

PENGEMBANGAN AKSELERATOR PERANGKAT KERAS BERBASIS RISC-V UNTUK REINFORCEMENT LEARNING

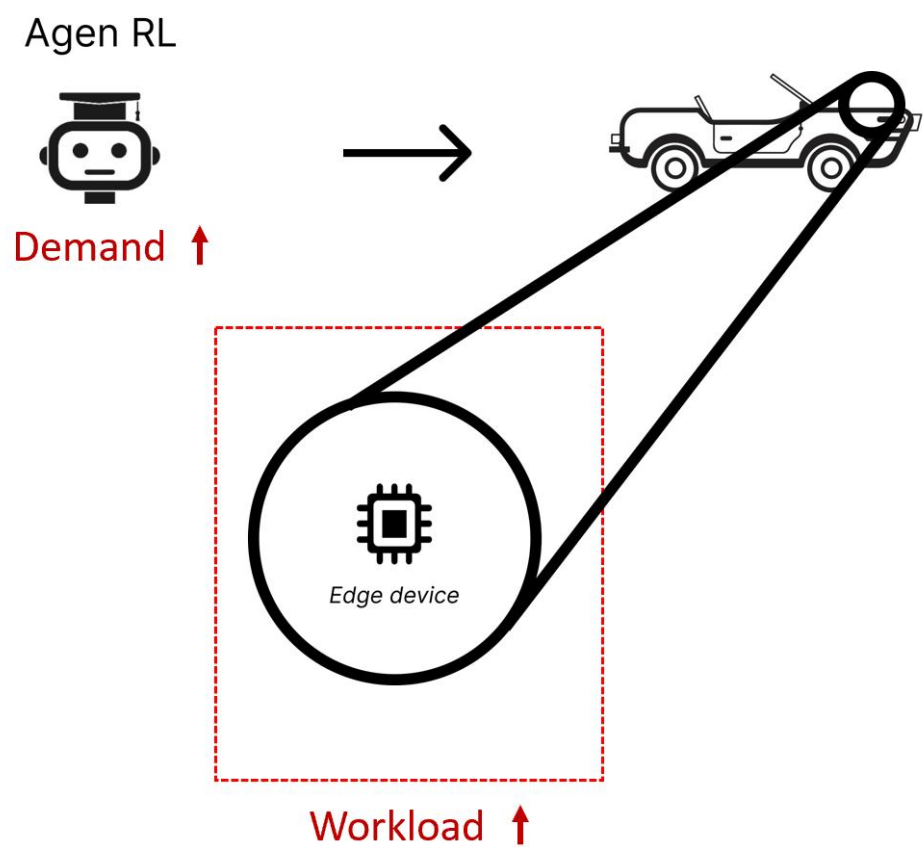
Mahasiswa: Muhammad Sulthan Mazaya (13320028)

Pembimbing 1: Dr. Ir. Eko Mursito Budi M.T.

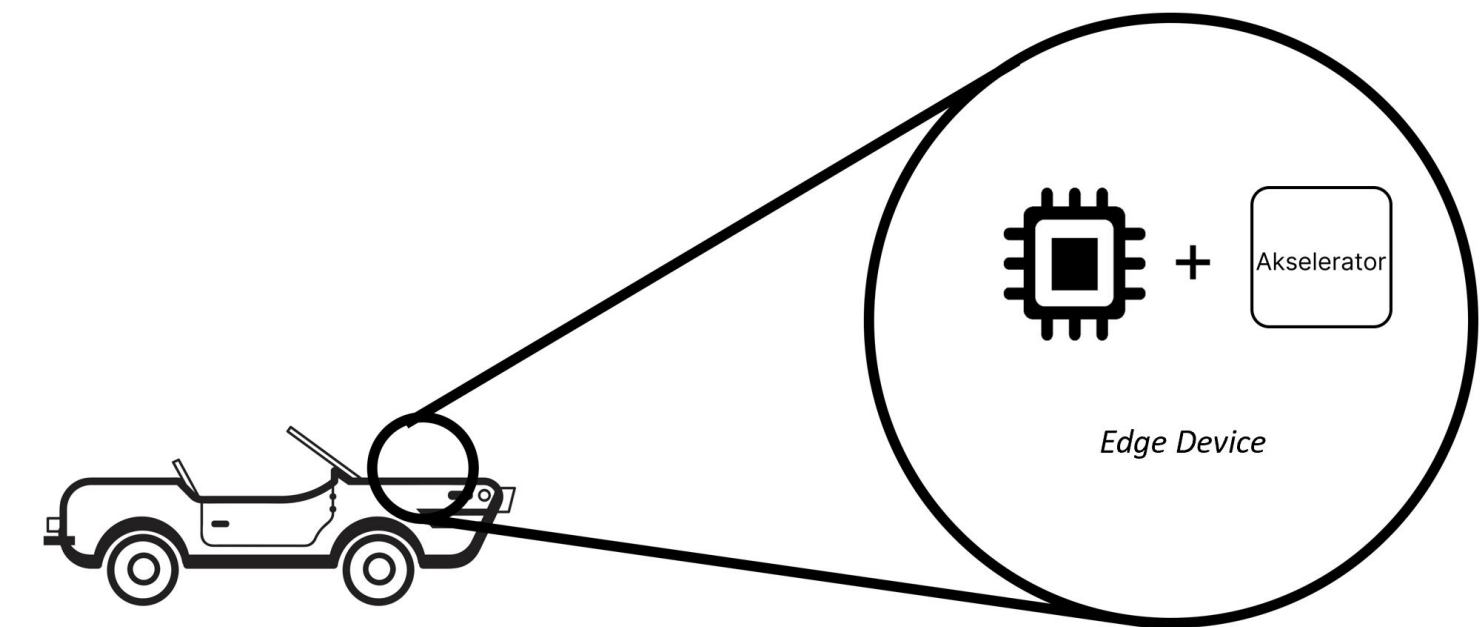
Pembimbing 2: Dr.Eng. Infall Syafalni, S.T., M.Sc.



Latar Belakang



Maraknya digunakan agen *Reinforcement Learning* (RL) [1] pada sistem otonom berbasis *Edge Computing* [2] menyebabkan diperlukannya sistem komputasi yang dapat melakukan komputasi algoritma RL secara efektif dan efisien.



Sehingga, pada penelitian tugas akhir ini dilakukan sebuah pengembangan akselerator perangkat keras pada prosesor RISC-V untuk menghasilkan sebuah perangkat komputasi yang rendah daya dan tetap mampu melakukan perhitungan algoritma RL secara efektif dan efisien.

Desain Sistem

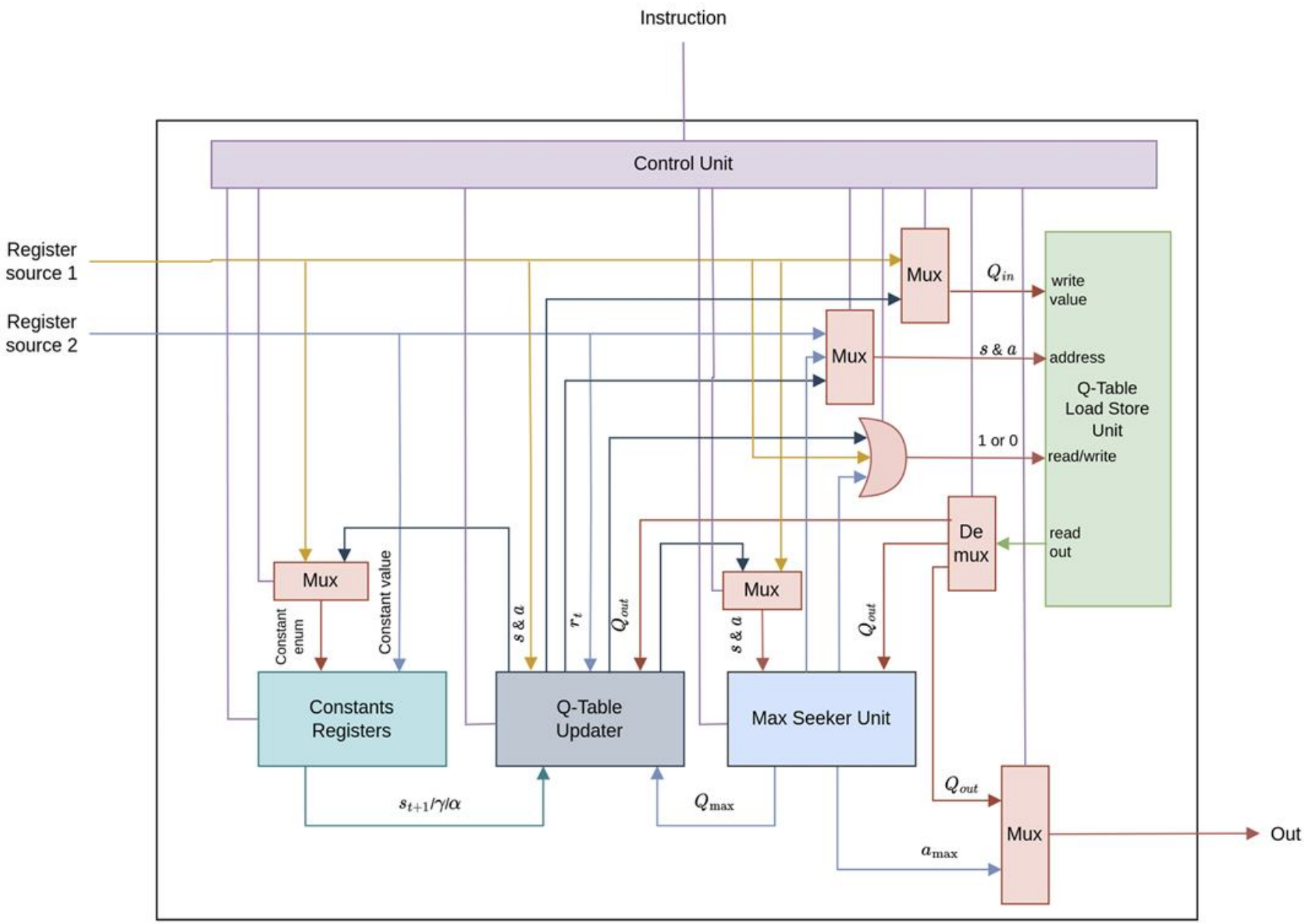
```
Algoritma 6 Pembagian HW/SW co-design
Masukan: jumlah episode, learning rate ( $\alpha$ ), dan discount factor ( $\gamma$ )
Keluaran: Q-Table
1: Inisialisasi Q-Table untuk setiap state dan aksi ▷ Perangkat keras
2: Simpan  $\gamma$  dan  $\alpha$  pada akselerator ▷ Perangkat keras
3: while banyak_episode < jumlah_episode do
4:    $s_t \leftarrow s_0$ 
5:   while  $s_t \neq s_{terminal}$  do
6:     Pilih aksi (A) untuk state kini ( $s_t$ ) ▷ Perangkat keras
7:     Lakukan A, dapatkan reward dan  $s_{t+1}$ 
8:     Simpan  $s_{t+1}$  pada akselerator ▷ Perangkat keras
9:     Hitung nilai Q-Table indeks terkini ▷ Perangkat keras
10:     $s_t \leftarrow s_{t+1}$ 
11:  end while
12:  banyak_episode  $\leftarrow$  banyak_episode + 1
13:   $cr_t \leftarrow$  cumulative reward dari Q-Table
14:  if  $cr_t > cr_{t-1}$  then
15:    Ubah nilai  $maxQ(s_{t+1}, a)$  dari Q-Table
16:  end if
17:   $cr_{t-1} \leftarrow cr_t$ 
18:  memori Q-Table  $\leftarrow$  Q-Table
19:  episode  $\leftarrow$  episode + 1
20: end while
```

Dilakukan sebuah teknik perancangan *HW/SW Co-design* untuk melakukan partisi algoritma RL pada akselerator perangkat keras.

Teknik ini dipilih karena menghasilkan desain akselerator yang mampu digunakan secara fleksibel oleh pengguna akselerator sehingga dapat dirancang *policy* dan *reward function* tanpa perlu mengubah konfigurasi perangkat keras.

Desain *HW/SW Co-design* tersebut kemudian diimplementasikan sebagai akselerator perangkat keras yang memiliki 4 modul: *Q-Table Load Store Unit*, *Constant Registers*, *Q-Table Updater*, dan *Max Seeker Unit*.

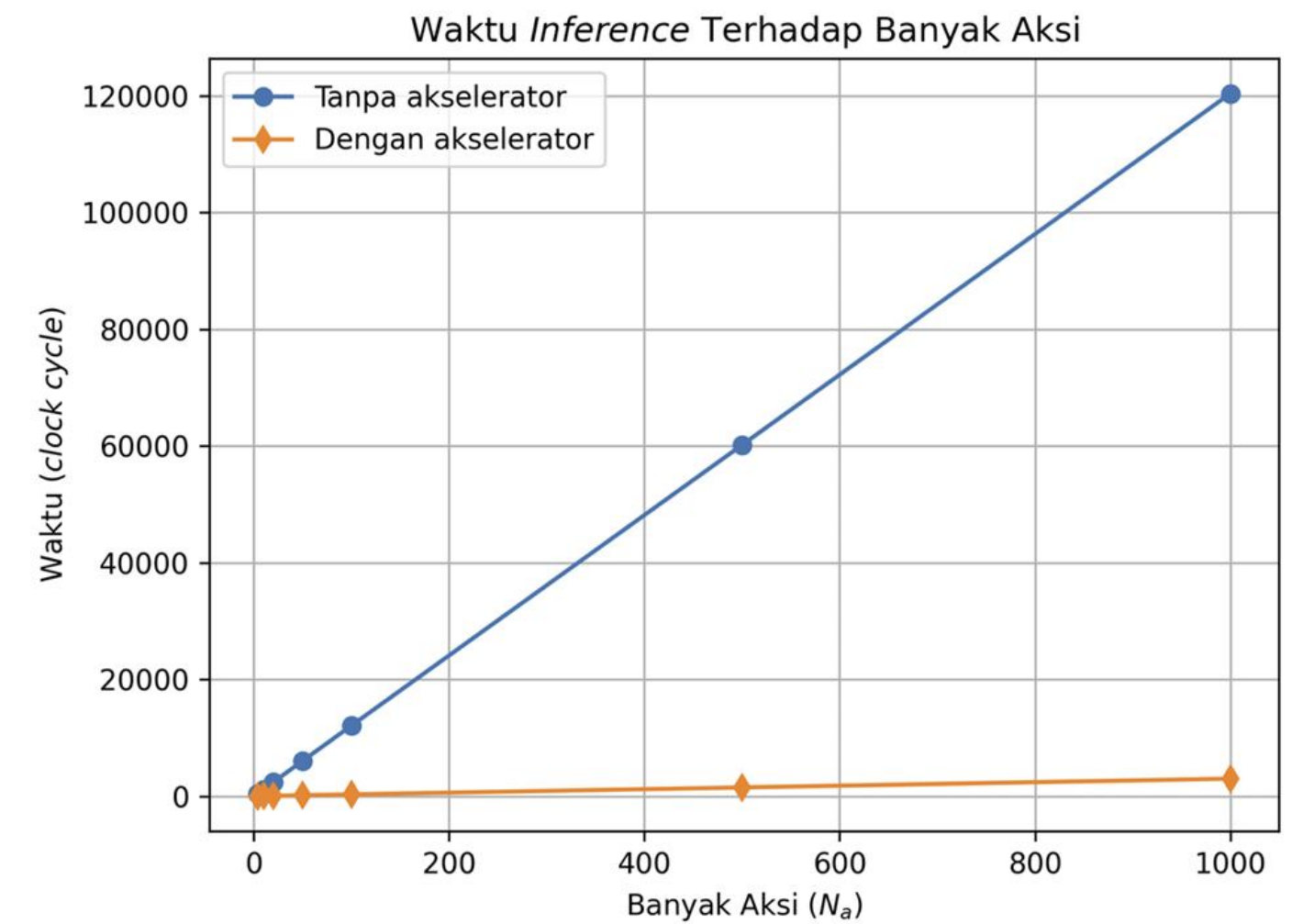
Masing-masing modul tersebut memiliki instruksi tambahan *assembly* RISC-V untuk mengontrol penggunaan modul.



Hasil

Reference	Spanò et al. [17]	Da Silva et al. [16]	Y. Meng et al. [32]	Sutisna et al. [15]	Proposed
Design Level	Standalone Core			System on Chip	
Action Policy	NA	random	ϵ -greedy	decreasing- ϵ	ϵ -greedy
Number of Agents (G)	single	single	single	double	single
Bit Width	16	32	30	16	32
LUT	333	682	77574	172	2490
Registers	258	606	13175	NA	2348
BRAM	NA	NA	266	NA	22

Didapatkan hasil implementasi akselerator yang memiliki efisiensi menyaingi *state of the art* [3] [4] dengan resolusi 32-bit. Selain itu, akselerator 40 kali lebih cepat dibandingkan impementasi pada perangkat lunak pada skala asimtotik.



Kesimpulan

Implementasi desain akselerator perangkat keras RISC-V untuk RL menghasilkan prosesor dengan performa yang lebih efektif dan efisien untuk digunakan pada *edge computing*.

[1] M. Waltz and K. Fu, "A heuristic approach to reinforcement learning control systems," in IEEE Transactions on Automatic Control, vol. 10, no. 4, pp. 390–398, October 1965, doi: 10.1109/TAC.1965.1098193.
[2] J. Sakhdari, B. Zolfaghari, S. Izadpanah, et al., Edge computing: A systematic mapping study, 2022. arXiv: 2102.02720 [cs.NI]
[3] S. Spano, G. C. Cardarilli, L. Di Nunzio, et al., "An efficient hardware implementation of reinforcement learning: The q-learning algorithm," IEEE Access, vol. 7, pp. 186 340–186 351, 2019. DOI: 10.1109/ACCESS.2019. 2961174
[4] N. Sutisna, A. M. R. Ilmy, I. Syafalni, R. Mulyawan, and T. Adiono, "Farane-q: Fast parallel and pipeline q-learning accelerator for configurable reinforcement learning soc," IEEE Access, vol. 11, pp. 144–161, 2023.