We start with introducing three complexity classes of languages which are of particular importance in circuit complexity:

- **TC\(^k\)**
  The class of languages recognizable by a Dlogtime-uniform circuit family of Size\((n^{O(1)})\), with Depth\((O(\log^k n))\), using Majority and \(\neg\) gates\(^1\).

- **AC\(^k\)**
  The class of languages recognizable by a Dlogtime-uniform circuit family of Size\((n^{O(1)})\), with Depth\((O(\log^k n))\), using unbounded fan-in \(\land, \lor, \text{ and } \neg\) gates.

- **NC\(^k\)**
  The class of languages recognizable by a Dlogtime-uniform circuit family of Size\((n^{O(1)})\), with Depth\((O(\log^k n))\), using bounded fan-in \(\land, \lor, \text{ and } \neg\) gates.

Among these, the class NC\(^0\) (polynomial size, constant depth, with bounded fan-in gates) have the least importance, since the output gate can only depend on a constant number of inputs, and having constant depth, the output gate is not able to check all the inputs (even a simple function like AND\((x_1, \ldots, x_n)\) can not be implemented in this model). The general hierarchy among the rest of the complexity classes is shown in figure 1 (The class NC is defined as \(\cup_k NC^k\)). In what follows, inclusions of the classes TC\(^k\) in NC\(^{k+1}\)

![Figure 1: The hierarchy of complexity classes involving circuits.](image-url)

will be justified, the rest of the inclusions in this hierarchy are left to the reader:

**Theorem 1** For all \(k\), TC\(^k\) \(\subseteq\) NC\(^{k+1}\).

\(^1\) *Majority* is an unbounded fan-in gate, and for the input \(x_1, \ldots, x_n\)

\[
\text{Majority}(x_1, \ldots, x_n) = \begin{cases} 
1 & \text{if } \sum_{i} x_i \geq n/2, \\
0 & \text{Otherwise}.
\end{cases}
\]
Proof. It is sufficient to show that the Majority gate is in NC$^1$, since if in a TC$^k$ circuit, each Majority gate is replaced by an NC$^1$ circuit, the resulting circuit will be a Dlogtime-uniform circuit of Size($n^{O(1)}$), which is built using bounded fan-in $\land$, $\lor$, and $\neg$ gates, and has depth $O(\log^{k+1} n)$.

In order to show Majority$(x_1, \ldots, x_n) \in NC^1$, we need to compute the sum of the bits $x_1, \ldots, x_n$. (Using the usual divide and conquer method to carry out this addition will not result in a Depth$(O(\log n))$ circuit.)

Observe that adding three $b$-bit numbers $<x_b, \ldots, x_1>, <y_b, \ldots, y_1>, <z_b, \ldots, z_1>$ can be reduced to adding two $b+1$-bit numbers, $<u_{b+1}, \ldots, u_1>$ and $<v_{b+1}, \ldots, v_1>$, defined as follows:

\[
\begin{align*}
u_i &= \text{Parity}(x_i, y_i, z_i), & 1 \leq i \leq b, \\
u_{b+1} &= 0, \\
v_{i+1} &= \text{Majority}(x_i, y_i, z_i), & 1 \leq i \leq b.
\end{align*}
\]

It is easy to see that Parity$(x_i, y_i, z_i)$ and Majority$(x_i, y_i, z_i)$ can be implemented with constant depth circuits.

Getting back to the original problem of computing the summation of the bits $x_1, x_2, \ldots, x_n$, we can start grouping the bits into sets of size $3$, \{ $x_1, x_2, x_3$, $x_4, x_5, x_6$, $\ldots$, $x_{n-2}, x_{n-1}, x_n$ \} (W.L.O.G assume $n = 3k$). By the previous observation the summation of the bits in each group $\{x, y, z\}$ can be replaced by the sum of two, two bit numbers $\{<v_2v_1, u_2u_1>, <z_2z_1>\}$. As a result of this reduction, the total number of additions required to compute the original sum will be reduced by a factor of $\frac{1}{2}$. Repeating the same procedure for $O(\log n)$ steps will reduce the original problem to the problem of adding two $O(\log n)$-bits numbers (see Figure 2). Up to this point we have only used $O(\log n)$ gates to compute the Majority. It remains to show that the final two numbers $<y_m, \ldots, y_1>$ and $<z_m, \ldots, z_1>$ can be added using a circuit in NC$^1$. To this ends, we will show that using an AC$^0$ circuit, this last sum can be easily computed. Let $<\sigma_{m+1}, \ldots, \sigma_1>$ denote the summand of $<y_m, \ldots, y_1>$ and $<z_m, \ldots, z_1>$ and $c_i$ denote the $i^{th}$ carry bit, then the following relations hold

\[
\begin{align*}
c_0 &= 0, \\
c_i &= \bigvee_{j \leq i} \left[ (y_j \land z_j) \land \bigwedge_{j \leq k < i} (y_k \lor z_k) \right], & 1 \leq i \leq m, \\
\sigma_i &= c_{i-1} \oplus y_i \oplus z_i, & 1 \leq i \leq m, \\
\sigma_{m+1} &= c_m.
\end{align*}
\]

Observe that, using bounded fan-in gates to compute $c_i$’s and $\sigma_i$’s will increase the depth of the circuit by

![Figure 2: Reduction of Majority to an NC$^1$ circuit.](image)
a multiplicative factor of $O(\log n)$. Define $\sigma = \sigma_{m+1} \ldots \sigma_2 \sigma_1$, $p = 2^{[\log n]-1} - \frac{p}{2}$ and form $s = 11 \ldots 1$ (the string of $p$ padded 1's). It can be shown that; \textbf{Majority}$(x_1, x_2, \ldots, x_n) = 1$ if and only if $\sigma + s \geq 2^{[\log n]-1}$. Finally, by using an $AC^0$ circuit which verifies the last bit of the summand $\sigma + s$, the value of \textbf{Majority} can be determined. $\square$