PU-NTM (T-DNC/NTM-PU)

QueenField

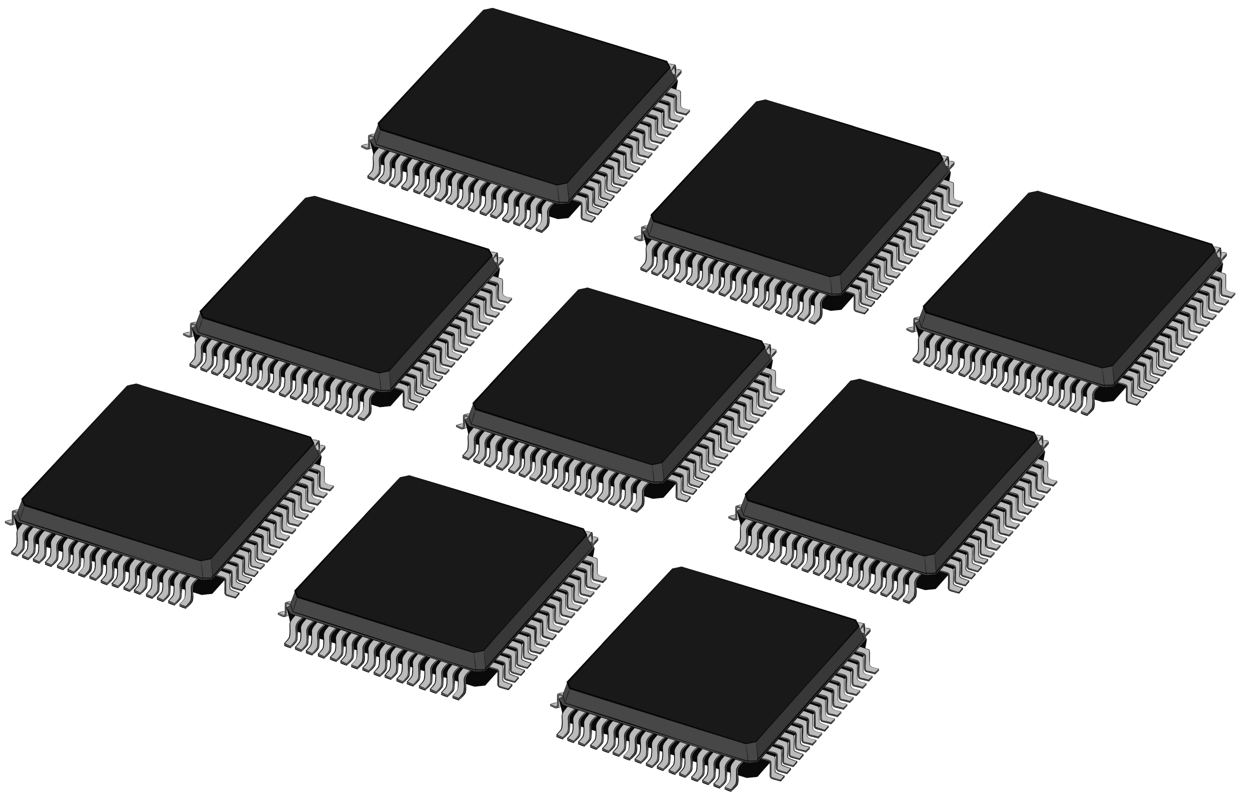

Figure 1: QueenField

1. Mechanics

2. Information

2.1. Bit

2.2.1. YES/NOT Gate

2.2.2. AND/NAND Gate

2.2.3. OR/NOR Gate

2.2.4. XOR/XNOR Gate

2.3. Combinational Logic

2.3.1. Arithmetic Circuits

2.3.2. Logic Circuits

2.3. Combinational Logic

2.4. Finite State Machine

$$T = (Q, \Sigma, \delta, q_0, F)$$
$$Q \subseteq H$$
$$\delta : Q \times \Sigma \otimes Q \to Q$$

2.5. Pushdown Automaton

$$T = (Q, \Sigma, b, \Gamma, \delta, q_0, F)$$
$$Q \subseteq H$$
$$\delta : \Sigma \times Q \otimes \Gamma \to \Sigma \times Q \otimes \Gamma \times \{L, R\}$$

3. Neural Network

3.1. Feedforward Neural Network

$$\begin{aligned} h_t &= \sigma_g(W_h \cdot x_t + U_h \cdot h_{t-1} + b_h) \\ y_t &= \sigma_g(W_y \cdot h_t + b_y) \end{aligned}$$

$$\begin{aligned} h_t &= \sigma_g(W_h \star x_t + U_h \star h_{t-1} + b_h) \\ y_t &= \sigma_g(W_y \star h_t + b_y) \end{aligned}$$

3.2. Long Short Term Memory Neural Network

$$\begin{aligned} a_t &= \sigma_g(W_a \cdot x_t + U_a \cdot h_{t-1} + b_a) \\ f_t &= \sigma_g(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \\ i_t &= \sigma_g(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i) \\ o_t &= \sigma_g(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \\ c_t &= f_t \circ c_{t-1} + i_t \circ a_t \\ h_t &= o_t \circ \sigma_g(c_t)) \end{aligned}$$

$$
\begin{aligned}
a_t &= \sigma_g(W_a \star x_t + U_a \star h_{t-1} + b_a) \\
f_t &= \sigma_g(W_f \star x_t + U_f \star h_{t-1} + b_f) \\
i_t &= \sigma_g(W_i \star x_t + U_i \star h_{t-1} + b_i) \\
o_t &= \sigma_g(W_o \star x_t + U_o \star h_{t-1} + b_o) \\
c_t &= f_t \circ c_{t-1} + i_t \circ a_t \\
h_t &= o_t \circ \sigma_g(c_t))
\end{aligned}
$$

3.3. Transformer Neural Network

$$\text{attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right) V$$

4. Turing Machine

$$T = (Q, \Sigma, b, \Gamma, \delta, q_0, F)$$

$$Q \subseteq H$$

$$\delta : \Sigma \times Q \otimes \Gamma \to \Sigma \times Q \otimes \Gamma \times \{L, R\}$$

4.1. Neural Turing Machine

- Definitions

$$\mathcal{D}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$$

- Reading

$$\sum_{i=0}^{M-1} w_t(i) = 1; \quad 0 \leq w_t(i) \leq 1$$

$$\mathbf{r}_t \longleftarrow \sum_{i=0}^{M-1} w_t(i) \mathbf{M}_t(i)$$

- Writing

$$
\begin{aligned}
\tilde{\mathbf{M}}_t(i) &\longleftarrow \mathbf{M}_{t-1}(i) \left[\mathbf{1} - w_t(i) \mathbf{e}_t\right] \\
\mathbf{M}_t(i) &\longleftarrow \tilde{\mathbf{M}}_t(i) + w_t(i)\, \mathbf{a}_t
\end{aligned}
$$

- Addressing

$$w_t^c(i) \longleftarrow \frac{\exp\Big(\beta_t \mathcal{D}\big[\mathbf{k}_t, \mathbf{M}_t(i)\big]\Big)}{\sum_{j=0}^{N-1} \exp\Big(\beta_t \mathcal{D}\big[\mathbf{k}_t, \mathbf{M}_t(j)\big]\Big)}$$

$$\mathbf{w}_t^g \longleftarrow g_t \mathbf{w}_t^c + (1 - g_t) \mathbf{w}_{t-1}$$

$$\tilde{w}_t(i) \longleftarrow \sum_{j=0}^{N-1} w_t^g(j)\, s_t(i - j)$$

$$w_t(i) \longleftarrow \frac{\tilde{w}_t(i)^{\gamma_t}}{\sum_{j=0}^{N-1} \tilde{w}_t(j)^{\gamma_t}}$$

4.1.1. Feedforward Neural Turing Machine

4.1.2. LSTM Neural Turing Machine

4.1.3. Transformer Neural Turing Machine

4.2. Differentiable Neural Computer

- Definitions

$$\mathcal{D}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$\text{oneplus}(x) = 1 + \log(1 + e^x)$$

$$\text{softmax}(\mathbf{x})_j = \frac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}}$$

- Addressing

$$M_t = M_{t-1} \circ (E - \mathbf{w}_t^w \mathbf{e}_t^\intercal) + \mathbf{w}_t^w \mathbf{v}_t^\intercal$$

$$\mathbf{u}_t = (\mathbf{u}_{t-1} + \mathbf{w}_{t-1}^w - \mathbf{u}_{t-1} \circ \mathbf{w}_{t-1}^w) \circ \boldsymbol{\psi}_t$$

$$\mathbf{p}_t = \left(1 - \sum_i \mathbf{w}_t^w[i]\right) \mathbf{p}_{t-1} + \mathbf{w}_t^w$$

$$L_t = (\mathbf{1} - \mathbf{I}) \left[(1 - \mathbf{w}_t^w[i] - \mathbf{w}_t^j) L_{t-1}[i,j] + \mathbf{w}_t^w[i] \mathbf{p}_{t-1}^j\right]$$

$$\mathbf{w}_t^w = g_t^w [g_t^a \mathbf{a}_t + (1 - g_t^a) \mathbf{c}_t^w]$$

$$\mathbf{w}_t^{r,i} = \boldsymbol{\pi}_t^i[1] \mathbf{b}_t^i + \boldsymbol{\pi}_t^i[2] \mathbf{c}_t^{r,i} + \boldsymbol{\pi}_t^i[3] \mathbf{f}_t^i$$

$$\mathbf{r}_t^i = M_t^\intercal \mathbf{w}_t^{r,i}$$

$$\mathcal{C}(M, \mathbf{k}, \beta)[i] = \frac{\exp\{\mathcal{D}(\mathbf{k}, M[i, \cdot])\beta\}}{\sum_j \exp\{\mathcal{D}(\mathbf{k}, M[j, \cdot])\beta\}}$$

$$\mathbf{a}_t[\phi_t[j]] = (1 - \mathbf{u}_t[\phi_t[j]]) \prod_{i=1}^{j-1} \mathbf{u}_t[\phi_t[i]]$$

$$\mathbf{c}_t^w = \mathcal{C}(M_{t-1}, \mathbf{k}_t^w, \beta_t^w)$$

$$\mathbf{c}_t^{r,i} = \mathcal{C}(M_{t-1}, \mathbf{k}_t^{r,i}, \beta_t^{r,i})$$

$$\mathbf{f}_t^i = L_t \mathbf{w}_{t-1}^{r,i}$$

$$\mathbf{b}_t^i = L_t^\intercal \mathbf{w}_{t-1}^{r,i}$$

$$\boldsymbol{\psi}_t = \prod_{i=1}^{R} \left(\mathbf{1} - f_t^i \mathbf{w}_{t-1}^{r,i}\right)$$

4.2.1. Feedforward Differentiable Neural Computer

4.2.2. LSTM Differentiable Neural Computer

4.2.3. Transformer Differentiable Neural Computer

5. Computer Architecture

5.1. von Neumann Architecture

5.1.1. Control Unit

5.1.2. ALU

5.1.3. Memory Unit

5.1.4. I/O Unit

5.2. Harvard Architecture

5.2.1. Control Unit

5.2.2. ALU

5.2.3.Memory Unit

5.2.4.I/O Unit