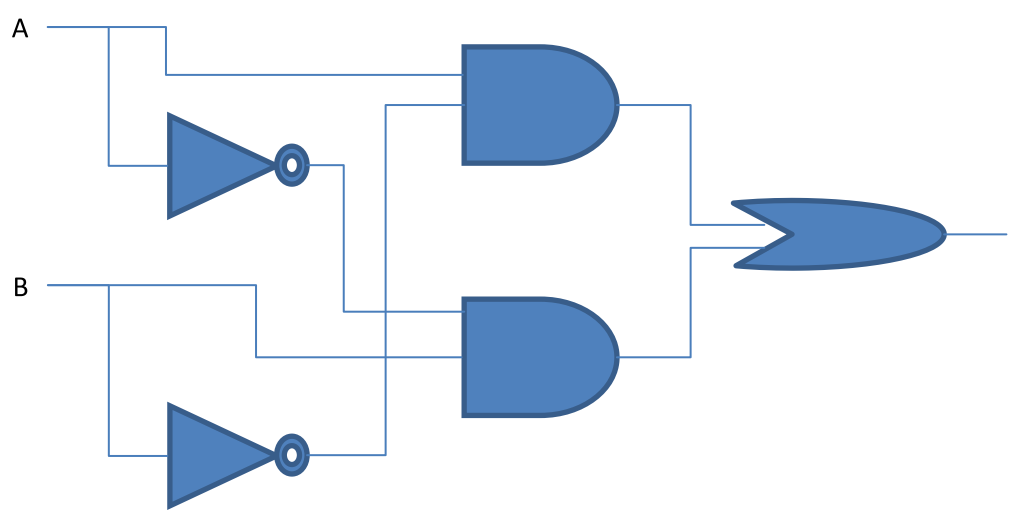
Eric Christopher / 15M53531

Xor Gate

There are 2 possibilities (using gate logic):

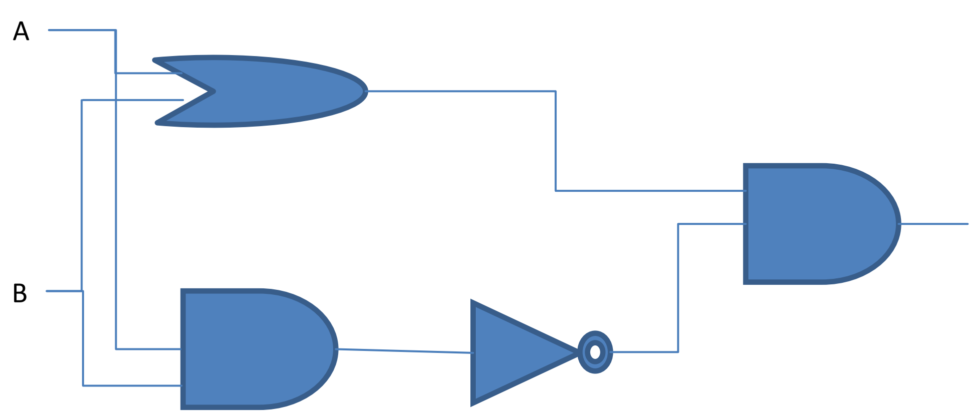
1. $A \oplus B = A\bar{B} + B\bar{A}$

Input Layer Hidden Layer Output Layer



2. $A \oplus B = (A + B)(\overline{AB})$

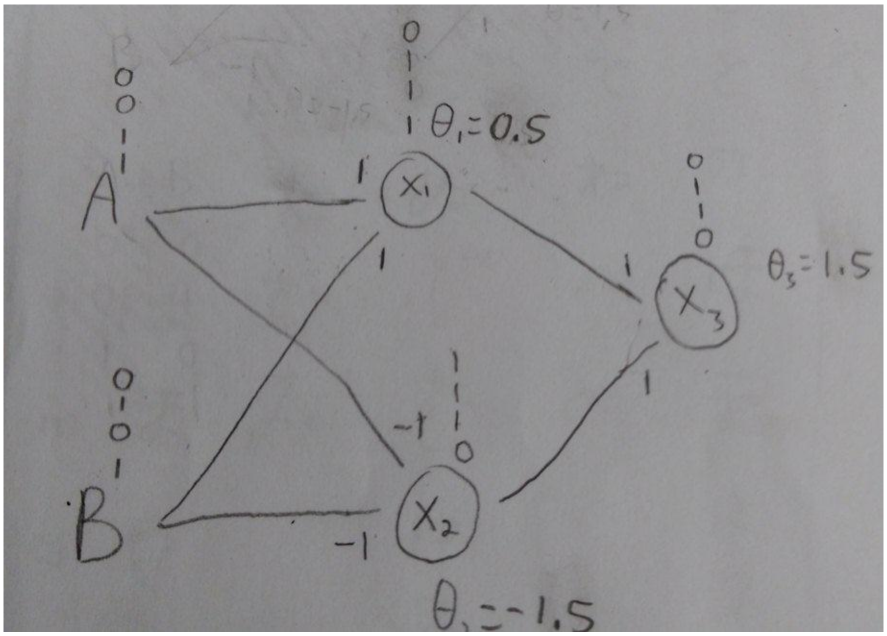
Input Layer Hidden Layer Output Layer



But I tried another possibility using neural network (not depend on gate logic)

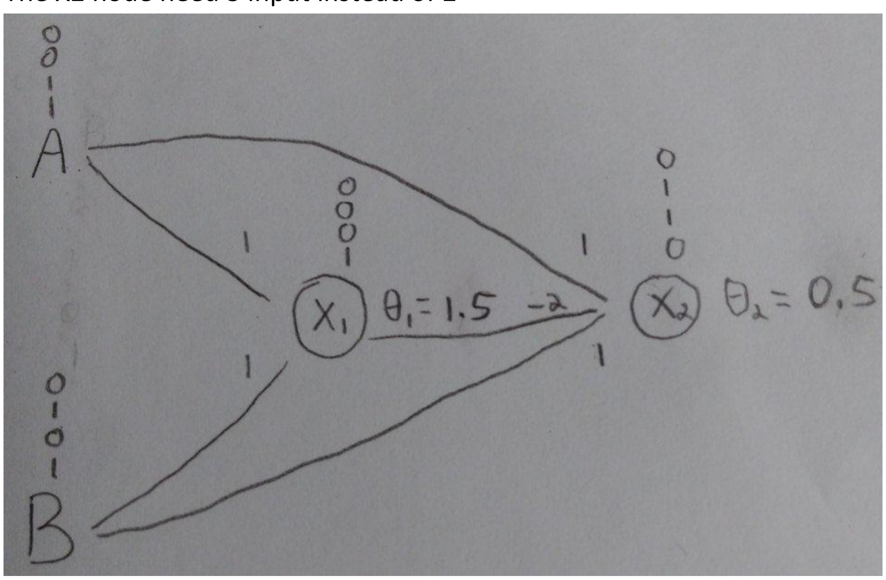
3. Using 3 nodes

   same with the 2, but using NAND (X2 is NAND gate) instead of AND and NOT gate



4. Using 2 nodes

   The X2 node need 3 input instead of 2



Many ways to create XOR gate, after a few trying, I can’t make it 1 node, so my best is using 2 nodes to create XOR gate.

For winner takes all, in my experience, we need to check each node at least once

Because I found that our program can be stopped before we get all 0 except 1 node 1

Here is the case

Input: 1,0,1,1,1 , all weight=-1, all threshold=-1,5

```
Result:
--(1,0,1,1,1) - E=4.0
================================================
-change node 1
--S=-3.0
--(0,0,1,1,1) - E=1.5
================================================
-change node 2
--S=-3.0
--(0,0,1,1,1) - E=1.5 ← ======================= we will stop in this step because this node same with the previous one
================================================
-change node 3
--S=-2.0
--(0,0,0,1,1) - E=0.0
================================================
-change node 4
--S=-1.0
--(0,0,0,0,1) - E=-0.5
================================================
-change node 5
--S=0.0
--(0,0,0,0,1) - E=-0.5
================================================
-change node 1
--S=0.0
--(0,0,0,0,1) - E=-0.5
================================================
======================END======================
================================================
```

Prove for E always decrease if we try to change input in deterministic and binary model

$$E = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_i x_j + \sum_{i=1}^{n}\theta_i x_i + C \qquad -\frac{1}{2}(w_{ij}x_i x_j + w_{ji}x_j x_i) = -\underline{\underline{w_{ij}x_i x_j}} \qquad w_{ij} = w_{ji} \quad \& \ \underline{\underline{w_{nn} = 0}}$$

$$\Delta E = E|_{x_i=1} - E|_{x_i=0}$$

$$= -w_{12}x_1x_2 - w_{13}x_1x_3 \ldots - w_{i1}x_ix_1 - \ldots - w_{in}x_ix_n - \ldots - w_{n-1\,n}x_{n-1}x_n + \theta_1x_1 + \theta_2x_2 + \ldots + \theta_ix_i + \ldots + \theta_nx_n + C$$

$$-(-w_{12}x_1x_2 - w_{13}x_1x_3 \ldots \quad \xrightarrow{\text{remaining}} \quad -w_{n-1\,n}x_{n-1}x_n + \theta_1x_1 + \theta_2x_2 + \ldots + \theta_nx_n + C)$$

$$= -w_{i1}x_ix_1 - \ldots - w_{in}x_ix_n + \theta_ix_i$$

$$= -\sum_{j=1, j\neq i}^{n} w_{ij}x_ix_j + \theta_ix_i \quad , x_i = 1 \text{ (it is from } E|_{x_i=1} \text{ after all)}$$

$$= -\sum_{j=1, j\neq i}^{n} w_{ij}x_j + \theta_i$$

According to the deterministic & binary model

$$S_i = +\sum_{j=1, j\neq i}^{n} w_{ij}x_j \implies \Delta E = -S_i + \theta_i$$

$$x_i = \begin{cases} 1 & \text{when } S_i \geq \theta_i \\ & \Delta E|_{x_i=0\to1} \leq 0 \to \text{decrease} \\ 0 & \text{when } S_i < \theta_i \\ & \Delta E|_{x_i=0\to1} > 0 \\ & \Delta E|_{x_i=1\to0} < 0 \to \text{decrease} \end{cases}$$