tinyML® Talks

Enabling Ultra-low Power Machine Learning at the Edge

"Multi-armed Bandit on System-on-Chip: Go Frequentist or Bayesian"

Sumit J Darak – Associate Professor, IIIT-Delhi

May 14, 2023

TINY ML

www.tinyML.org

Thank you, **tinyML Strategic Partners**,

for committing to take tinyML to the next Level, together

Executive Strategic Partners

TINY ML TALKS webcast

EDGE IMPULSE

The Leading Development Platform for Edge ML

edgeimpulse.com

Qualcomm
AI research

Advancing AI research to make efficient AI ubiquitous

Power efficiency

Model design, compression, quantization, algorithms, efficient hardware, software tool

Personalization

Continuous learning, contextual, always-on, privacy-preserved, distributed learning

Efficient learning

Robust learning through minimal data, unsupervised learning, on-device learning

A platform to scale AI across the industry

Perception

Object detection, speech recognition, contextual fusion

Reasoning

Scene understanding, language understanding, behavior prediction

Action

Reinforcement learning for decision making

Edge cloud

Cloud

IoT/IIoT

Automotive

Mobile

Accelerate Your Edge Compute

SYNTIANT

Making Edge AI A Reality

www.syntiant.com

Platinum Strategic Partners

Renesas is enabling the next generation of AI-powered solutions that will revolutionize every industry sector.

DEPLOY VISION AI
AT THE EDGE AT SCALE

Gold Strategic Partners

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

Where what if becomes what is.

Witness potential made possible at analog.com.

arm AI

Powering tinyML Innovation

Arm AI Virtual Tech Talks

The latest in AI trends, technologies & best practices from Arm and our Ecosystem Partners.

Demos, code examples, workshops, panel sessions and much more!

Fortnightly Tuesday @ 4pm GMT/8am PT

Find out more:
www.arm.com/techtalks

Driving decarbonization and digitalization. Together.

Infineon serving all target markets as
Leader in Power Systems and IoT

www.infineon.com

innatera

NEUROMORPHIC INTELLIGENCE FOR THE SENSOR-EDGE

www.innatera.com

Microsoft


The Right Edge AI Tools Can Make or Break Your Next Smart IoT Product
SensiML
Analytics Toolkit Suite
AutoML
Data Collection
Team Collaboration
Version Control and Model Management
sensiml.com/tinyML

STMicroelectronics provides extensive solutions to make tiny Machine Learning easy

synaptics®

ENGINEERING EXCEPTIONAL EXPERIENCES

We engineer exceptional experiences for consumers in the home, at work, in the car, or on the go.

www.synaptics.com

Silver Strategic Partners

Join Growing tinyML Communities:

14.5k members in
47 Groups in 39 Countries

tinyML - Enabling ultra-low Power ML at the Edge

https://www.meetup.com/tinyML-Enabling-ultra-low-Power-ML-at-the-Edge/

4k members
&
11.6k followers

The tinyML Community

https://www.linkedin.com/groups/13694488/

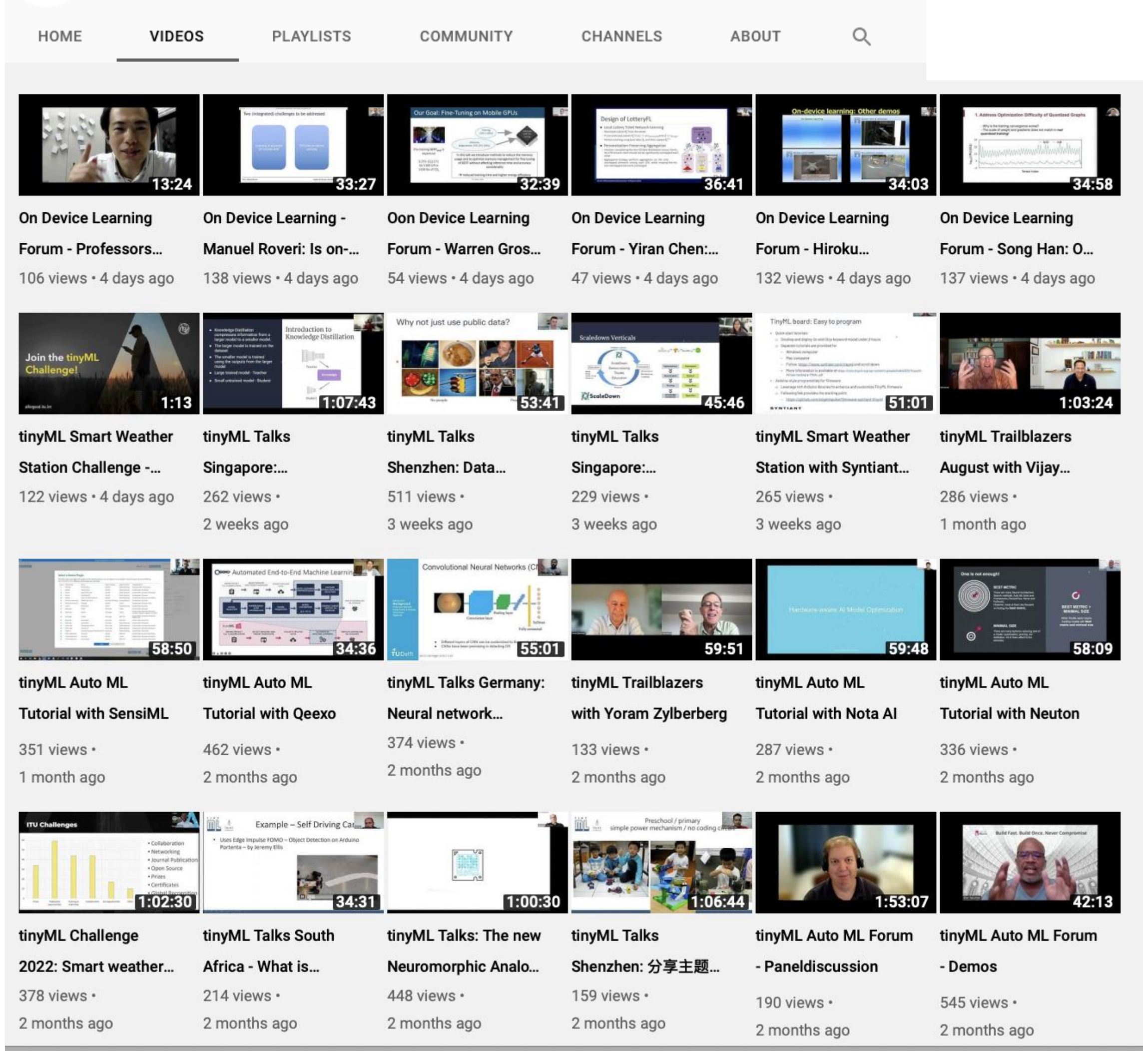
TINY ML TALKS webcast

Subscribe to
tinyML YouTube Channel
for updates and notifications
(including this video)

www.youtube.com/tinyML

YouTube

9.3k subscribers, 558 videos with 325k views



EMEA 2023

https://www.tinyml.org/event/emea-2023

More sponsorships are available: sponsorships@tinyML.org

Reminders

Slides & Videos will be posted tomorrow

tinyml.org/forums youtube.com/tinyml

Please use the Q&A window for your questions

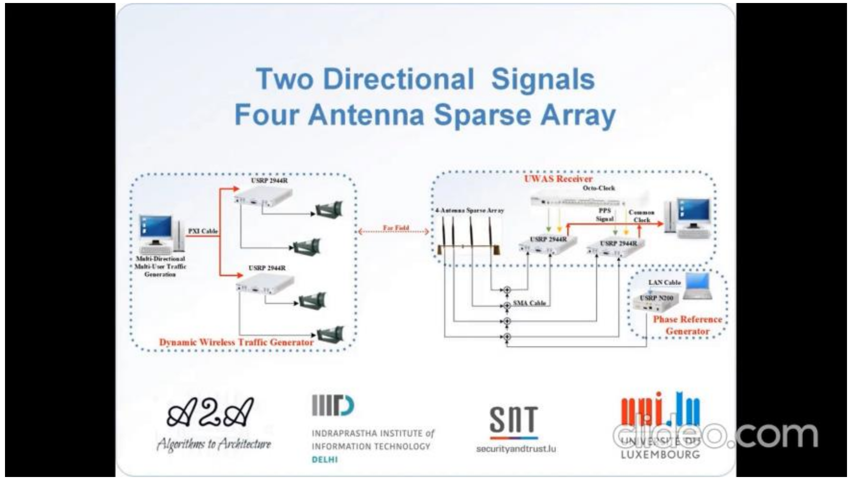
Sumit J Darak

Dr. Sumit J Darak received an Engineering degree from Pune University, India, and PhD from NTU, Singapore, in 2007 and 2013, respectively. He is an Associate Professor with IIIT-Delhi, and SoC Consultant with Apexplus Technologies, Hyderabad, India.

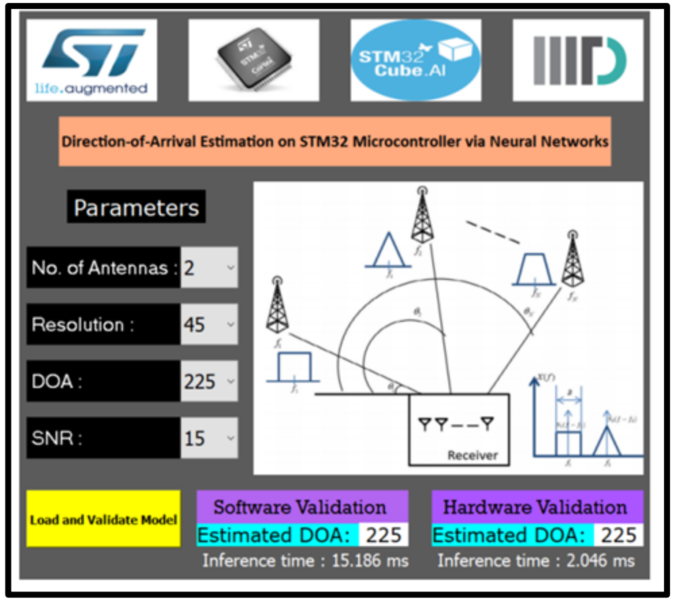
His research interests include the design of efficient synthesizable algorithms for wireless, radar, and artificial intelligence (AI) applications and mapping to reconfigurable and intelligent architectures.

Algorithms to Architecture Lab, IIIT Delhi

TINY ML TALKS webcast

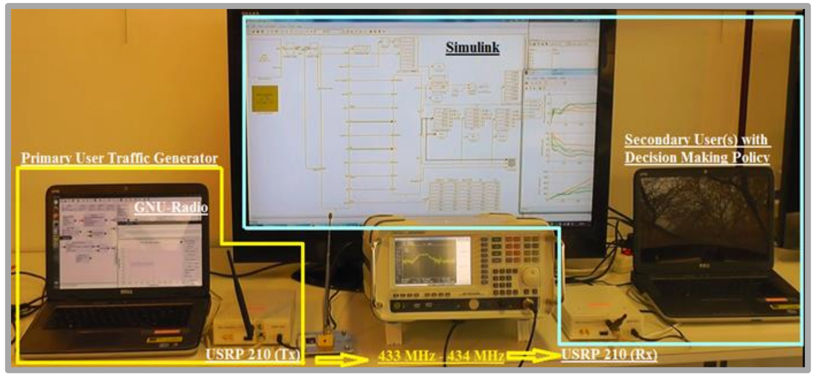


COMSNETS 2022 Best Thesis Award
IIIT Delhi 2022 Best Thesis Award

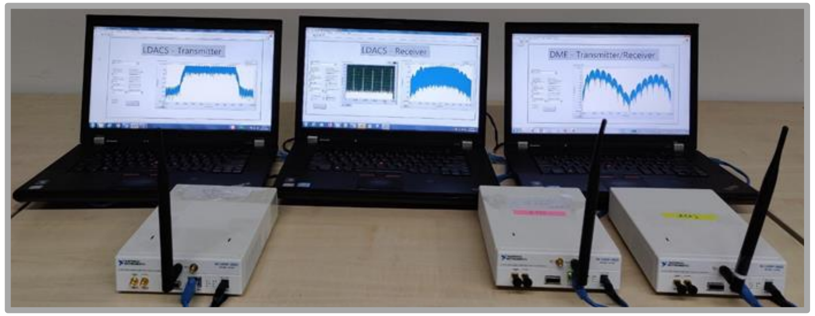


Best Paper Award in AIML Systems 2021

CloudLab: Remote Hardware Access



Distributed Learning in Wireless Networks: Best Demo Award at CROWNCOM 2016



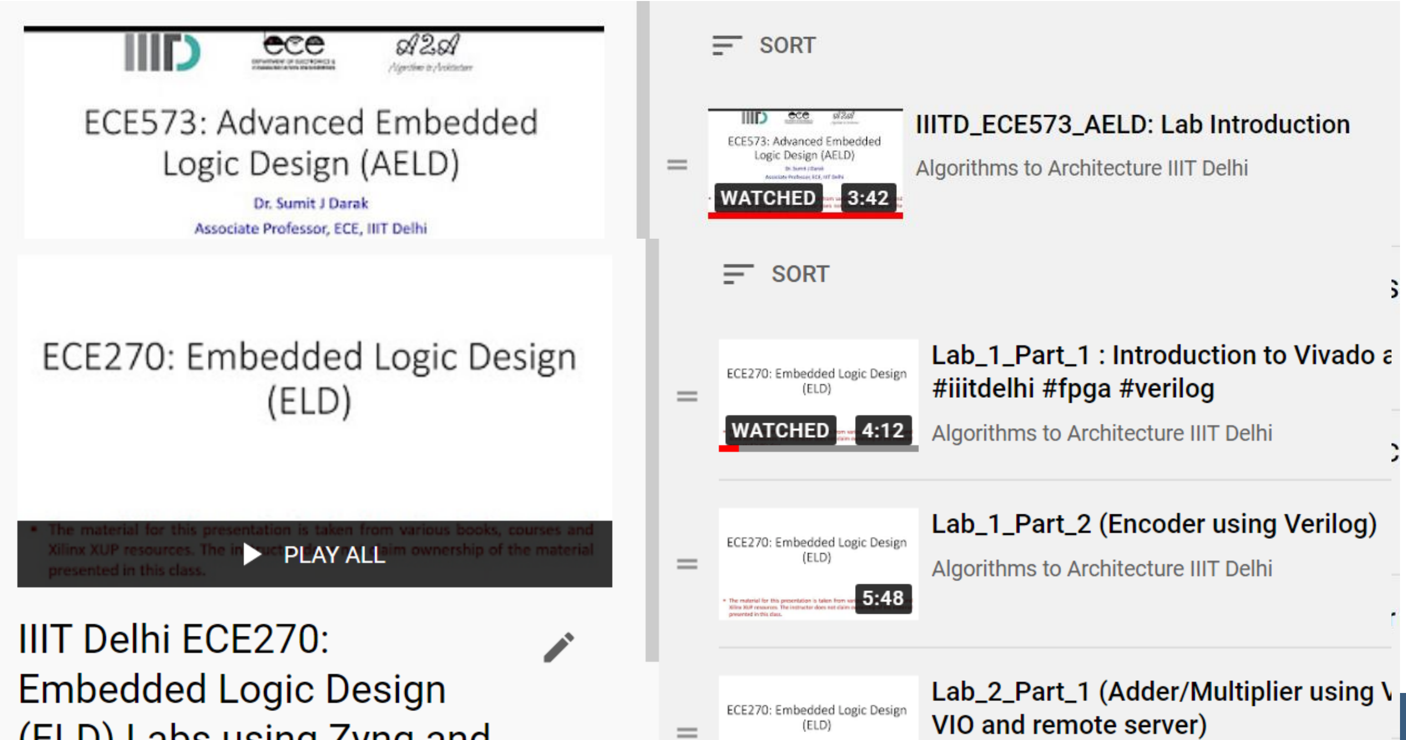
Air-to-Ground Communication in L Band: Second Best Paper Award, IEEE DASC 2017

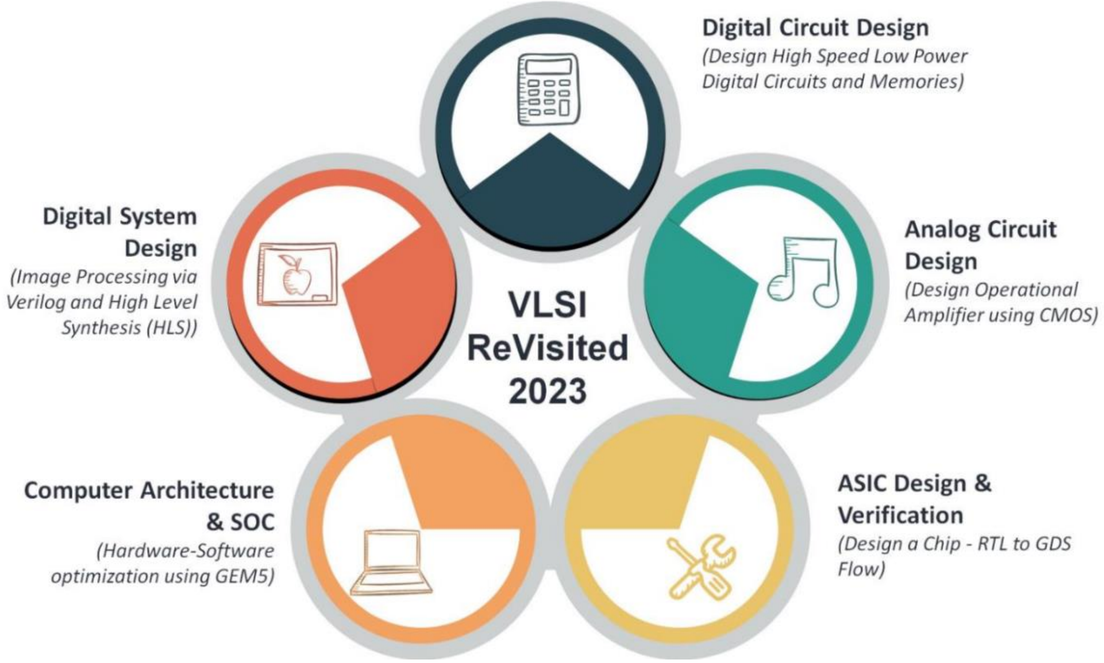
Other Awards: VLSID 2023 Design Contest Runner-up, Qualcomm Innovation Fellowship (2022), VLSID 2022 Design Contest Winner, 2021 IIITD Research Excellence Award, Second-Best Poster Award in COMSNETS 2019, Young Scientist Paper Award in URSI 2017, National Instruments (NI) Academic Research Grant (2017, 2018)

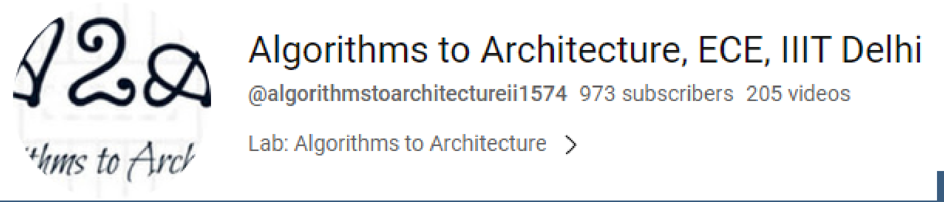
Algorithms to Architecture Lab, IIIT Delhi

- 14-Day Summer school on FPGA Design Flow
- Video game design using Verilog
- July 6-July 20, 2023 (Offline)
- Contact: sumit@iiitd.ac.in

- **VLSI ReVisited: from Analog to Digital**
- **July 3 – July 28, 2023 (online)**







Outline

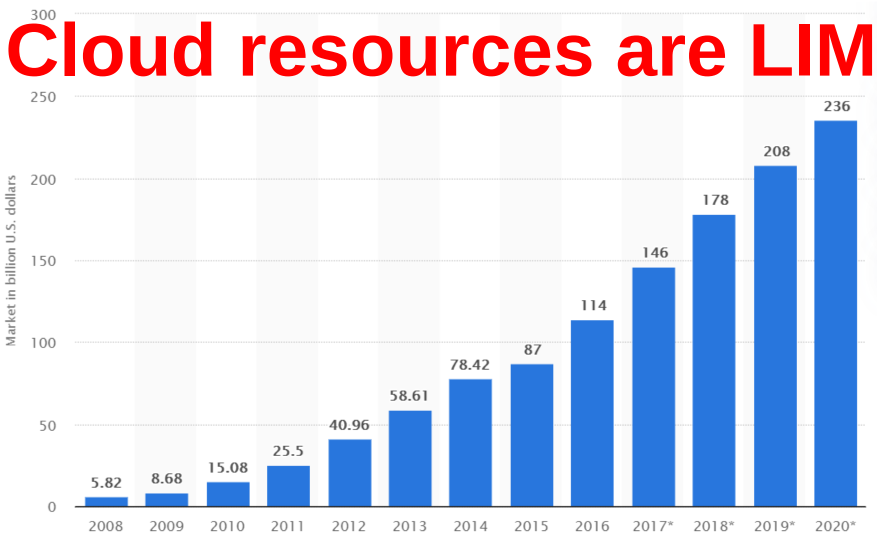
- Brief Introduction
  - ✓ Edge Computing
  - ✓ Hardware Software Co-design
  - ✓ Intelligent and Reconfigurable Architecture
- Multi-Armed Bandit Algorithms and Architectures
- Discussion: Results and Future Works

- Work credits: S. V. Sai Santosh (Research Intern, IIIT Delhi)
- Detailed handouts and source codes: https://github.com/Sai-Santosh-99
- Video: https://youtu.be/1WOpdyr7cTU
- Publications: IEEE ISCAS 2020, IEEE TCAS-II, IEEE TII, IEEE OJCAS, and IEEE TNNLS

Cloud Computing Market

TINY ML TALKS webcast

Cloud resources are LIMITED



https://www.statista.com/statistics/510350/worldwide-public-cloud-computing/

- Other than privacy, when the data is transferred outside your network, it is always susceptible to cyber attacks.
- Data centres are NOT green
- Edge Computing is one alternative

LISTEN UP Amazon workers listen to Alexa recordings – change these three settings NOW

Sean Keach, Digital Technology and Science Editor
20 Feb 2020, 16:04 | Updated: 21 Feb 2020, 15:39

TECHNOLOGY

Alexa can hear and record what is typed in nearby devices, study finds

Aditya Saroha

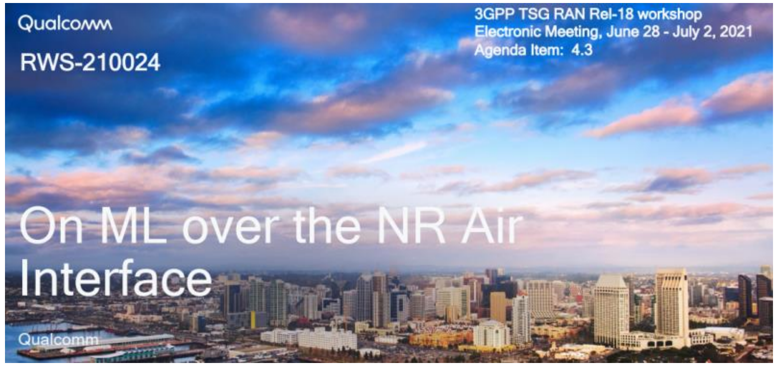
DECEMBER 28, 2020 14:02 IST
UPDATED: DECEMBER 28, 2020 14:02 IST

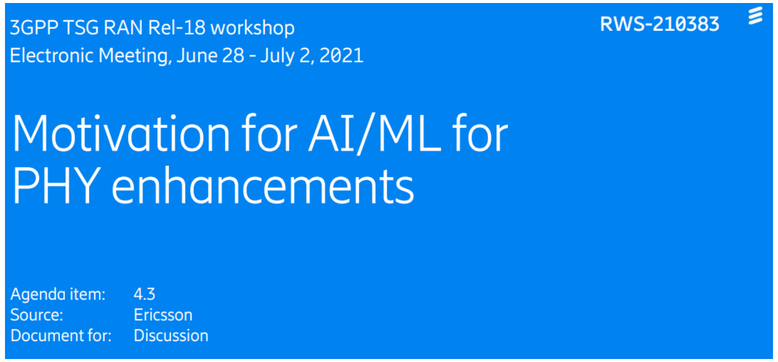
SHARE ARTICLE | PRINT | A | A | A

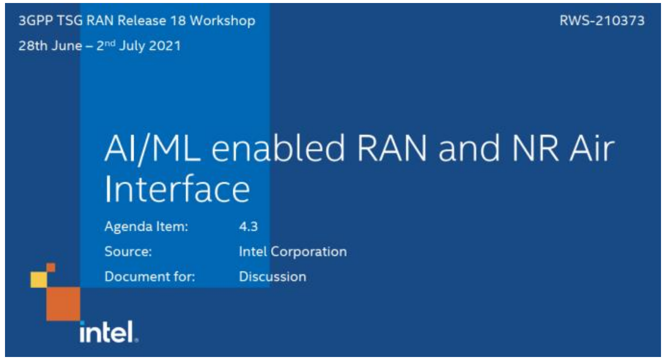
IoT Devices
WFH
Industry 4.0

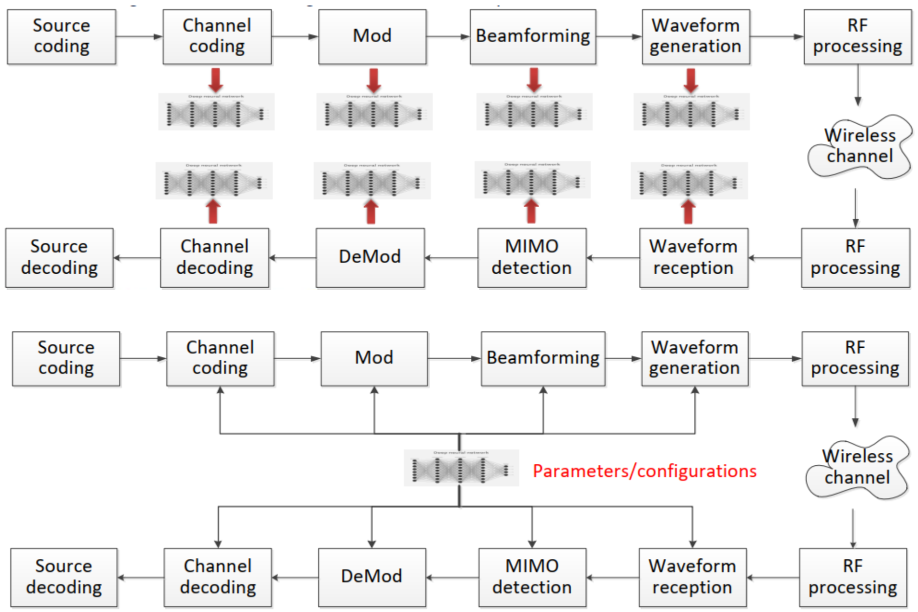
https://www.independent.co.uk/life-style/gadgets-and-tech/news/google-home-recordings-listen-privacy-amazon-alexa-hack-a9002096.html

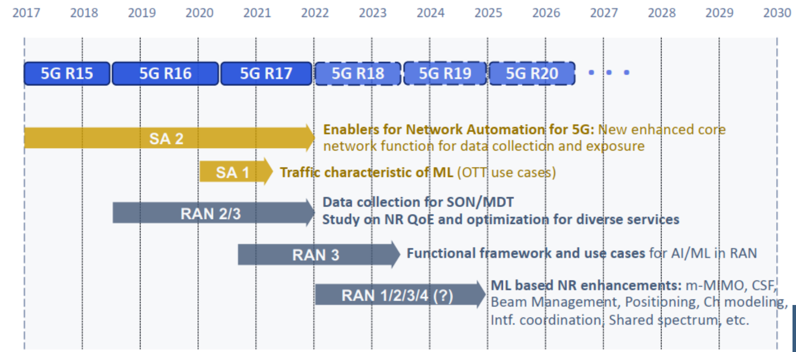
Edge Computing for AI/ML



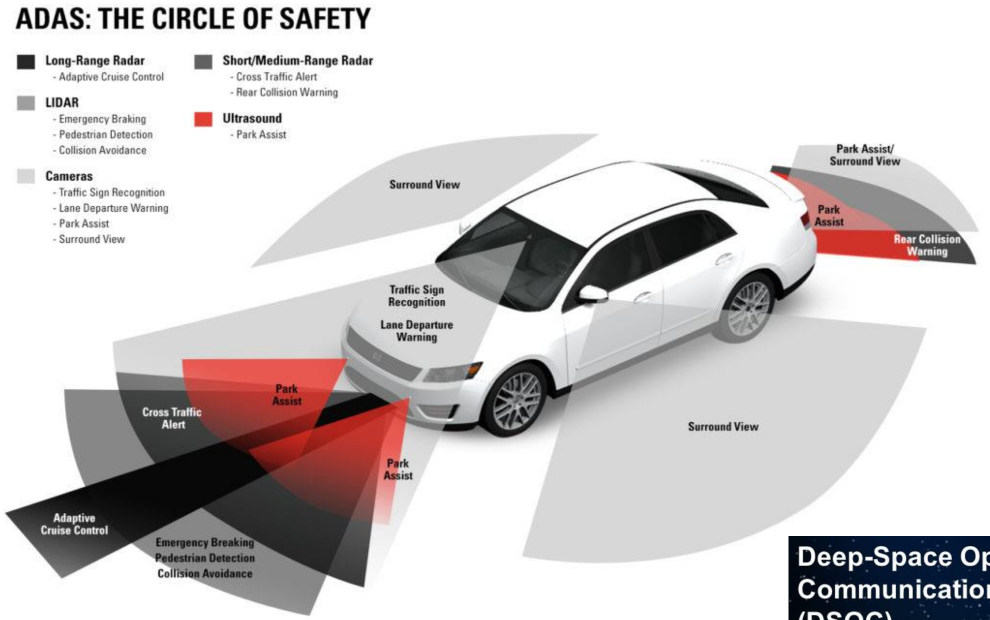


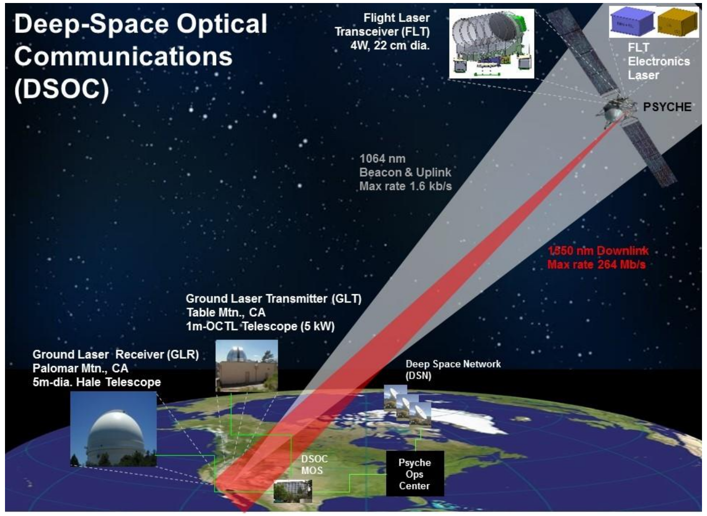






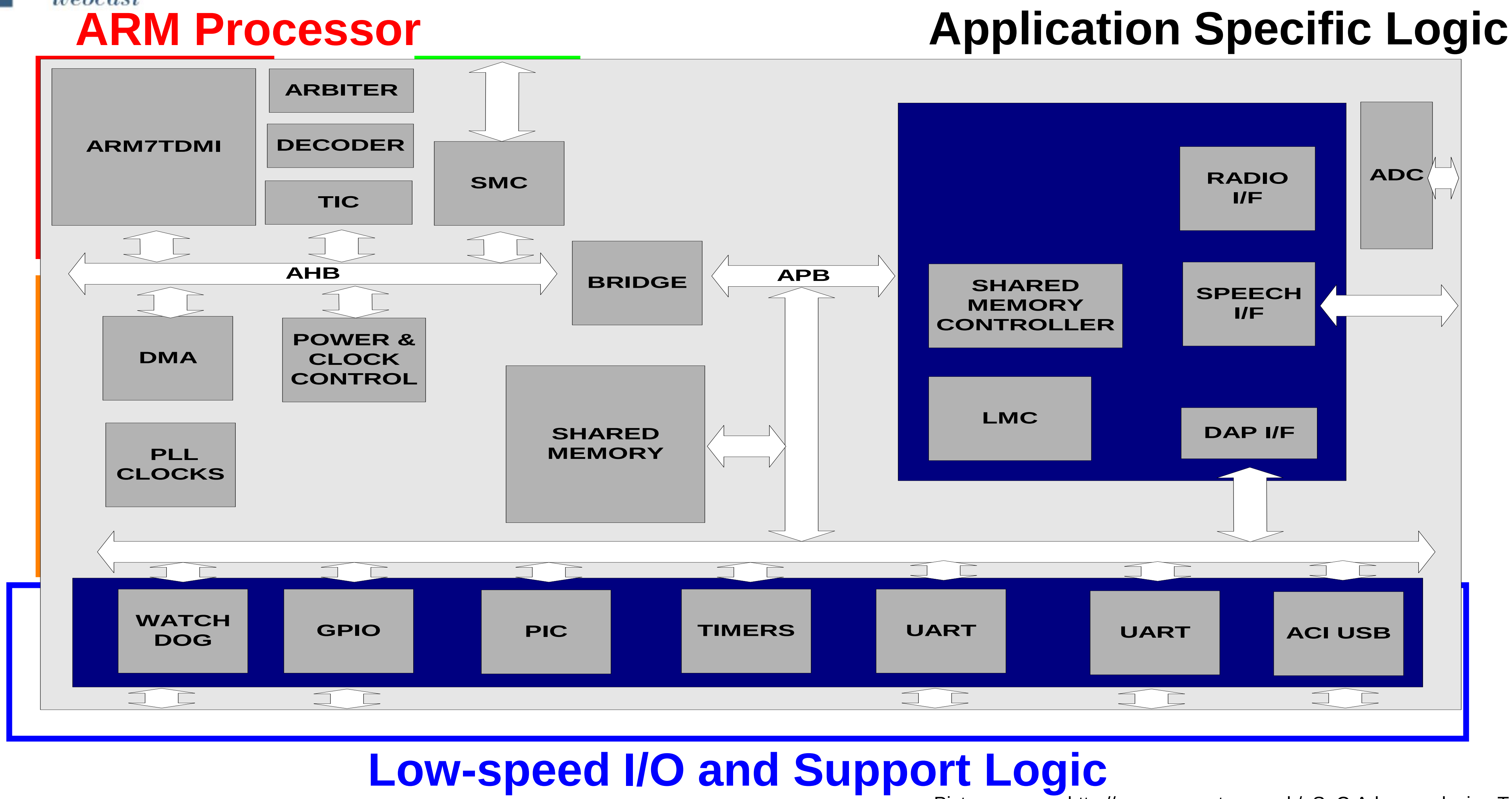
Edge Computing for AI/ML





Edge Platforms

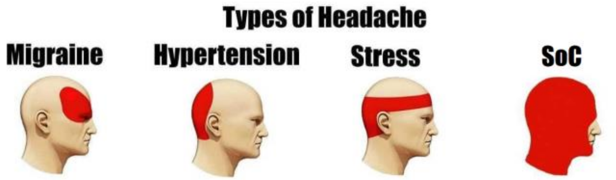
Bluetooth System-on-Chip (SoC)

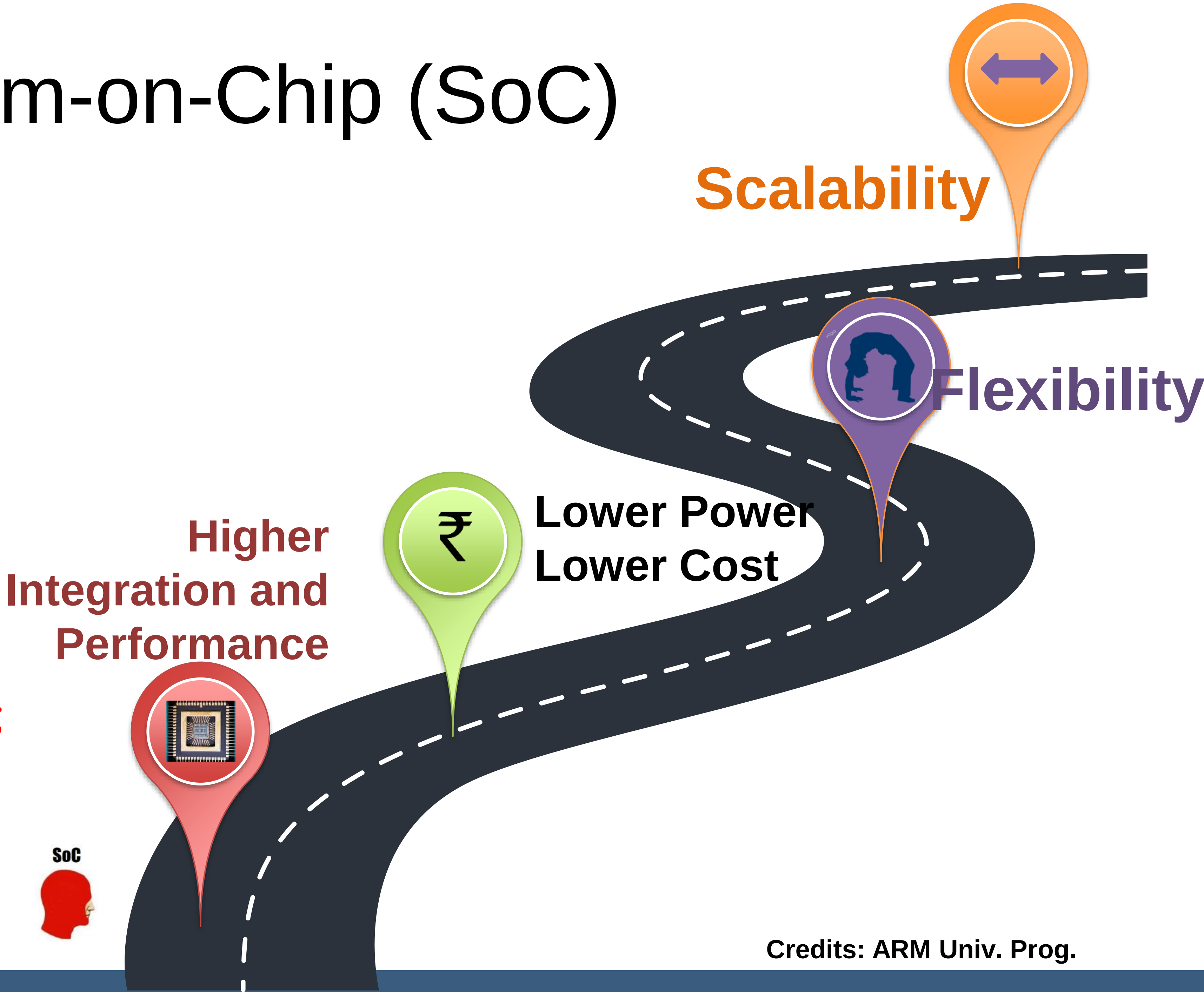


Picture source: http://www.ecs.soton.ac.uk/, SoC Advance design Technique

System-on-Chip (SoC)

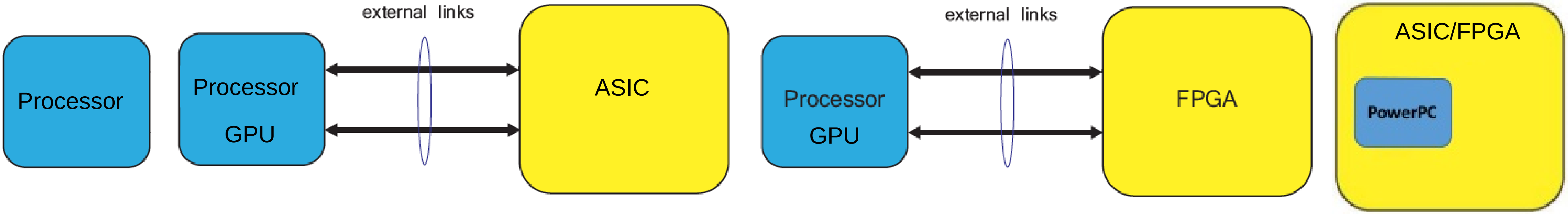
- **Advantages:**
  - **Higher performance and high Power efficiency**
  - **Lighter footprint**
  - **Higher reliability**
  - **Low cost**
- **Challenges:** **Application Specific, Less flexibility, High design complexity**
- **Expectations from upcoming applications:** **Scalability and Flexibility**

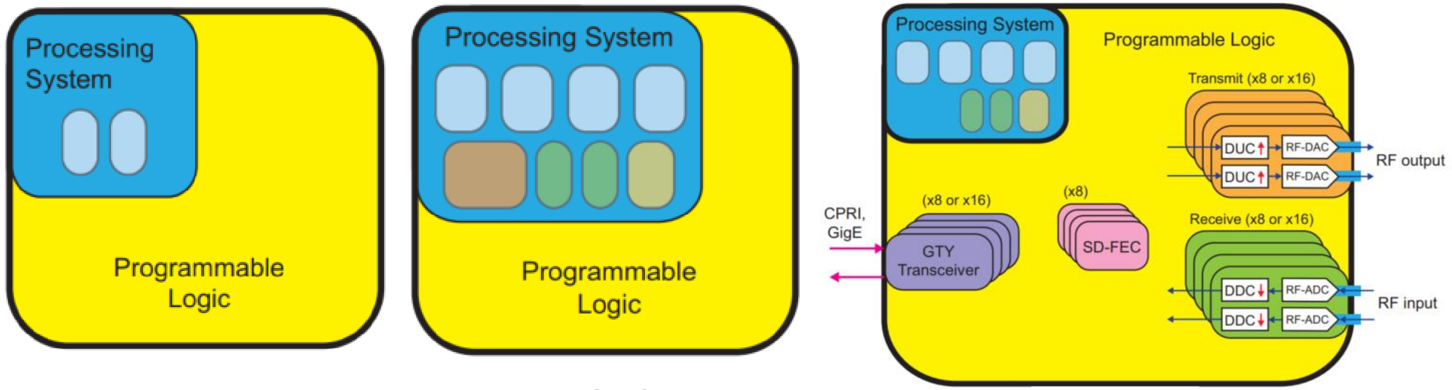




Credits: ARM Univ. Prog.

Edge Platforms

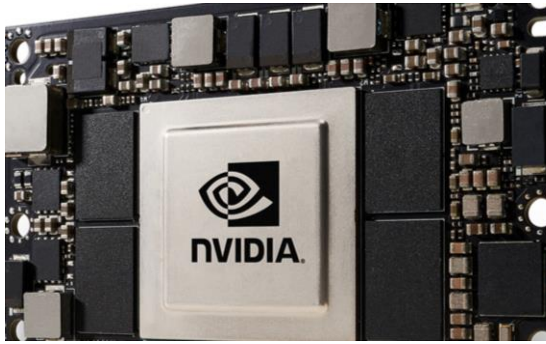


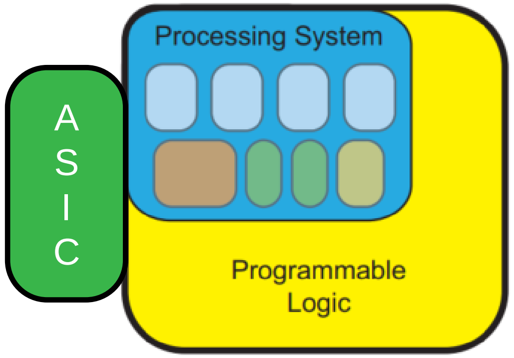


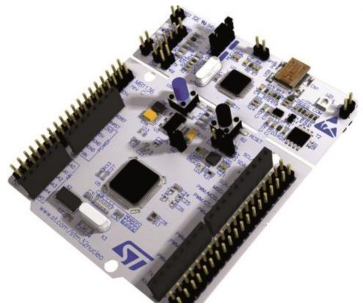
APSOC

MPSoC

RFSOC



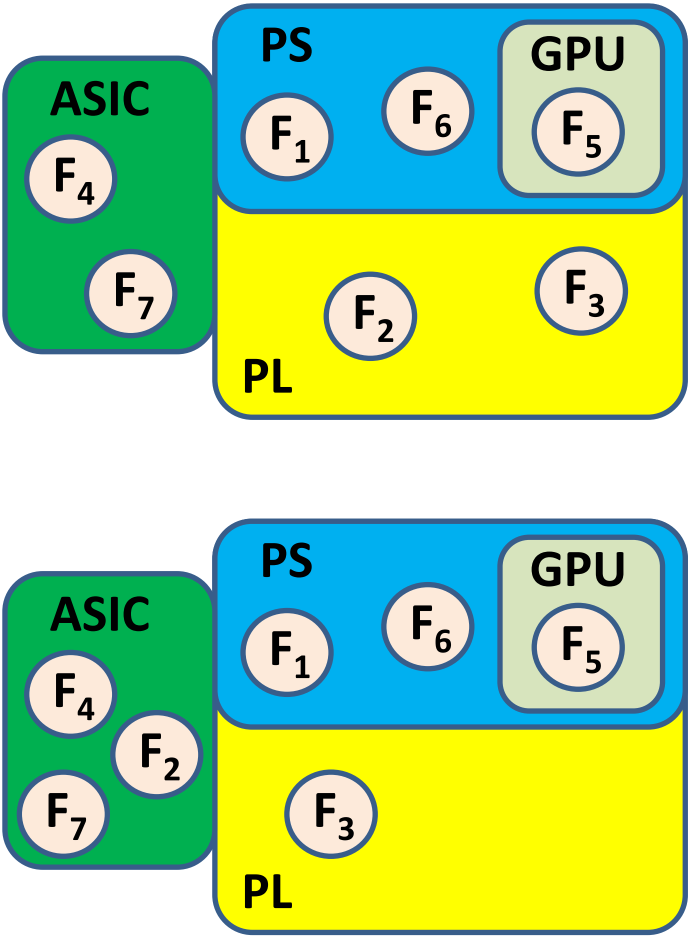


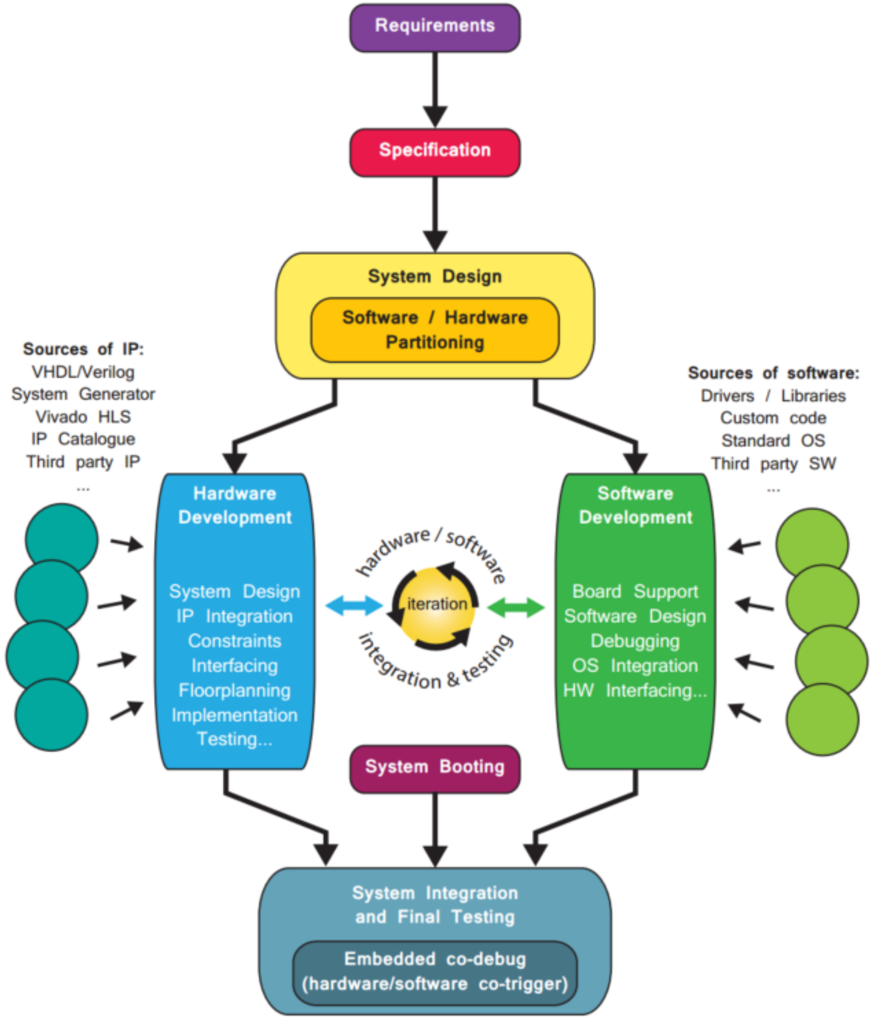


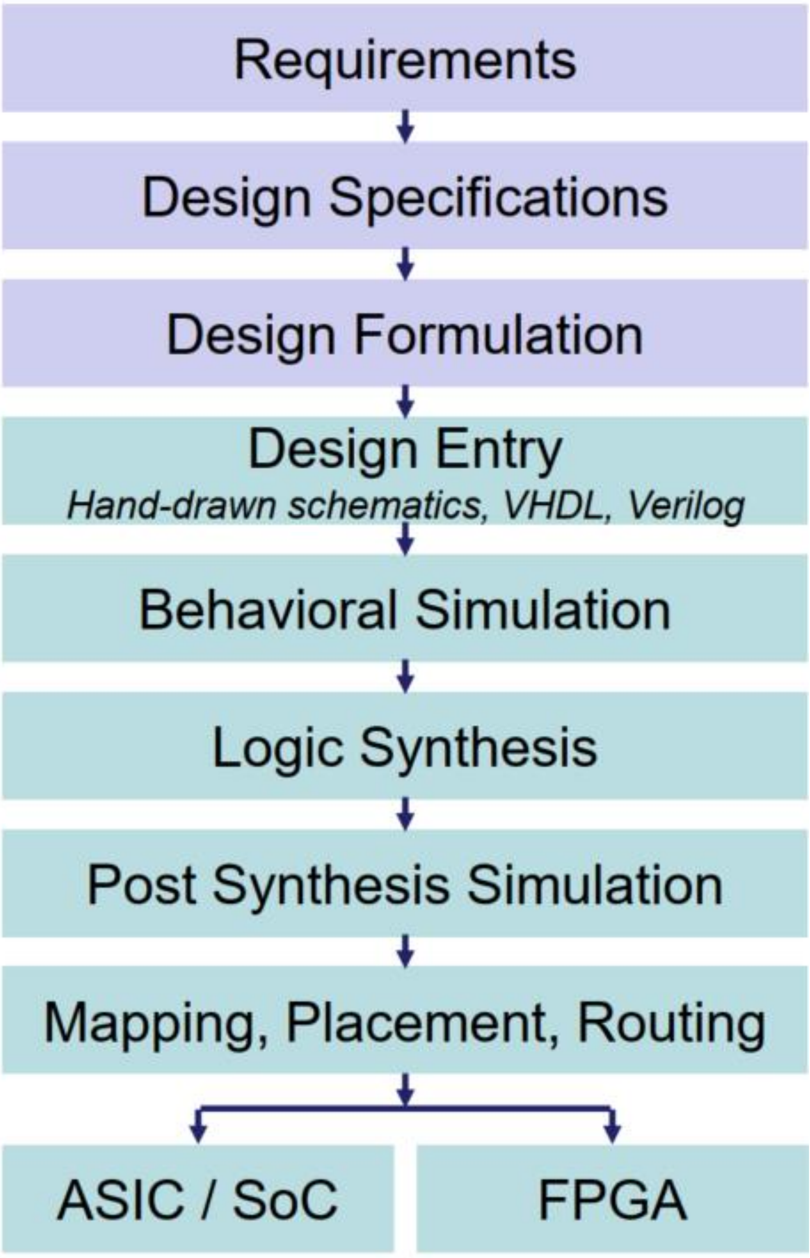
Heterogenous All Programmable SoC

Zynq Book by Xilinx

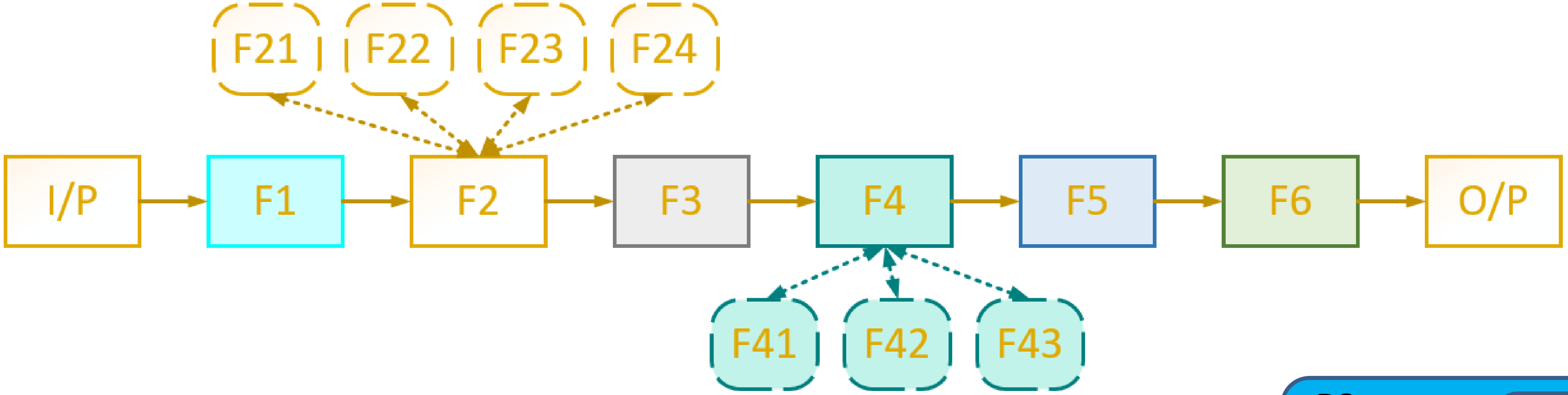
Hardware Software Co-design (HSCD)



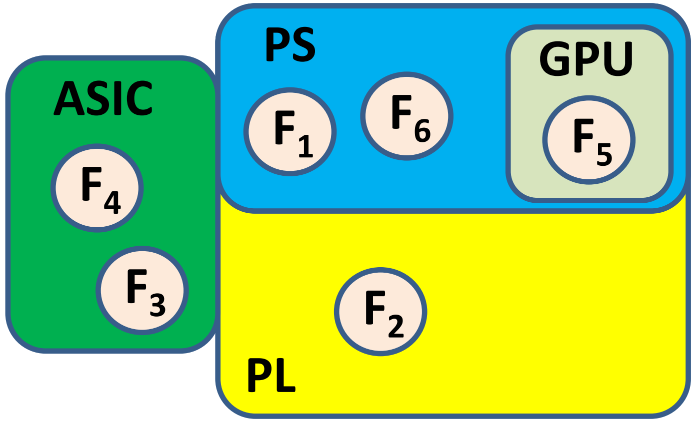




Reconfigurable HSCD

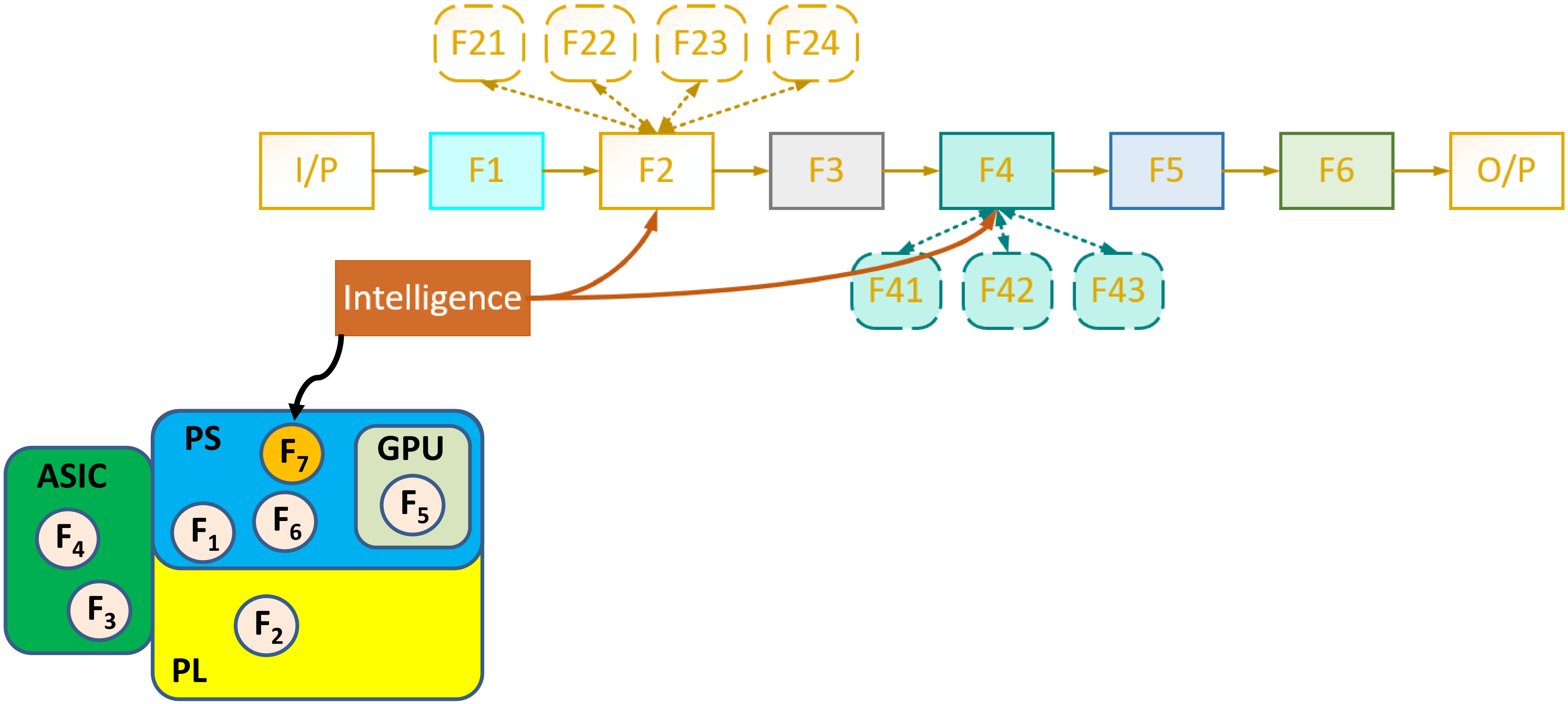


- Lower power consumption due to fewer resources
- Larger and complex design can be efficiently mapped on smaller FPGAs
- Direct cost and operation cost savings since smaller and cheaper FPGA device is needed
- Feature richness and Upgradability



Support for Reconfigurable/Adaptive architecture ???

Intelligent Reconfigurable HSCD



Multi-Armed Bandit Algorithms

Introduction: MAB Algorithms

- Online learning algorithms with good analytical tractability
- Each arm gives different mean rewards/rate
- **Goal:** Maximize the expected sum-rate
- Exploration-exploitation trade-off
- **Applications:** Website advertise, Wireless networks, Neural networks, Healthcare
- **Simple strategy:** User selects the "*top arm*."
- But arm statistics are unknown 😐

- A. Slivkins, "Introduction to multi-armed bandits," in Foundations and Trends in Machine Learning, vol. 12, no. 1, 2019, pp. 1–286.
- P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," Machine Learning, vol. 47, no. 2, 2002.

Introduction: MAB Algorithms

- **Simple Strategy:** User selects the "*top arm*." But arm statistics are unknown
- The reward of an arm i is stochastic with distribution v_i and mean μ_i
- Rewards are IID across time and arms
- Using policy π, algorithm select the arm π_t is time slot t.
- Performance metric: Regret

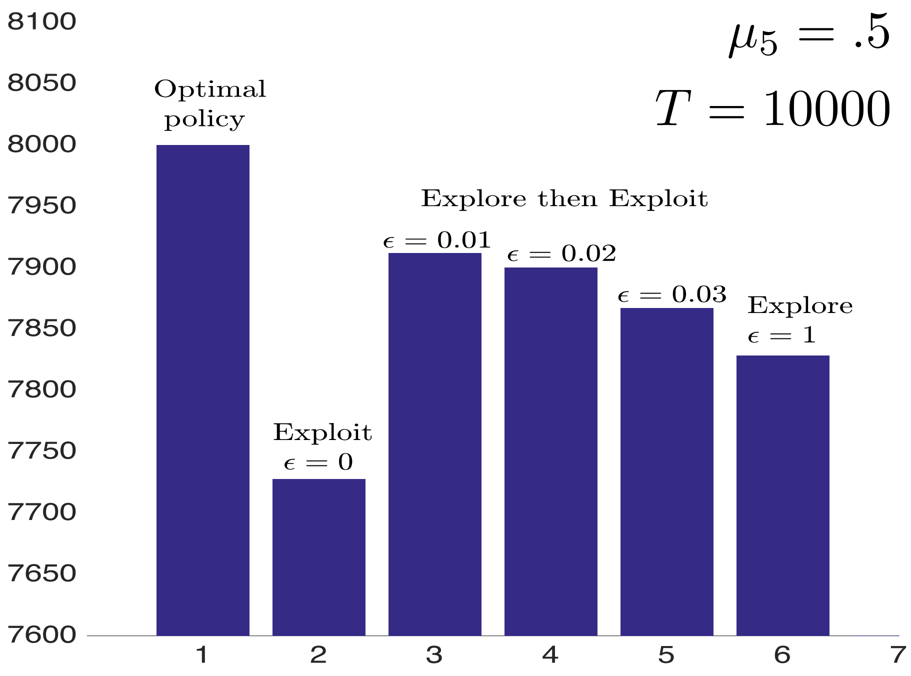
$$R_T(\pi) = T \max_{i \in [N]} \mu_i - \mathbb{E}\left[\sum_{t=1}^{T} \mu_{\pi_t}\right]$$

Introduction: MAB Algorithms

- **Policy 1:** Select each arm T/K times (or select arms randomly)
  - ➢ Explore each arm equally
- **Policy 2:** Choose the empirical best arm in each round
  - ➢ Exploit the best arm
- **Policy 3:** Explore for the first ϵT time slots and then exploit
  - ➢ Explore then Exploit
- Too much exploration or ‘premature’ exploitation not good
  - ➢ If exploration high, suboptimal arms selected often
  - ➢ Premature exploitation may miss optimal arm

$K = 5$ actions with means

$$\mu_1 = 0.3$$
$$\mu_2 = .35$$
$$\mu_3 = .78$$
$$\mu_4 = .8$$
$$\mu_5 = .5$$
$$T = 10000$$



How to find optimal ϵ?

Introduction: MAB Algorithms

$\mu^* = \max_{i \in [K]} \mu_i$ $\Delta_i = \mu^* - \mu_i$ $\mu = (\mu_1, \mu_2, \ldots, \mu_K)$

$i^* = \arg\max \mu_i$ $\Delta = \min_{i \neq i^*} \Delta_i$

Lower Bound [Lai & Robins 1985]: For any 'uniformly efficient' policy π on μ

$$R_T(\pi) \gtrapprox \sum_{i \neq i^*} \Delta_i \frac{\log T}{d(u_i, u^*)}$$

where $d(\mu_i, \mu^*)$ is KL divergence given by

$$d(\mu_i, \mu^*) = \mu_i \log \frac{\mu_i}{\mu^*} + (1 - \mu_i) \log \frac{(1 - \mu_i)}{(1 - \mu^*)}$$

Problem independent bound [Cesa-Bianchi and Lugosi 2006]: For any policy π, there exits a distribution of on arm rewards such that

$$R_T(\pi) \geq \frac{1}{20} \sqrt{KT}$$

Optimism in the Face of Uncertainty (OFU)

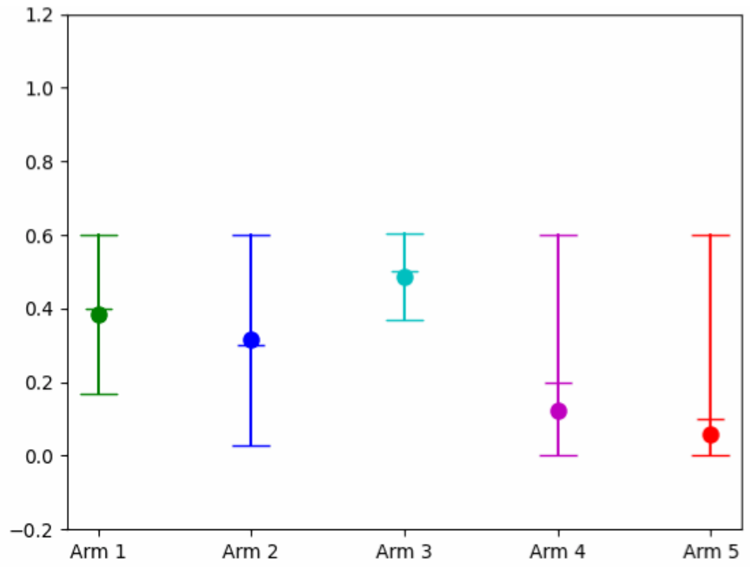
- For each arm i build a confidence on its mean estimate

$$\mathcal{I}_i(t) = [LCB_i(t), UCB_i(t)]$$

LCB: Lower Confidence Bound
UCB: Upper Confidence Bound

- Take the current upper bound on the estimate as the true value (optimism)

$$\pi_t = \arg\max_{i \in [K]} UCB_i(t)$$



Upper Confidence Bound (UCB)

$$UCB_i(t) = \hat{\mu}_i(t) + \sqrt{\frac{\alpha \log T}{T_i(t)}}$$

1. **Input** $K, T, \alpha \geq 2$
2. Play each arm once
3. For $t = K+1, K+2, \ldots t$
4. Play $\pi_t = UCB_i(t-1)$ and observe reward
5. Update pull count and mean estimate of π_t
6. End

- Problem dependent: (bounded support) $R_T(\text{UCB1}) \leq \mathcal{O}(\sum_{i \neq i^*} \frac{\log T}{\Delta_i})$
- Problem independent: $R_T(\text{UCB1}) \leq \mathcal{O}(\sqrt{KT \log(T)})$

KL-UCB

$$UCB_i(t) = \max_q \left\{ d(\hat{\mu}_i(t), q) \leq \frac{\log t + 3\log(\log t)}{T_i(t)} \right\}$$

$$\mathcal{I}_i^{\text{KL-UCB}}(t) \subset \mathcal{I}_i^{\text{UCB1}}(t)$$

- Problem dependent: (bounded support)
  $R_T(\text{KL-UCB}) \leq \mathcal{O}(\sum_{i \neq i^*} \Delta_i \frac{\log T}{d(\mu_i, \mu^*)})$ (optimal!)
- Problem independent:
  $R_T(\text{KL-UCB}) \leq \mathcal{O}(\sqrt{KT \log(T)})$

MAB: Frequentist or Bayesian?

- The frequentist modeling-based UCB and KLUCB algorithms assume the mean reward of an arm is proportional to the average reward in repeated plays of a given experiment.
- Bayesian modeling-based TS algorithm assumes the mean reward of an arm is proportional to a degree of belief that the arm is optimal.
- These beliefs are updated based on the past observations via Baye’s rule that takes a prior belief as an argument and returns a posterior belief for a given likelihood.

MAB: Frequentist or Bayesian?

- Since the arm statistics are unknown, the uncertainty about arm optimality is modeled as probabilities and the arm with the highest probability of being optimal under the posterior distribution is selected.
- In the MAB, posterior belief becomes a prior in subsequent time slots, and the distributions which exhibit such behavior are known as conjugate prior.
- Beta distribution is a conjugate prior for Bernoulli likelihood function.

$$\nu_i \left(\mu_i | X_{i,1}, X_{i,2}, \ldots, X_{i,T_i(t)}\right) = Beta \left(\sum_{s=1}^{T_i(t)} X_{i,s} \quad , \quad T_i(t) - \sum_{s=1}^{T_i(t)} X_{i,s} \right)$$

Thompson Sampling

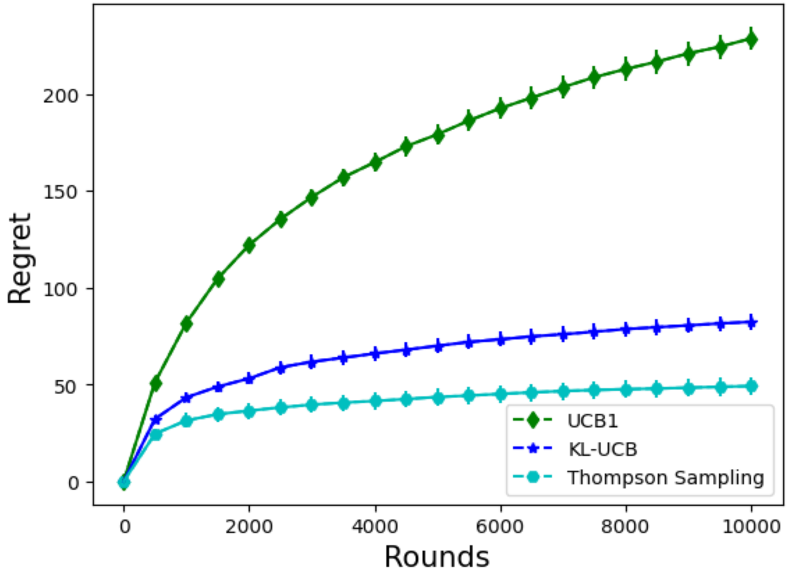
$$\nu_i\left(\mu_i \mid X_{i,1}, X_{i,2}, \ldots, X_{i,T_i(t)}\right) = Beta\left(\underbrace{\sum_{s=1}^{T_i(t)} X_{i,s}}_{\text{No. of success}}\ ,\ \underbrace{T_i(t) - \sum_{s=1}^{T_i(t)} X_{i,s}}_{\text{No. of failures}}\right)$$

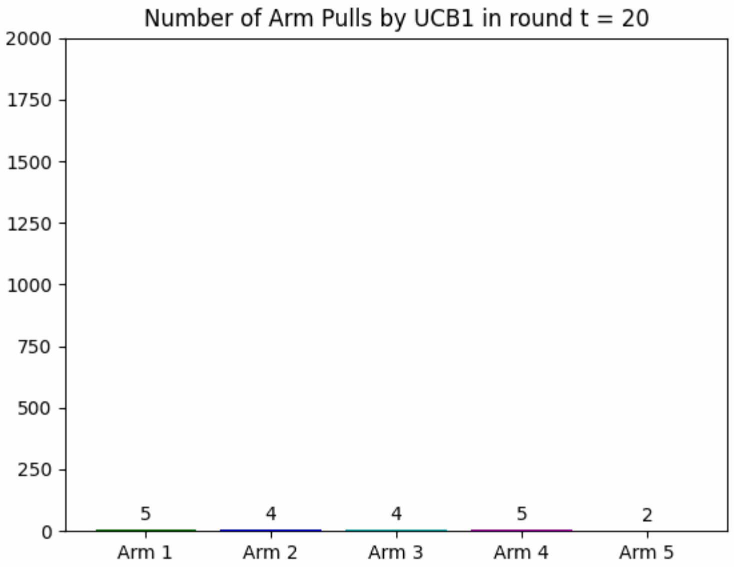
- **Input:** K, T
- Set $S_i = 0, F_i = 0\ \ \forall i \in [K]$
- For $t = 1, 2, \ldots, T$
- For each $i \in [K]$, draw $\hat{\mu}_i(t) \sim Beta(S_i + 1, F_i + 1)$
- Play arm $\pi_t = \arg\max_{i \in [K]} \hat{\mu}_i(t)$ and observer $X_{\pi_t,t}$
- Update $S_{\pi_t} = S_{\pi_t} + X_{\pi_t,t}$ and $F_{\pi_t} = F_{\pi_t} + 1 - X_{\pi_t,t}$
- End

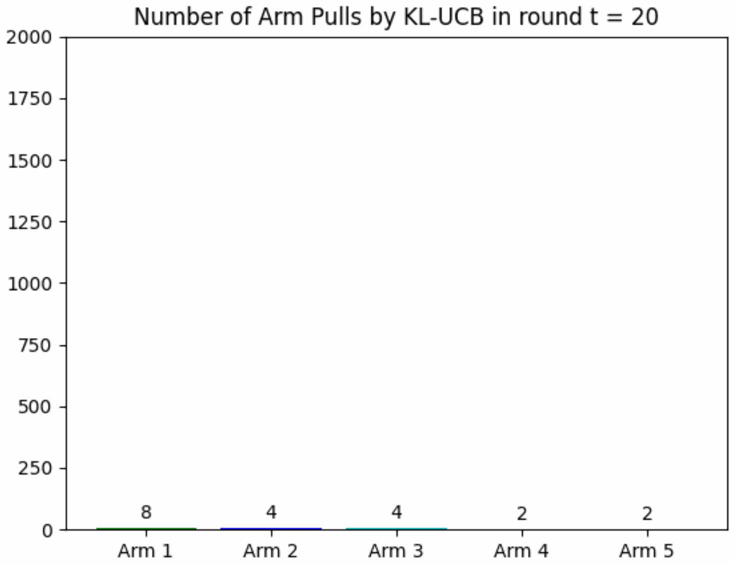
- Problem dependent: (bounded support) $R_T(\text{TS}) \leq \mathcal{O}(\sum_{i \neq i^*} \Delta_i \frac{\log T}{d(\mu_i, \mu^*)})$
- Problem independent: $R_T(\text{TS}) \leq \mathcal{O}(\sqrt{KT \log(T)})$

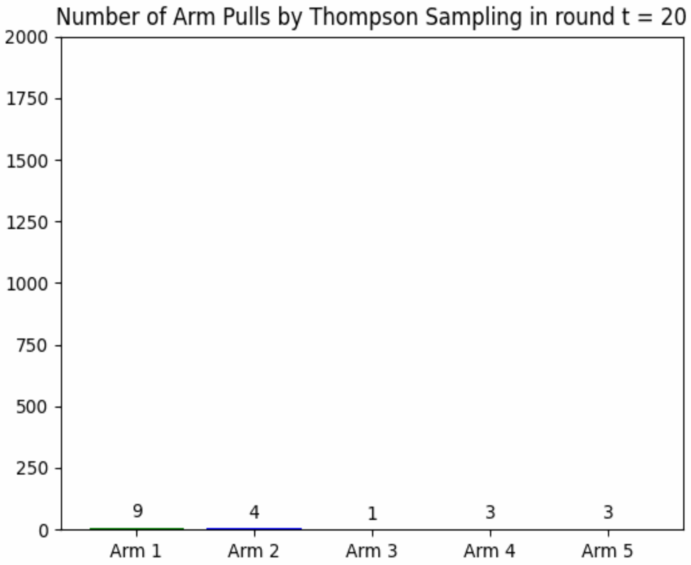
MAB Algorithms

$$\mu = (0.4, 0.3, 0.5, 0.2, 0.1)$$

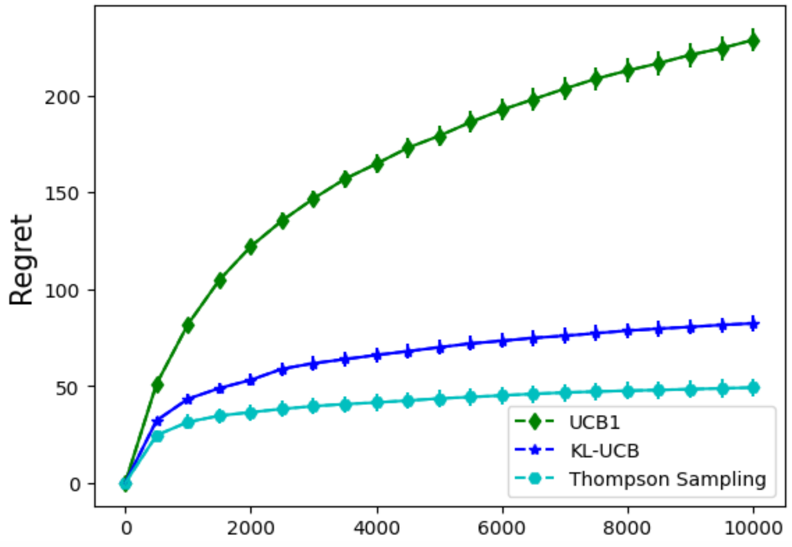




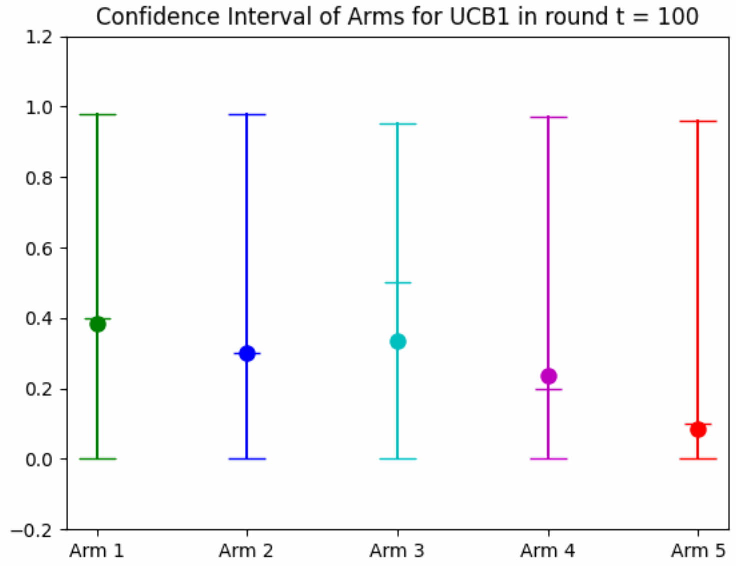


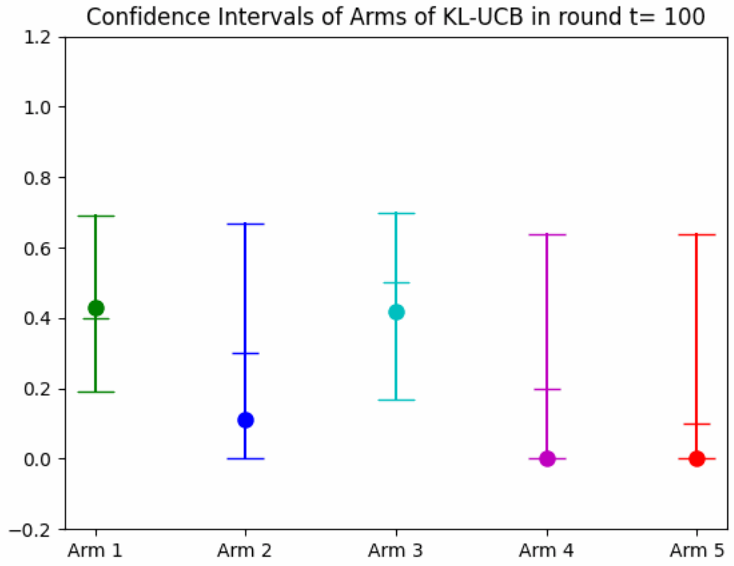


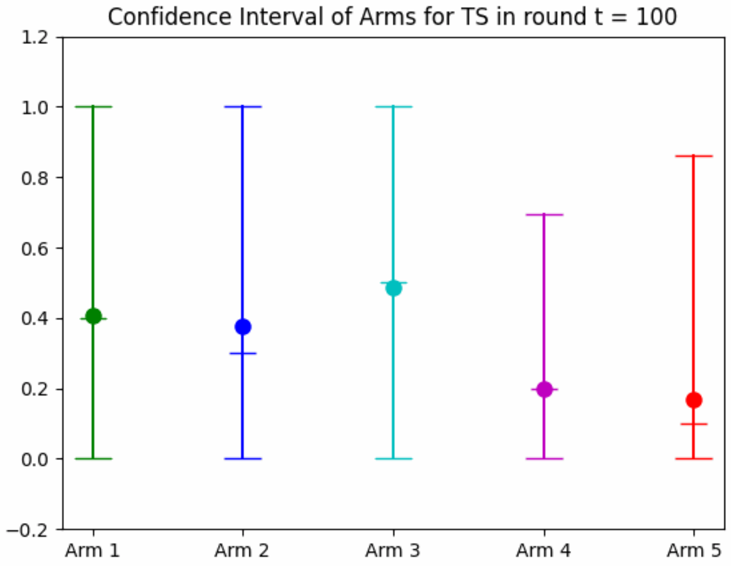
MAB Algorithms



$$\mu = (0.4, 0.3, 0.5, 0.2, 0.1)$$



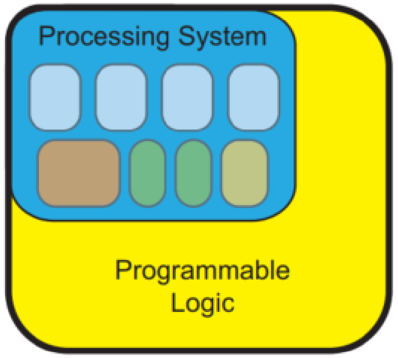




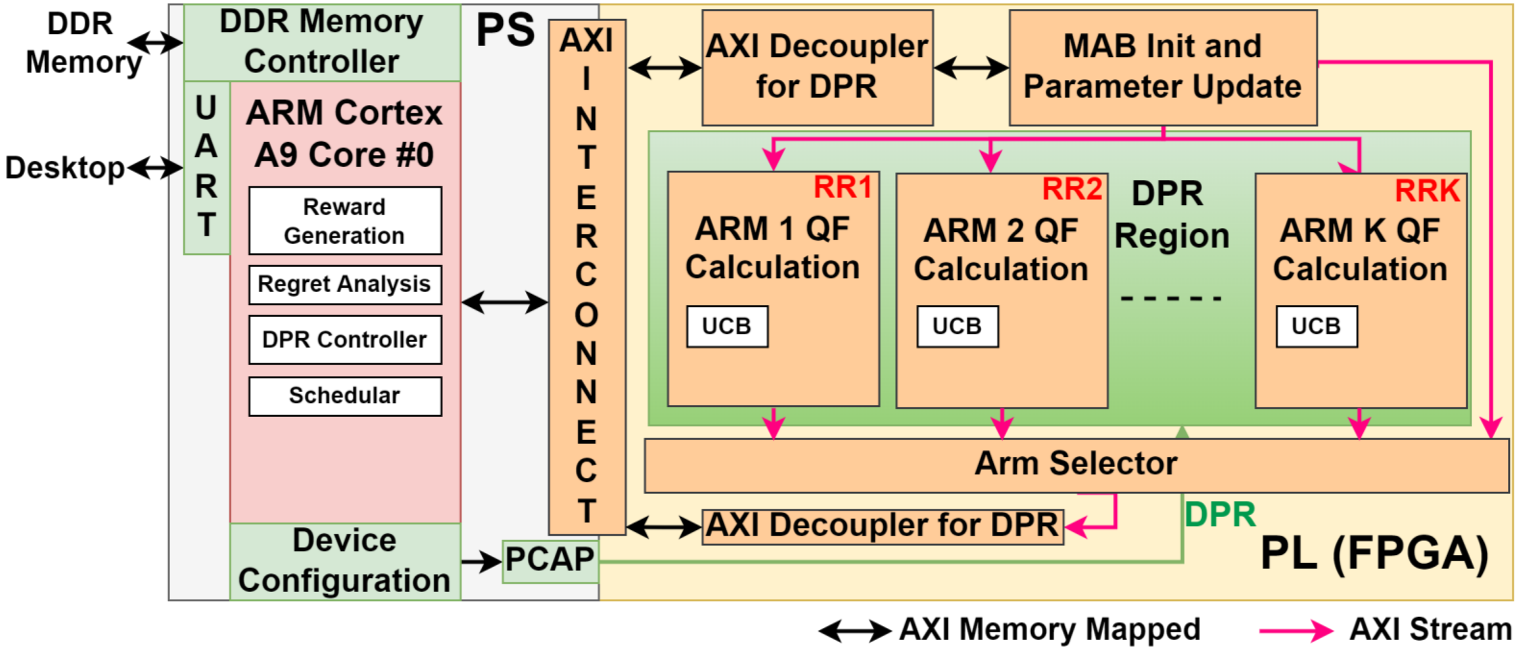
MAB Algorithms

Algorithms	Advantages	Disadvantages
UCB	1. Easy to implement 2. Distribution independent	1. Has a tuning parameter
KL-UCB	1. Asymptotically optimal 2. Good empirical performance 3. Distribution independent	1. Computational intensive (need to solve a convex problem in each round) 2. Hard to implement (in Hardware)
Thompson Sampling	1. Asymptotically optimal 2. No tuning parameter 3. Good Empirical performance	1. Distribution dependent 2. Hard to implement in hardware

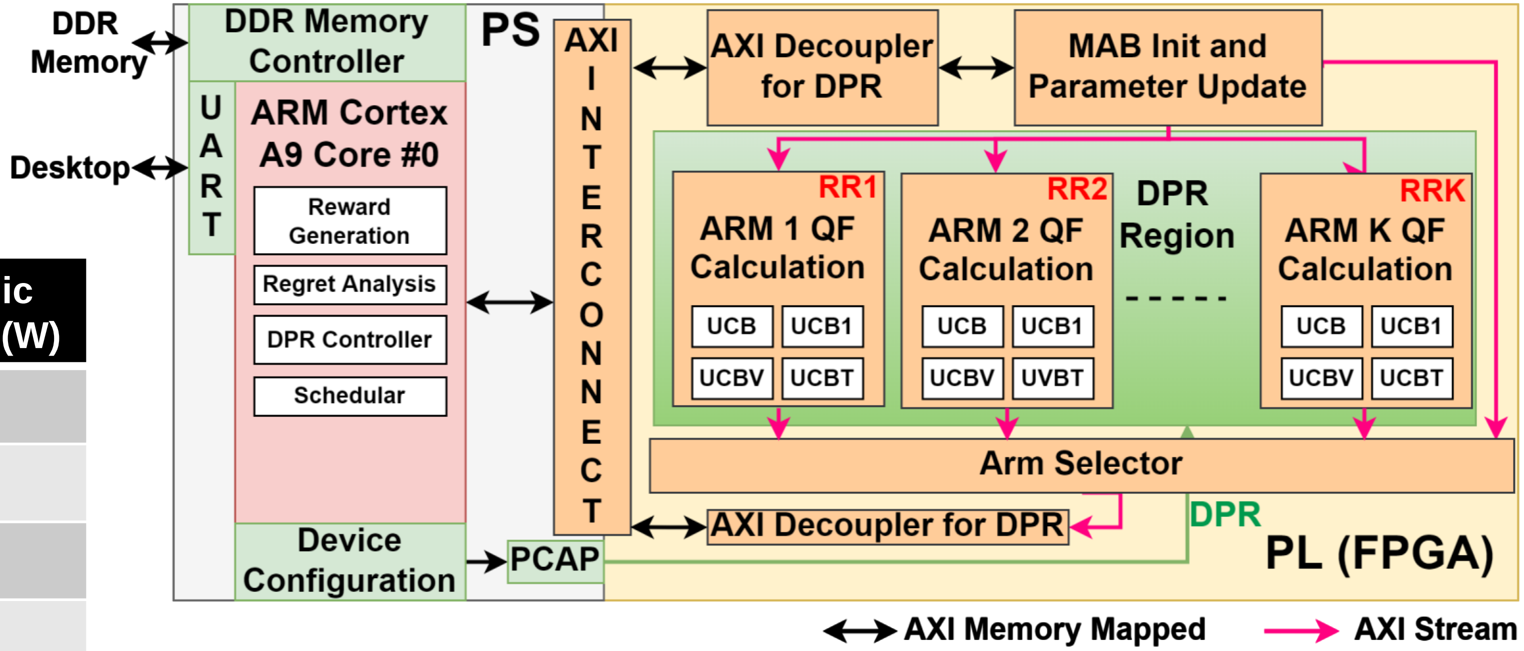
UCB Architecture on SoC



Zynq MPSoC



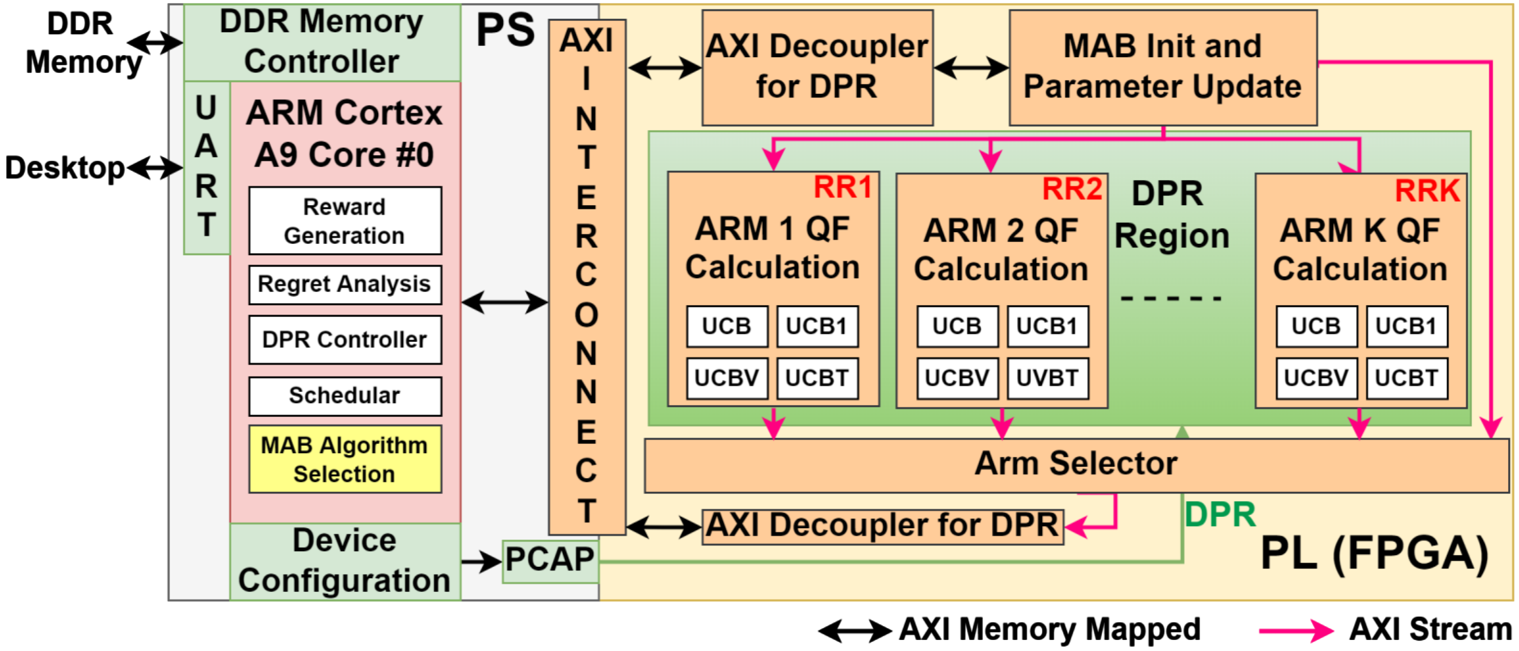
UCB Reconfigurable Architecture



	LUTs	FFs	DSP	BRAM	Dynamic Power (W)
UCB (K=5)	12068	11126	85	7.5	0.2
UCBV (K=5)	16655	14389	98	9	0.22
UCBT (K=5)	17843	13856	112	10	0.23
Velcro	69620	44928	220	25	0.4
Reconf. Architecture	19274	15962	112	11	0.234

UCB Intelligent and Reconfigurable Architecture

- Selection of appropriate UCB algorithms is not challenging since each one is used in specific application.
- For instance, UCB is used for low latency, UCBV is used when the arm with optimal mean and low variance needs to be selected and UCBT offers lower regret than UCB but incurs higher latency.



- **Why to use UCB if TS is better?**

Thompson Sampling Architecture

- Bayesian approach compared to frequentist approach based UCB algorithm
- Offers better performance than UCB but direct hardware mapping is not trivial

$$\nu_i\left(\mu_i | X_{i,1}, X_{i,2}, \ldots, X_{i,T_i(t)}\right) = Beta\left(\sum_{s=1}^{T_i(t)} X_{i,s} \quad , \quad T_i(t) - \sum_{s=1}^{T_i(t)} X_{i,s} \right)$$

- Distribution dependent (Note: Type of distribution may not known)

Thompson Sampling Architecture

- Direct mapping of Beta function on SoC does not exist and its computationally complex.
- Instead of directly mapping, we develop its approximate architecture.
- **Proposed Idea:** Generate $T_i(t)$ number of random numbers for arm i. Sort them and select the $\left(\sum_{s=1}^{T_i(t)} X_{i,s}\right)^{th}$ random number.
- For generation for random numbers, we use existing pseudo-random number generator (PRNG)

$$\nu_i\left(\mu_i | X_{i,1}, X_{i,2}, \ldots, X_{i,T_i(t)}\right) = Beta\left(\sum_{s=1}^{T_i(t)} X_{i,s} \;,\quad T_i(t) - \sum_{s=1}^{T_i(t)} X_{i,s} \right)$$

S. V. Sai Santosh, and **S. J. Darak**, “Multi-armed Bandit Algorithms on Zynq System-on-Chip: Go Frequentist or Bayesian?,” in *IEEE Transactions on Neural Networks and Learning Systems*, accepted in June 2022.

Thompson Sampling Architecture

- **Proposed Idea:** Generate $T_i(t)$ number of random numbers for arm i. Sort them and select the $\left(\sum_{s=1}^{T_i(t)} X_{i,s}\right)^{th}$ random number.
- For generation for random numbers, we can use existing pseudo-random number generator (PRNG)
- **Drawbacks:**
  - ➢ Large number of random numbers to be generated in each time slot
  - ➢ Number of random numbers to be generated increases with time
  - ➢ Huge increase in sorting complexity with large number of memory read and write
  - ➢ Huge memory requirement since precision of random numbers is critical for achieving high accuracy

S. V. Sai Santosh, and **S. J. Darak**, “Multi-armed Bandit Algorithms on Zynq System-on-Chip: Go Frequentist or Bayesian?,” in *IEEE Transactions on Neural Networks and Learning Systems*, accepted in June 2022.

Thompson Sampling Architecture

- **Proposed Idea:** Generate $T_i(t)$ number of random numbers for arm i. Sort them and select the $\left(\sum_{s=1}^{T_i(t)} X_{i,s}\right)^{th}$ random number.
- For generation for random numbers, we can use existing pseudo-random number generator (PRNG)
- **Efficient Architecture:**
  - ➢ Efficient sorting via grouping random numbers in predefined ranges
  - ➢ Grouping random numbers allows us to reduce the word length significantly
  - ➢ Significant reduction in number of comparators
  - ➢ Reduce the number of random numbers to be generated in each slot by reusing previously generated random numbers.
  - ➢ No need for separate random number generators for each arm.

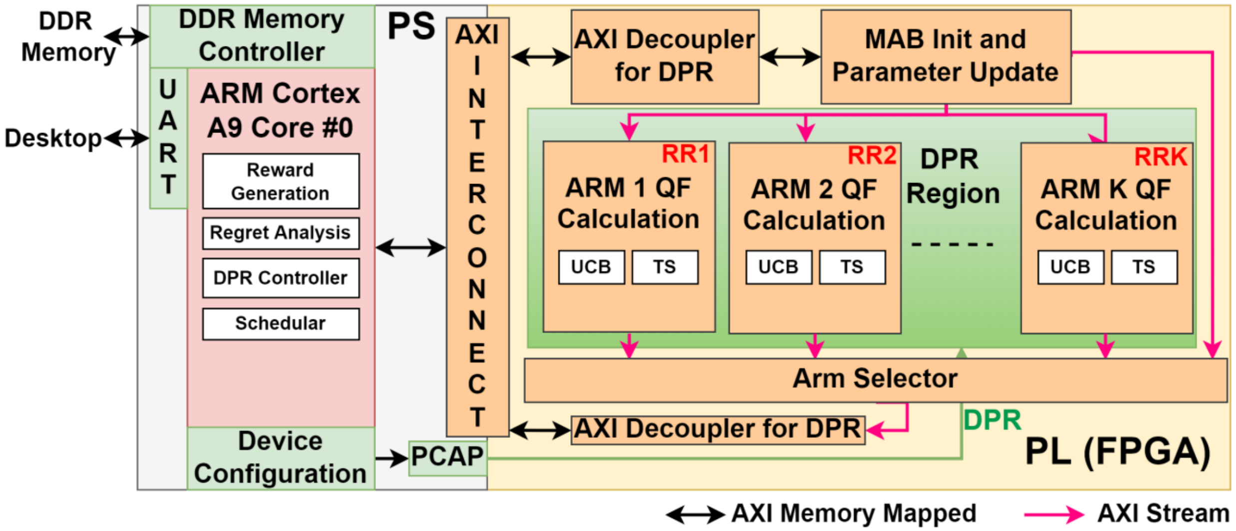
S. V. Sai Santosh, and **S. J. Darak**, "Multi-armed Bandit Algorithms on Zynq System-on-Chip: Go Frequentist or Bayesian?," in *IEEE Transactions on Neural Networks and Learning Systems*, accepted in June 2022.

Thompson Sampling Architecture

Floating Point

	LUTs	FFs	DSP	BRAM	Dynamic Power
UCB (K=5)	12068	11126	85	7.5	0.2
TS (K=5)	11414	6941	32	45	0.076
Velcro	21563	17201	115	52.5	0.358
Reconf. Architecture	12751	12162	85	45	0.2

	SoC	ARM Cortex A9	ARM Cortex A9 + NEON SIMD
UCB (K=5)	0.1	47	33
TS (K=5)	3.9	54	18



S. V. Sai Santosh, and **S. J. Darak**, "Multi-armed Bandit Algorithms on Zynq System-on-Chip: Go Frequentist or Bayesian?," in *IEEE Transactions on Neural Networks and Learning Systems*, accepted in June 2022.

Thompson Sampling Architecture

Floating Point

	LUTs	FFs	DSP	BRAM	Dynamic Power
UCB (K=5)	12068	11126	85	7.5	0.2
TS (K=5)	11414	6941	32	45	0.076
Velcro	21563	17201	115	52.5	0.358
Reconf. Architecture	12751	12162	85	45	0.2

	SoC	ARM Cortex A9	ARM Cortex A9 + NEON SIMD
UCB (K=5)	0.1	47	33
TS (K=5)	3.9	54	18

Fixed Point (27 bits)

	LUTs	FFs	DSP	BRAM	Dynamic Power
UCB (K=5)	8989	6758	0	0	0.027
TS (K=5)	2481	2428	0	5	0.014
Velcro	12381	9272	0	5	0.048
Reconf. Architecture	9753	8310	0	5	0.03

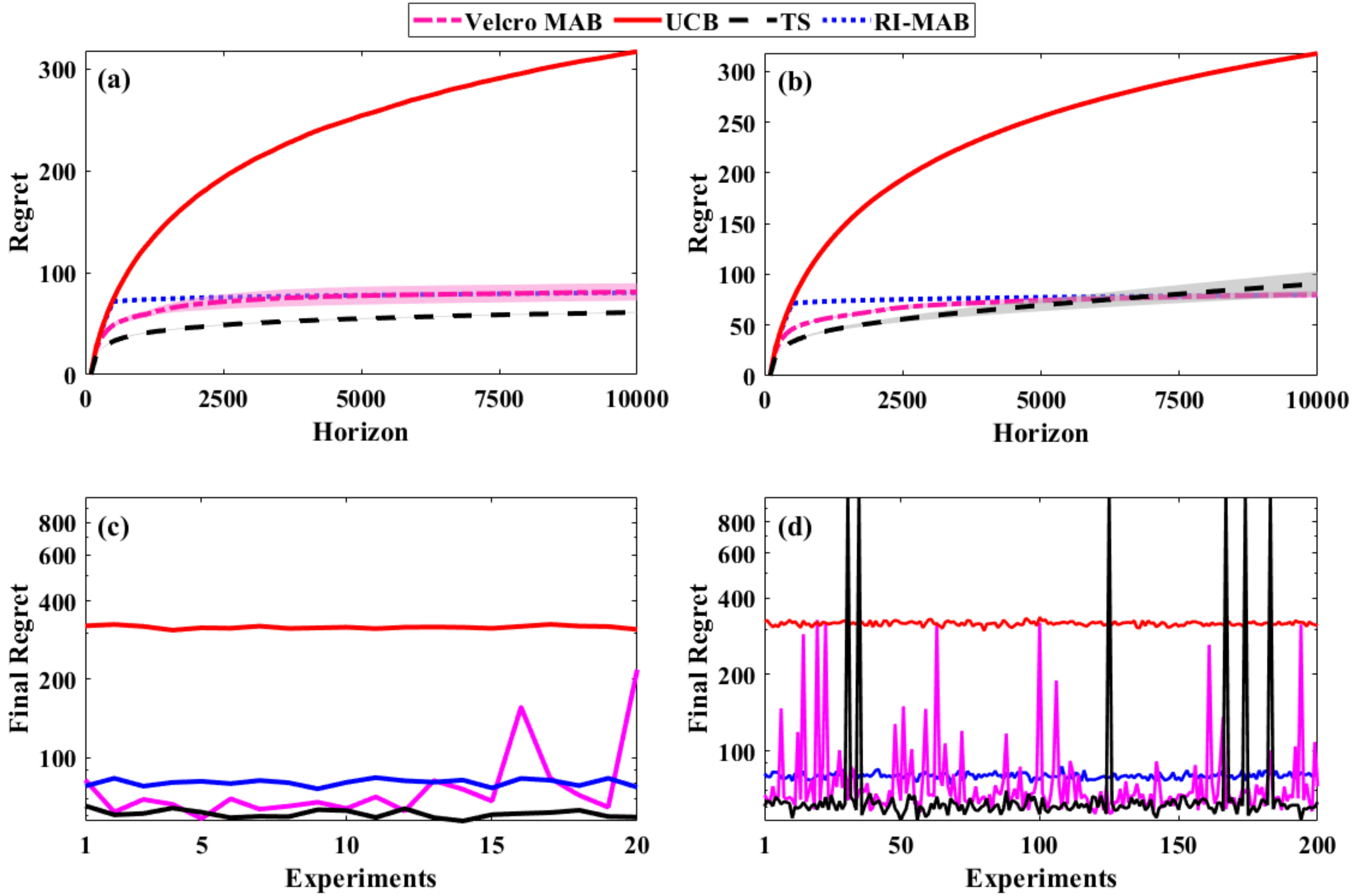
	SoC	ARM Cortex A9	ARM Cortex A9 + NEON SIMD
UCB (K=5)	0.1	47	33
TS (K=5)	1.5	54	18

Which MAB Algorithms to use?

- In real applications, arm statistics may not be fixed to single distribution
- UCB algorithm has been shown to work well in any distribution
- TS algorithm changes depending on the underlining distribution and as of now, architecture for Bernoulli distribution is available.
- However, right TS algorithm significantly outperforms UCB algorithm
- Can we design Intelligent architecture that can switch between UCB and TS architectures?
- **Proposed idea:** Exploration-exploitation trade-off among UCB and TS algorithm. We referred it as RI-MAB

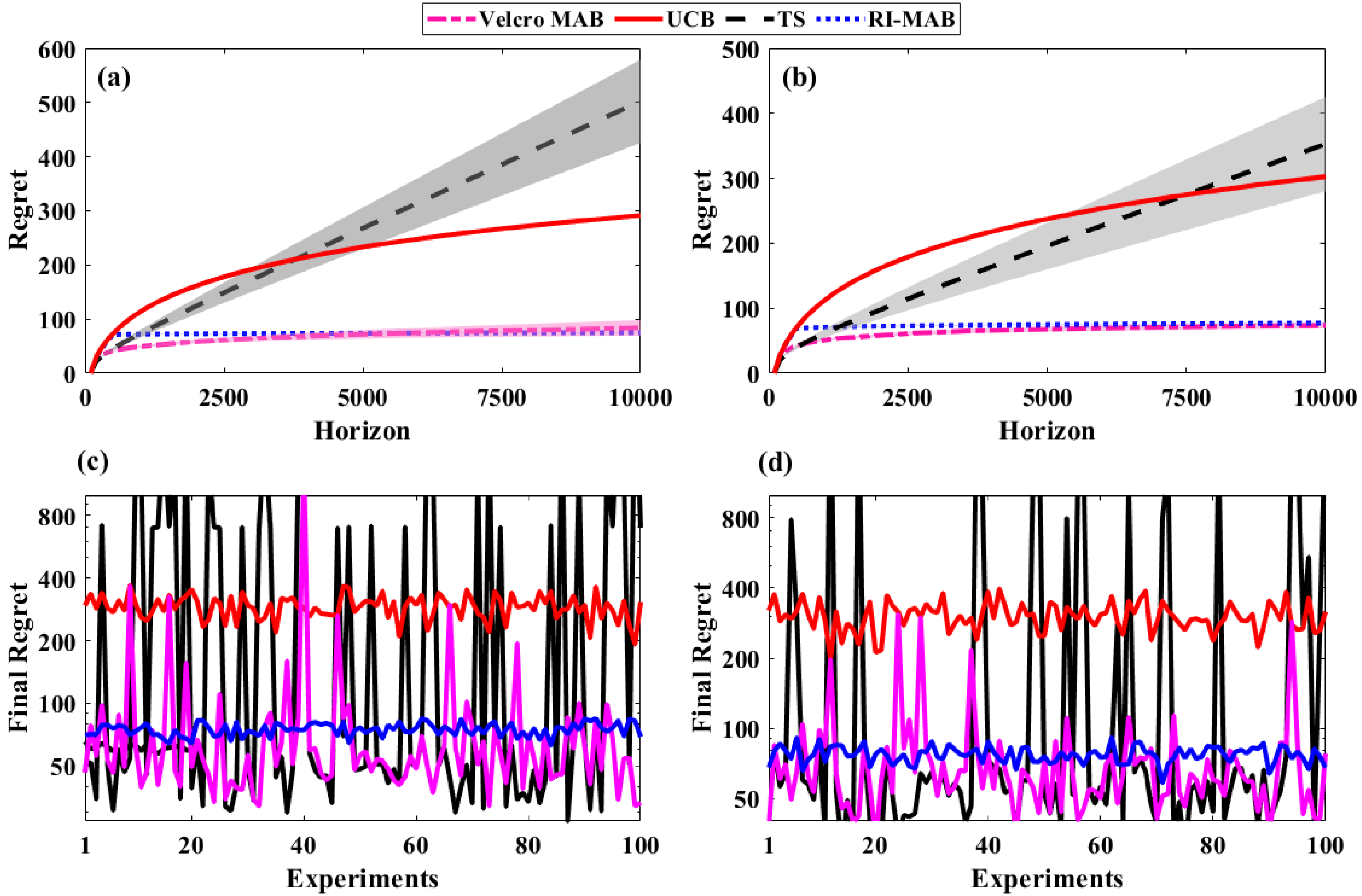
S. V. Sai Santosh, and **S. J. Darak**, "Multi-armed Bandit Algorithms on Zynq System-on-Chip: Go Frequentist or Bayesian?," in *IEEE Transactions on Neural Networks and Learning Systems*, accepted in June 2022.

RI-MAB



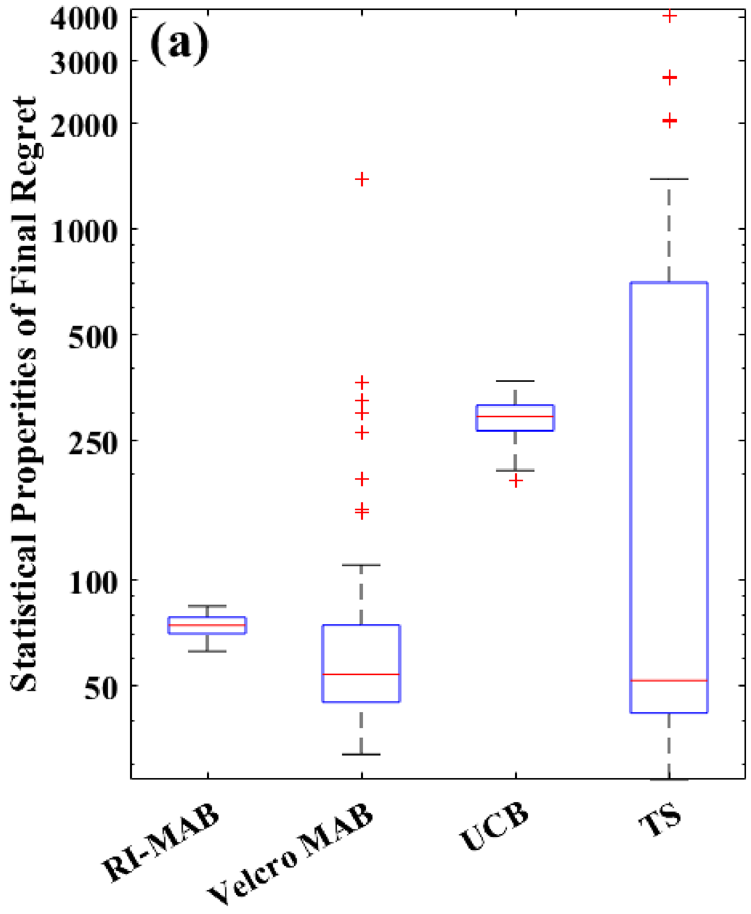
Velcro MAB
UCB
TS
RI-MAB
(a)
Regret
Horizon
0
100
200
300
0
2500
5000
7500
10000
(b)
Regret
Horizon
0
50
100
150
200
250
300
(c)
Final Regret
Experiments
100
200
400
600
800
1
5
10
15
20
(d)
Final Regret
Experiments
1
50
100
150
200

RI-MAB

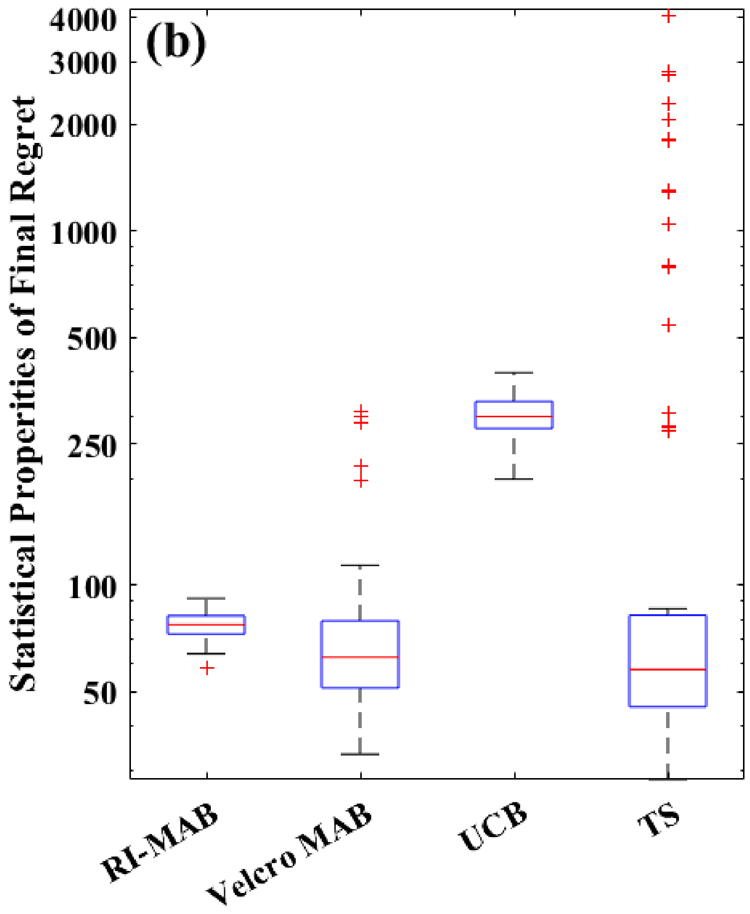


Velcro MAB
UCB
TS
RI-MAB
(a)
(b)
(c)
(d)
Regret
Horizon
Final Regret
Experiments
0
100
200
300
400
500
600
2500
5000
7500
10000
50
800
1
20
40
60
80

RI-MAB

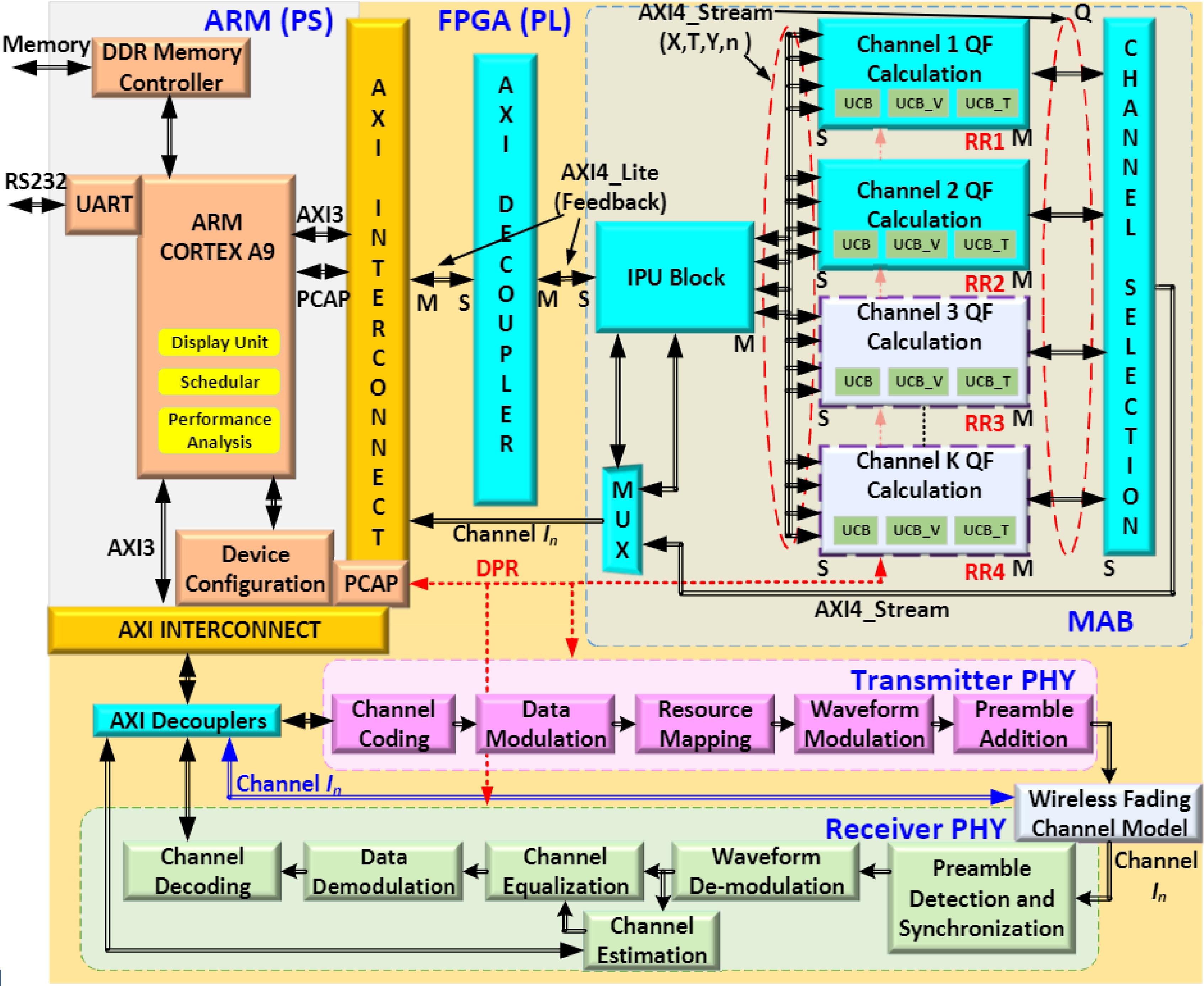


(a)
Statistical Properitics of Final Regret
4000
3000
2000
1000
500
250
100
50
RI-MAB
Velcro MAB
UCB
TS



(b)
Statistical Properitics of Final Regret
4000
3000
2000
1000
500
250
100
50
RI-MAB
Velcro MAB
UCB
TS

Reconfigurable and Intelligent PHY



Conclusions and Future Directions

- For algorithms to architecture mapping, efficient hardware software co-design and word length analysis is must.
- For next-generation applications, intelligent and reconfigurable architectures are desired.
- MAB algorithms are widely used in wireless, neural networks, e-commerce and health applications. Architectures for various MAB algorithms does not exist.
- Extension for quasi-stationary scenario
- What happens when number of arms are large in numbers?

tinyML® Talks

Enabling Ultra-low Power Machine Learning at the Edge

Thank You !

Email: sumit@iiitd.ac.in
YouTube: Algorithms to Architecture, IIIT Delhi

TINY ML

www.tinyML.org

Copyright Notice

www.tinyml.org

Copyright Notice