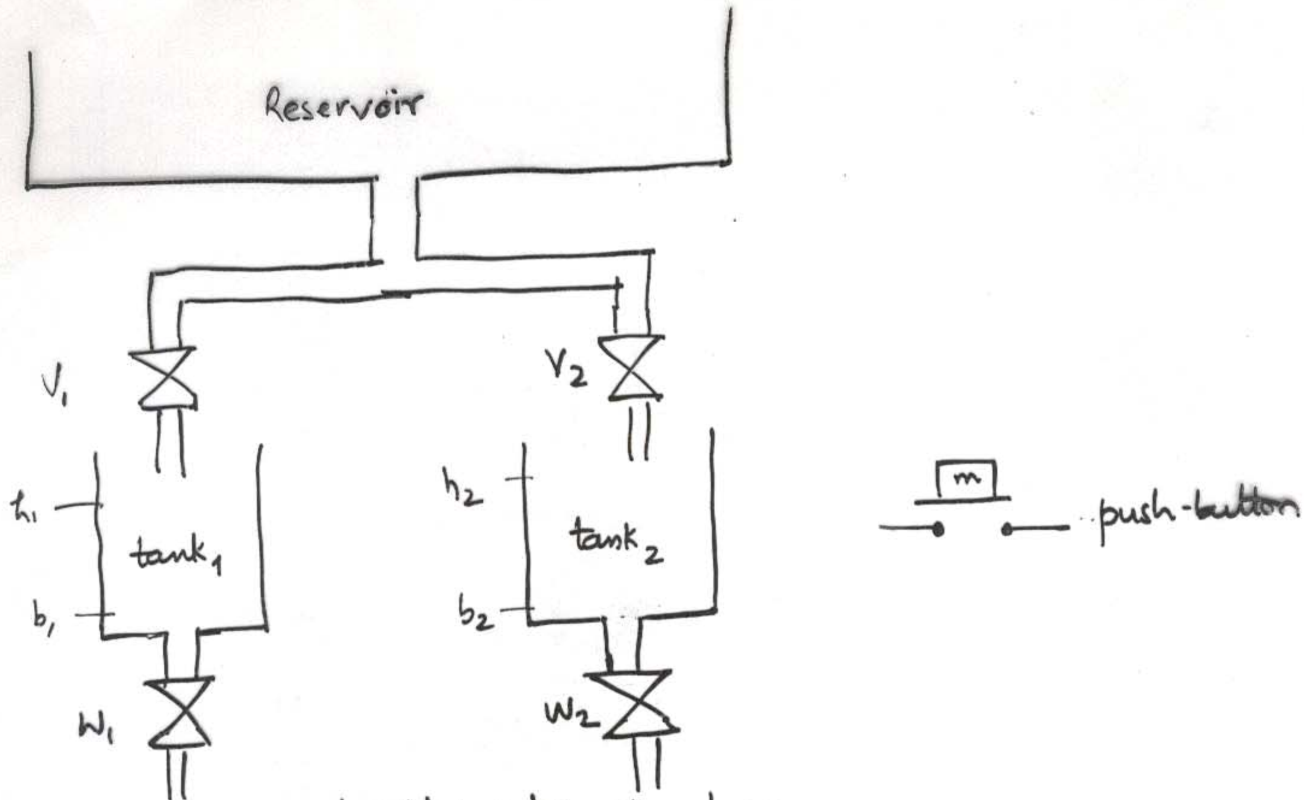
Logic Control (Example)



* State aggregation $\begin{cases} height_i < b_i & \Rightarrow low_i \\ height_i > h_i & \Rightarrow high_i \\ \text{otherwise} & \Rightarrow medium_i \end{cases}$

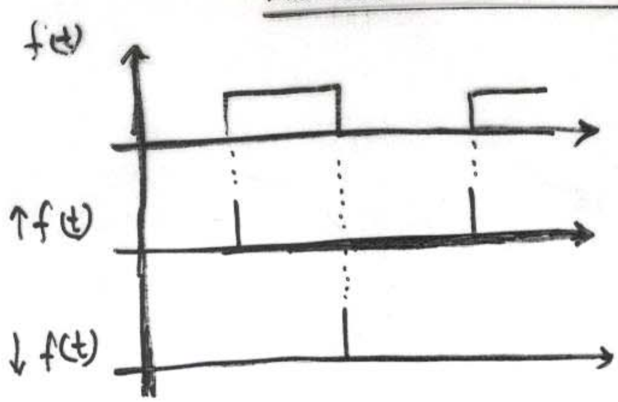
* level sensors : $low_i \Leftrightarrow b_i = 0$
  $high_i \Leftrightarrow h_i = 1$

* valve actuators: V_i, $W_i = 1 \Leftrightarrow$ on; push button actuator : m

Control spec.

* Initially tanks low
- m pushed ⇒ open both valves
- tank$_i$ full ⇒ close valve V_i, open valve W_i
- tank$_i$ low ⇒ close valve W_i
- Tanks low, m pushed again ⇒ resume filling both tanks

Notation and Signals

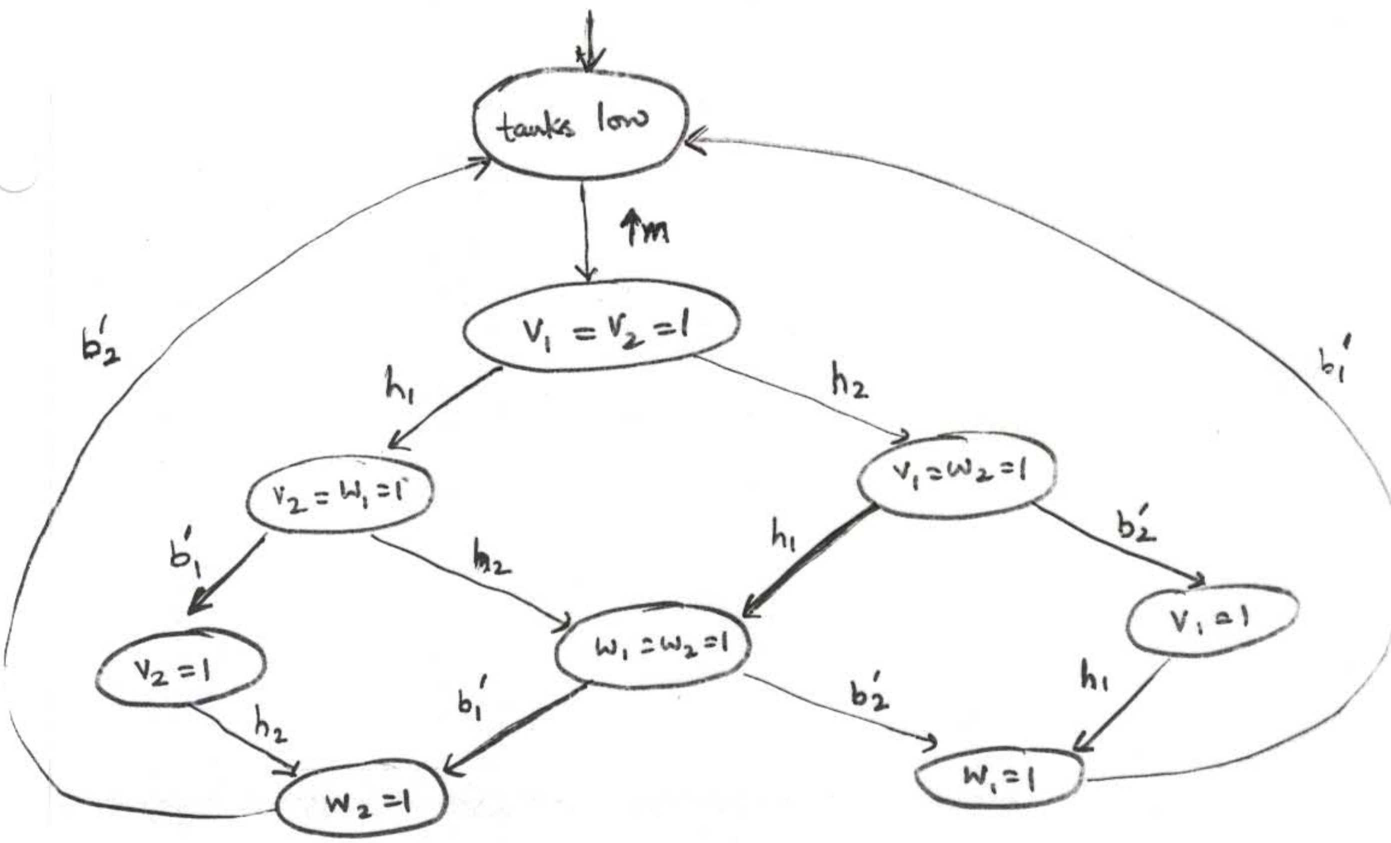


f : boolean signal

$\uparrow f$; $\downarrow f$: event signals

__Definition__: $\uparrow a + \uparrow b$, $a . \uparrow b$, $\uparrow a . \uparrow b$ etc.

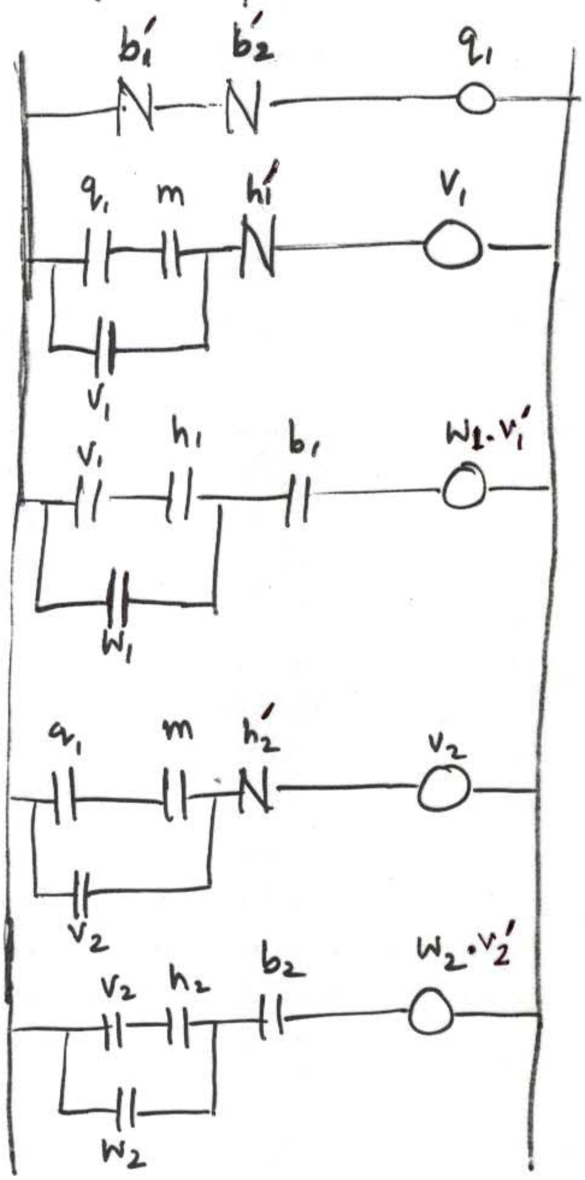
State machine / Automaton representation



- Each node represents a state :
  $w_2 = 1$ node : tank 1 low, tank 2 medium or high
- Each arc represents a transition of state on event label

Relay Ladder logic. Description of Controller

- RLL graphical language for encoding a logic/automaton in PLC
- Each "rung" corresponds to a Boolean equation
- pair of vertical bars represent rung inputs; rung output represented by circle (physical relays)
- crossed bar represents Boolean complement
- AND / OR by series/parallel placement of bars



$q_1 = b_1' b_2'$

$v_1 = (q_1 m + v_1)\, h_1'$

$v_1' \cdot w_1 = (v_1 h_1 + w_1)\, b_1$

$v_2 = (q_1 m + v_2)\, h_2'$

$v_2' \cdot w_2 = (v_2 h_2 + w_2)\, b_2$

- Any automaton can be encoded by RLL
- Limited by limited descriptive power of automata
- RLL encoding may not be directly obvious from control spec.