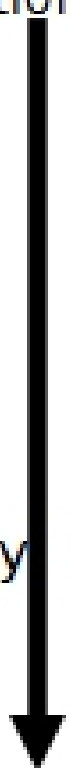


# Evolution of Implementation Technologies

- Discrete devices: relays, transistors (1940s-50s)
  - Discrete logic gates (1950s-60s)
  - Integrated circuits (1960s-70s)
    - e.g. TTL packages: Data Book for 100's of different parts
    - Map your circuit to the Data Book parts
  - Gate Arrays (IBM 1970s)
    - "Custom" integrated circuit chips
    - Design using a library (like TTL)
    - Transistors are already on the chip
    - Place and route software puts the chip together automatically
    - + Large circuits on a chip
    - + Automatic design tools (no tedious custom layout)
    - - Only good if you want 1000's of parts
- trend toward  
higher levels  
of integration
- 

# Gate Array Technology (IBM - 1970s)

## Simple logic gates

- Use transistors to implement combinational and sequential logic

## Interconnect

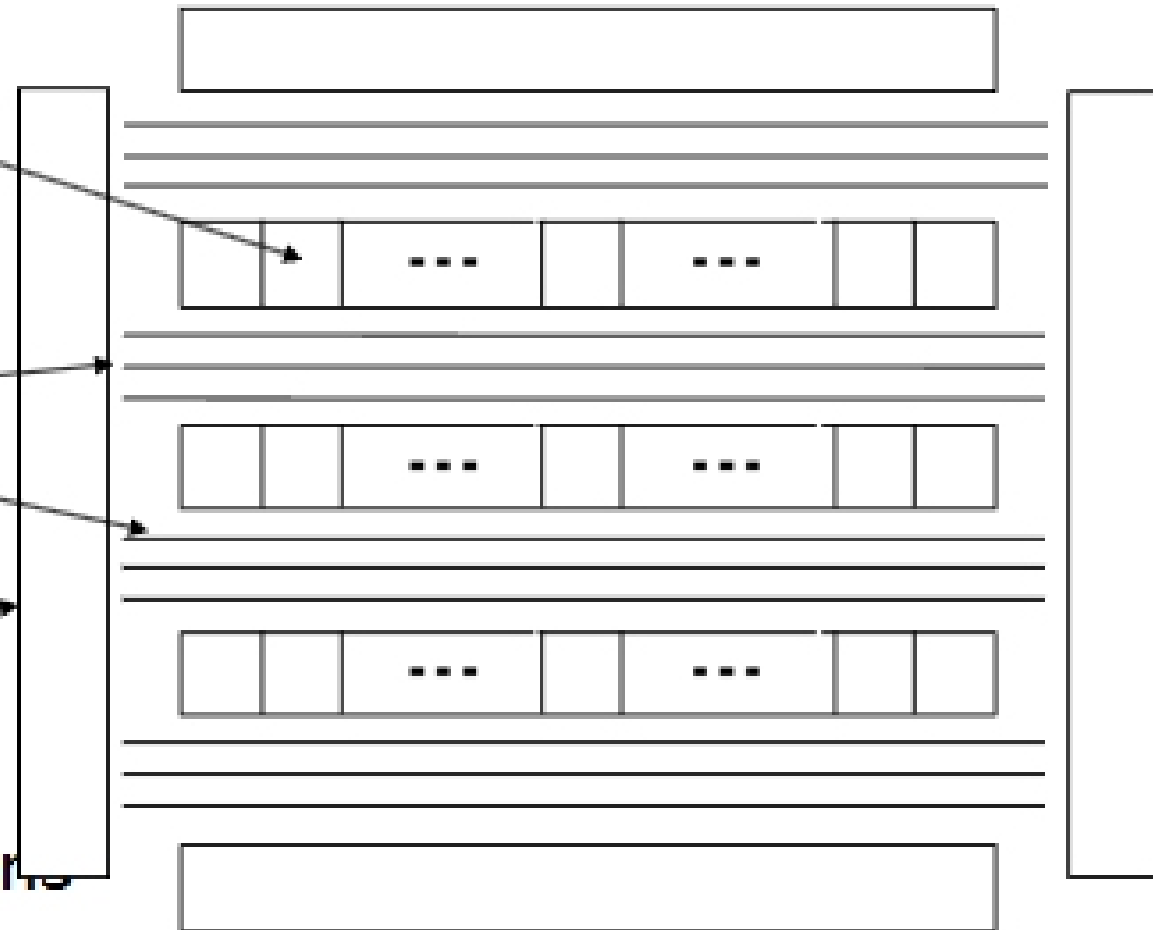
- Wires to connect inputs and outputs to logic blocks

## I/O blocks

- Special blocks at periphery for external connections

## Add wires to make connections

- Done when chip is fabbed
  - "mask-programmable"
- Construct any circuit



# Programmable Logic



- Disadvantages of the Data Book method
  - Constrained to parts in the Data Book
  - Parts are necessarily small and standard
  - Need to stock many different parts
- Programmable logic
  - Use a single chip (or a small number of chips)
  - Program it for the circuit you want
  - No reason for the circuit to be small