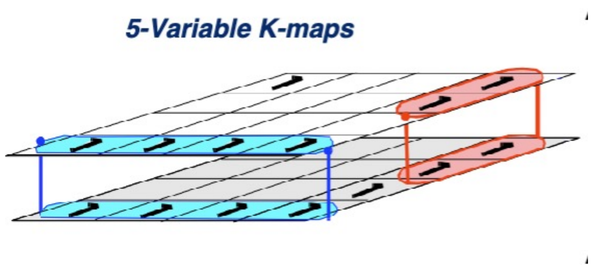
Parallel Quine-McCluskey Algorithm

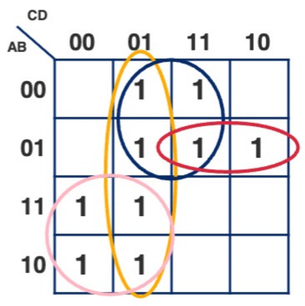
Andrew Wang & Emily Allendorf

Motivation: Boolean Expression Reduction

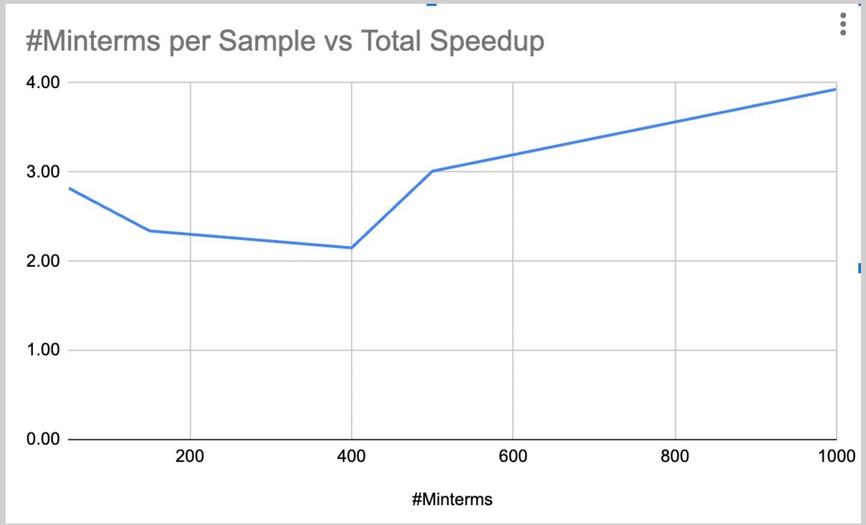
- Fundamental process in the design and optimization of digital circuits.
- Minimizes the number of gates required, reducing complexity, power consumption, and overall cost.
- Critical for hardware description compilation and FPGA Synthesis where physical resource constraints make optimization vital.

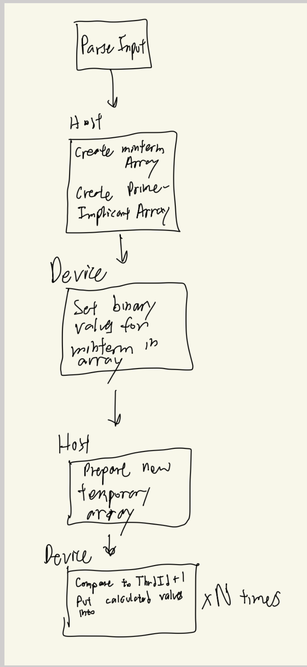
Karnaugh Maps Visualization





Quine-McClusky Algorithm in CUDA

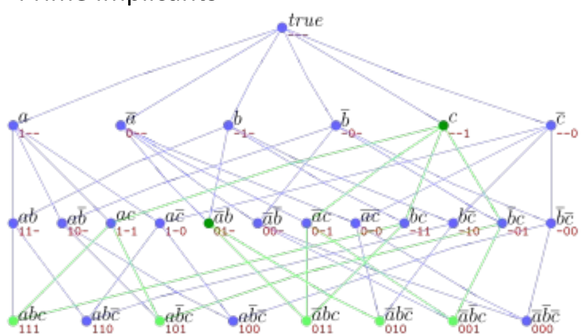




Quine-McCluskey Algorithm

- List all min-terms
- Group all min-terms by the number of ones
- Compare every term with the terms in its adjacent group
  - replace “one-off” entries with “_” and record the min-terms involved
- Repeat pairing process to get the prime implicants
- Select the minimal covering set to get the essential implicants

Prime Implicants



Essential Implicants

Prime implicant	Minterm 0	2	5	6	7	8	9	13
$p_1 = 0\ 0\ x\ 0$	✓	✓						
$p_2 = 0\ x\ 1\ 0$		✓		✓				
$p_3 = 0\ 1\ 1\ x$				✓	✓			
$p_4 = x\ 0\ 0\ x$	✓					✓	✓	
$p_5 = x\ x\ 0\ 1$			✓				✓	✓
$p_6 = 1\ x\ 0\ x$						✓	✓	✓
$p_7 = x\ 1\ x\ 1$			✓		✓			✓