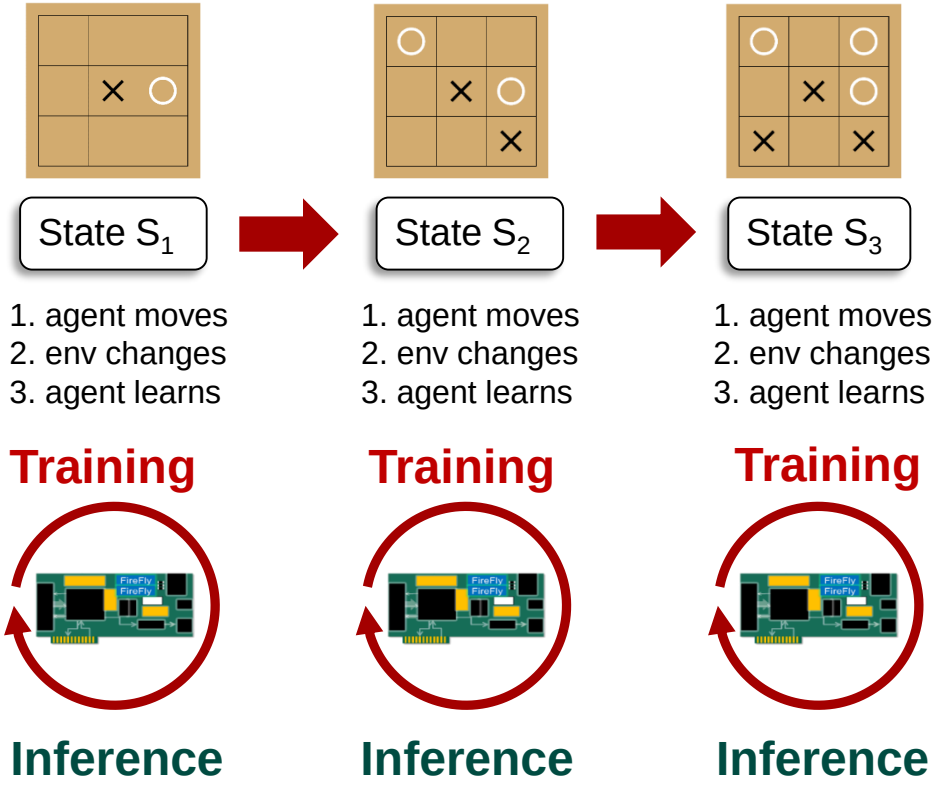
Modulation of the RF amplitude

## Reward

R = $\mu_{CSR} - w\,\sigma_{CSR}$ where $w$ is a weight

Could we improve the reward definition?

# Hardware Requirement of Reinforcement Learning




## Online Training

Each step involves training and inference!

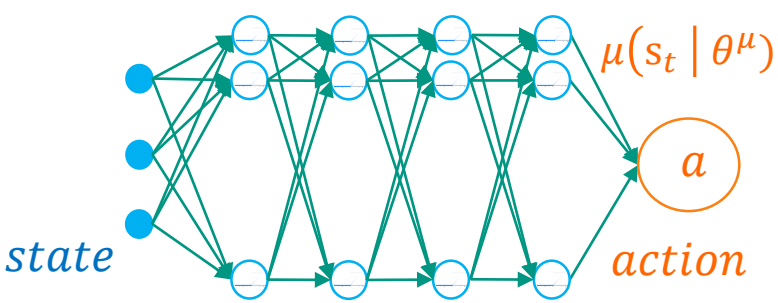
$$\sum_i t_i <$$

**Hundred of µs**

+ 3. Collect and interpret the CSR signal
+ 4. Transfer the control to BBB
+ 5. Generate the control signal
+ 6. Collecting training data
+ 7. Backward propagation
+ 8. Updating networks

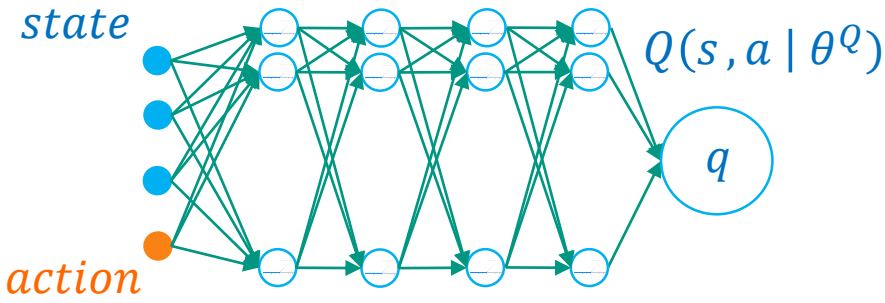
# Deep Deterministic Policy Gradient

Actor Network (Policy)




- Actor is component that learns policies
- Actor follows the instruction coming from the Critic Network, in practice, it update towards a direction that maximum the q value.

Critic Network (Value)

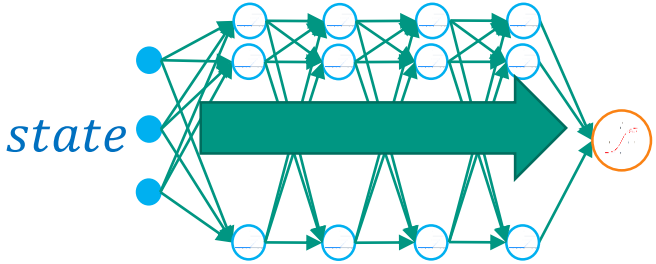



- Critic learns the state-value function that used for “criticize” the actor’s policy.
- Critic update through the TD errors, in practice, it is updated by the reward signal collected from the environment

# Hardware Implementation of DDPG

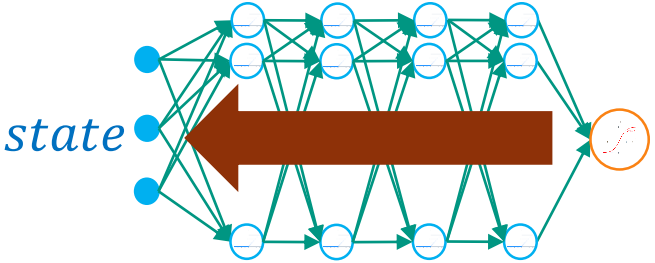
Actor Network (Policy)
Inference

$$\mu(s_t \mid \theta^{\mu}) = action$$




- Actor is used for generate action signal
- This needs a relatively easy calculation and fixed-point value can fulfill the requirement.

Actor Network (Policy)
Training




- Actor network training use the information from critic network for a backpropagation.
- This needs a relatively complex calculation and floating point value.