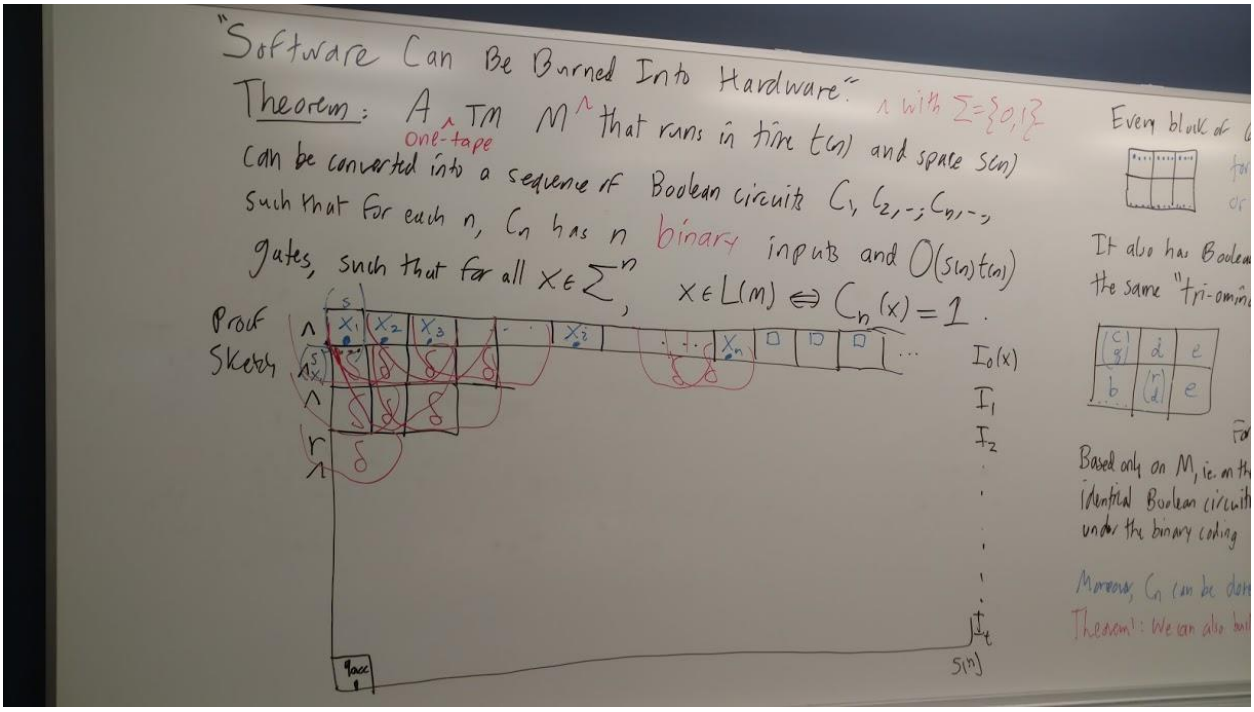
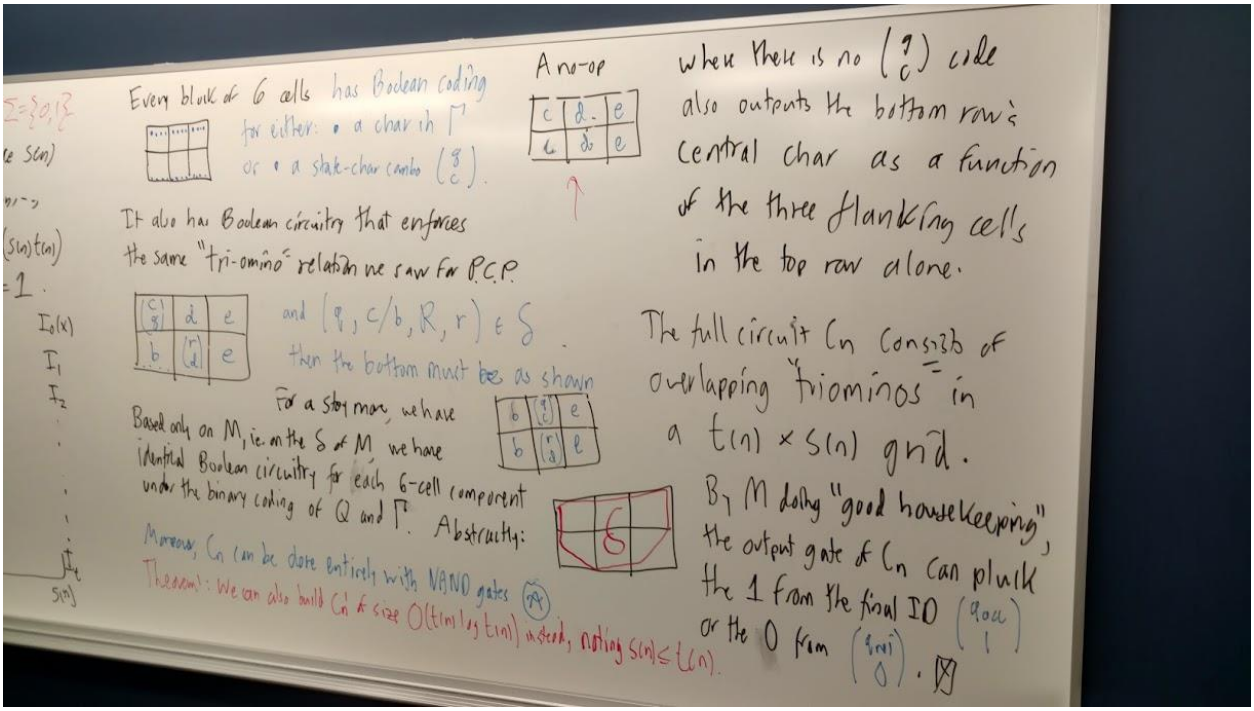
CSE596 Lecture Wed. Oct. 23





Toronto "Software Can Be Burned Into Hardware"

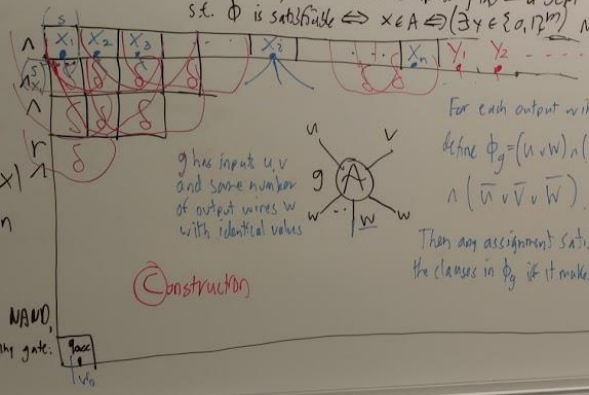
Steve-Cook-Levin Theorem: For any $A \in NP$, $A \leq_m^p 3SAT$.

Leonid 1970-71 1969?-1973 Stronger Theorem.

in the USSR, now Boston

Proof: We can take a 1-tape TM M that computes the Verifier relation $R(x,y)$ for A in some polynomial time t of $n+m$ (where $n=|x|$, $m=|y|$) where $m = p(|x|)$ for some polynomial $p(n)$. Given any x, we want to compute $f(x) =$ a 3CNF formula ϕ_x s.t. ϕ is satisfiable $\Leftrightarrow x \in A \Leftrightarrow (\exists y \in \{0,1\}^m)$ M accepts $\langle x, y \rangle$.

Proof Sketch



Take $n = |x|$ and build C_n for M as before.

Every gate is a NAND, so consider any gate:

g has inputs u, v and some number of output wires w with identical values

For each output wire w, define $\phi_g = (u \vee w) \wedge (v \vee w) \wedge (\bar{u} \vee \bar{v} \vee \bar{w})$.

Then any assignment satisfies the clauses in ϕ_g iff it makes $w = u$ NAND v.

Construction

For the formula ϕ_x, we first allocate variables

$x_1, \ldots, x_n$ for the x-inputs

$y_1, \ldots, y_m$ for the y-inputs

w_0 for the output wire

$u, v, w, w' \ldots$ for all other wires

In the illustration of a gate g, u and/or v could be an x_i-input or y_j-input. Use the same variable x_i for every wire out of the input x_i.

We have: After substituting the bit values of the given x for the variables $x_1, \ldots, x_n$, we have $x \in A \Leftrightarrow$ there exists an assignment to $y_1 \ldots y_m$ such that: all other wire variables have values forced by the gates as shown and w_0 gets value 1. Then ϕ_x will have the form

$$\phi_x = (\text{clauses that force the input substitutions for the given } x) \wedge (w_0) \wedge \bigwedge_{\text{gates } g} \phi_g$$

clauses that force all the NAND gates to function correctly.

These can also be singleton clauses (x_i) if bit $x_i = 1$, else $(\bar{x}_i)$.

(C) Correctness. Thus $x \in A \Leftrightarrow (\exists y)(C_n(|x,y|) = 1$ $\Leftrightarrow$ (∃ an assignment to y and wire variables) $\phi_x(\vec{a}) = 1$ $\Leftrightarrow \phi_x \in 3SAT$.

Finally, ϕ_x is produced in $O(t(n) \cdot s(n)) = O(t(n)^2)$ = polynomial time by simple translation of the gates and wires in C_n. Complexity [X]

The full circuit C_n consists of overlapping "triominos" in a $t(n) \times s(n)$ grid.

By M doing "good housekeeping", the output gate of C_n can pluck the 1 from the final IO $\begin{pmatrix} q_{acc} \\ 1 \end{pmatrix}$ or the 0 from $\begin{pmatrix} q_{rej} \\ 0 \end{pmatrix}$. [X]