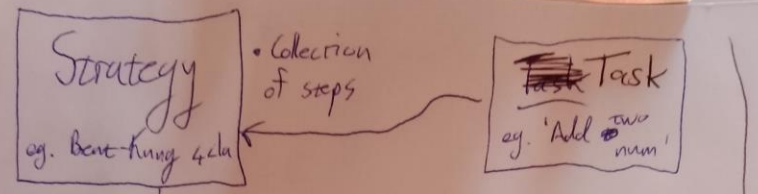
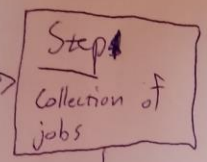
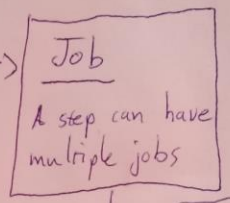
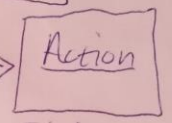




eg. 'Lay out B'
'Calculate P_0'





Job has a serial number of actions to ~~comp~~ perform
eg.
1. Transit to grain
2. Pick up grain
3. Transit to some cell.
4. Drop grain.

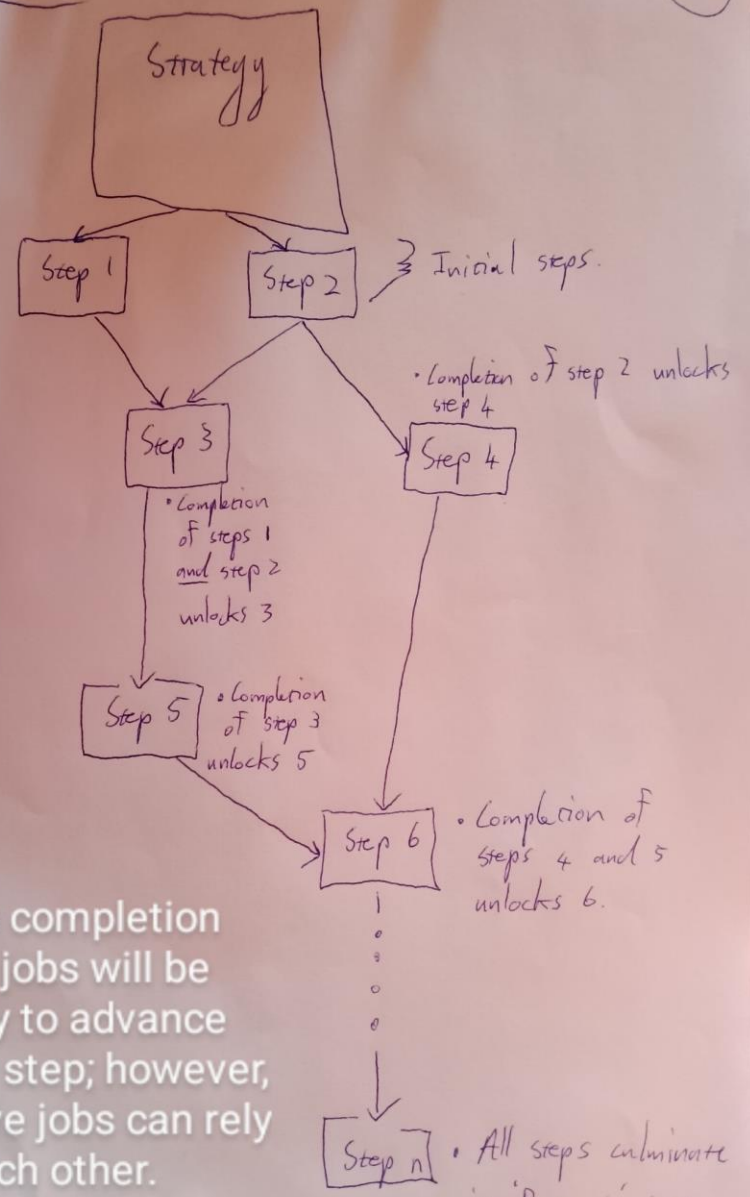
↓

Job complete.

Notes:

1. Completion of a step can emit more steps should they depend on results/prerequisites from others.
2. Completion of a job in a step can emit other jobs;
3. A job's actions are strictly sequential, completion of one action unlocks the next in a chain.

Dependency example:



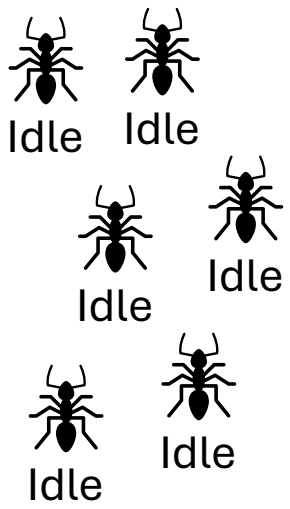
Note: the completion of some jobs will be necessary to advance through the step; however, no two active jobs can rely on each other.

Example of Task progression:

Board snapshot 1 (not started)

Component: Decimal input display	
A	~
B	~

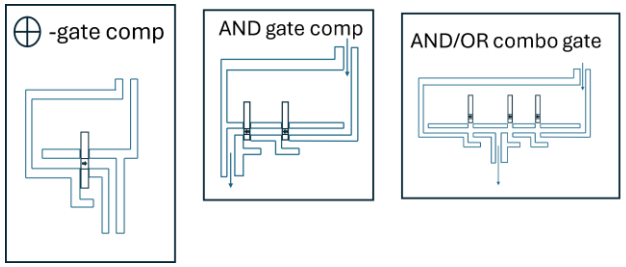
Component: Input tracks			



Component: Job Board

Steps available:
- None

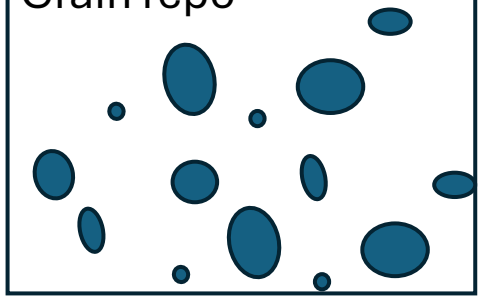
Steps complete:
- None



Component: Memory Table

HUMAN READABLE										Value	Reserved	Read ready
	G	P	C	S	4	3	2	1	0			

Component:
Grain repo

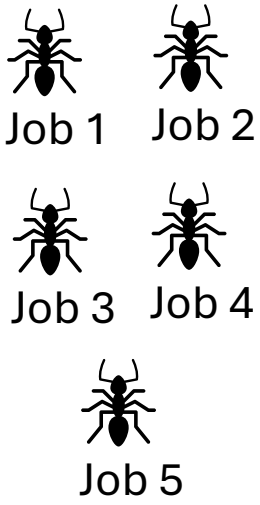


Example of Task progression:

Board snapshot 2 (numbers submitted)

Component: Decimal input display	
A	11
B	3

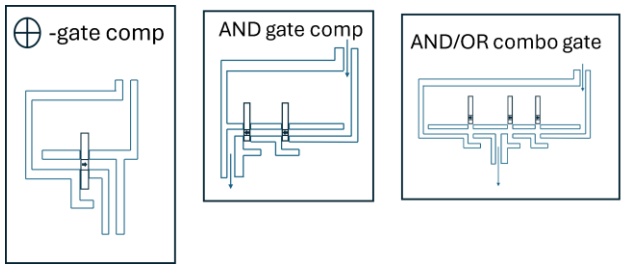
Component: Input tracks			



Component: Job Board
1. Place grain in A_0 (assigned)
2. Place grain in A_1 (assigned)
3. Place grain in A_3 (assigned)
4. Place grain in B_0 (assigned)
5. Place grain in B_1 (assigned)

Steps available:
- Lay out input A
- Lay out input B

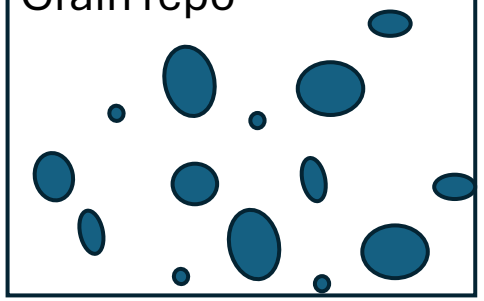
Steps complete:
- None



Component: Memory Table

HUMAN READABLE										Value	Reserved	Read ready
	G	P	C	S	4	3	2	1	0			

Component: Grain repo



Example of Task progression:

Board snapshot 3 (calculation in progress)

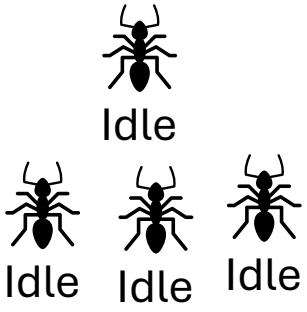
Component: Decimal input display	
A	11
B	3

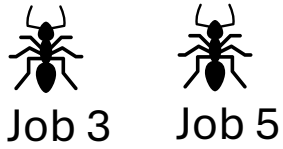
	Component: Input tracks			
A			•	•
B				•

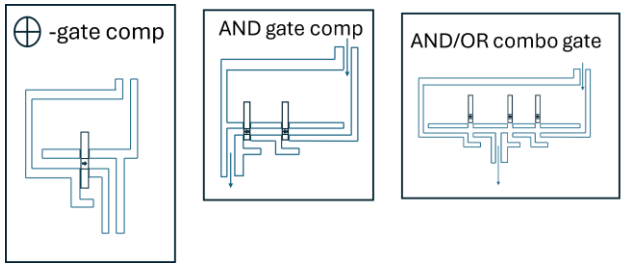
Steps available:
- Lay out input A
- Lay out input B
- Calculate P_i values (a pair (A_i , B_i available)

Steps complete:
- None

Component: Job Board
3. Place grain in A_3 (assigned)
5. Place grain in B_1 (assigned)
6. Place grain in Row 1 ‘reserved’
7. Place grain in XOR input 1
8. Place grain in XOR input 2



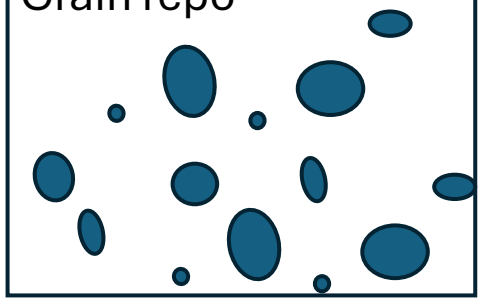




Component: Memory Table

HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready

Component: Grain repo



Example of Task progression:

Board snapshot 4 (calculation in progress)

Steps available:

- Lay out input A
- Calculate P_0 (pair (A_0 , B_0) available)
- Other P values…

Steps complete:

- Lay out input B

Component: Job Board
3. Place grain in A_3 (assigned)
6. Place grain in Row 1 ‘reserved’ (assigned)
7. Place grain in XOR input 1 (assigned)
8. Place grain in XOR input 2 (assigned)

Job 3

	Component: Input tracks			
A			•	•
B			•	•

Idle

Idle

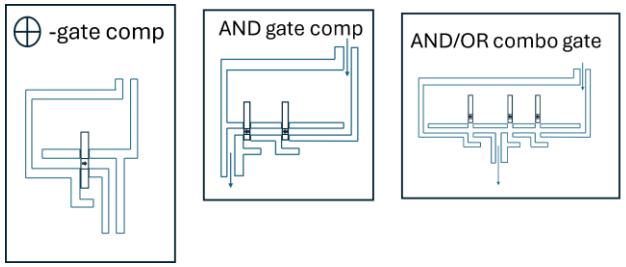
Job 6

Job 7

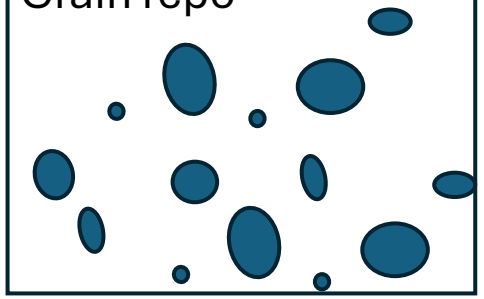
Job 8

Component: Memory Table

HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready



Component: Grain repo



Example of Task progression:

Board snapshot 5 (calculation in progress)

Steps available:

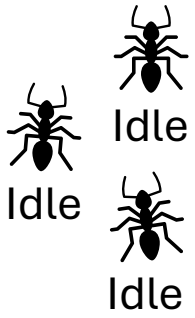
- Calculate P_0 (pair (A_0 , B_0) available)
- Other avail P val…

Steps complete:

- Lay out input B
- Lay out input A

Component: Job Board
8. Place grain in XOR input 2 (assigned)
9. Place grain in Row 1 ‘P’ slot (assigned)
10. Place grain in Row 1 ‘0’ slot (assigned)

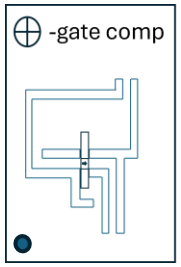
	Component: Input tracks			
A	●		●	●
B			●	●

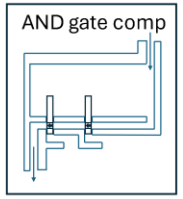


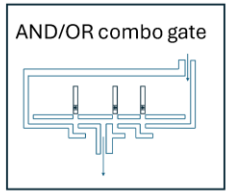
Component: Memory Table

HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready
											●	

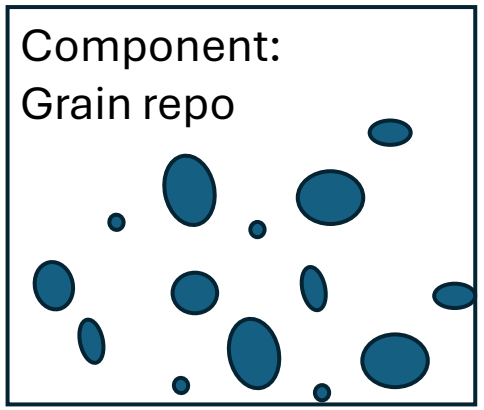

Job 8







Component: Grain repo



Example of Task progression:

Board snapshot 6 (calculation in progress)

Steps available:

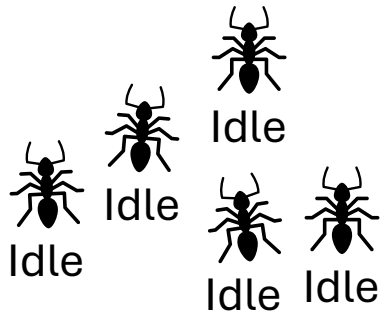
- Calculate P_0 (pair (A_0 , B_0) available)
- Other avail P val…

Steps complete:

- Lay out input B
- Lay out input A

Component: Job Board
10. Place grain in Row 1 ‘0’ slot (assigned)

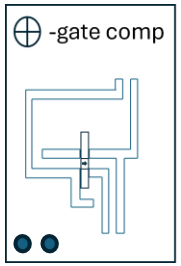
	Component: Input tracks			
A	•		•	•
B			•	•

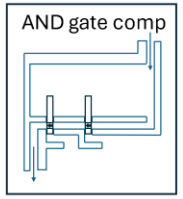


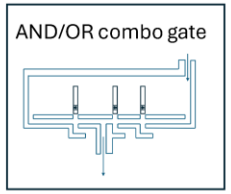

Job 10

Component: Memory Table

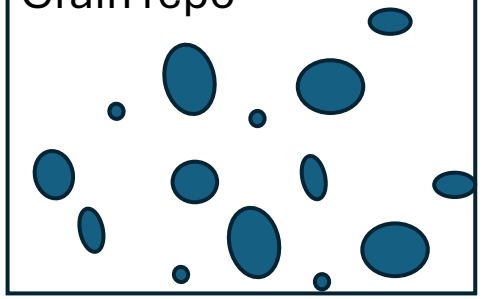
HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready
		•									•	







Component:
Grain repo



Example of Task progression:

Board snapshot 7 (calculation in progress)

Steps available:

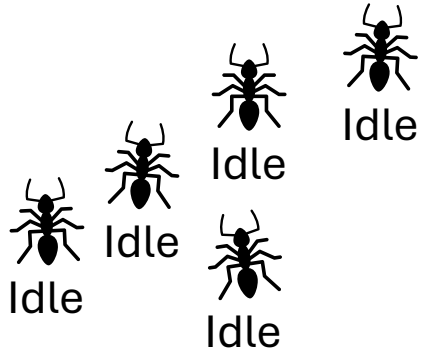
- Calculate P_0 (pair (A_0 , B_0) available)
- Other avail P val…

Steps complete:

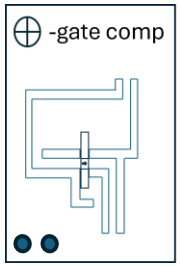
- Lay out input B
- Lay out input A

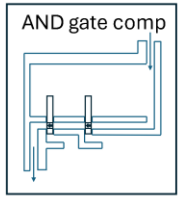
Component: Job Board
11. Operate XOR gate (assigned)

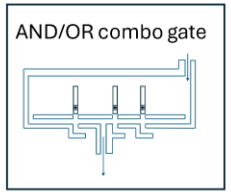
	Component: Input tracks			
A	•		•	•
B			•	•



Job 11



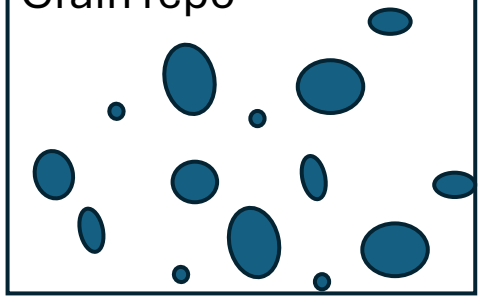




Component: Memory Table

HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready
P_0		•							•		•	

Component:
Grain repo



Example of Task progression:

Board snapshot 8 (calculation in progress)

Steps available:

- Calculate P_0 (pair (A_0 , B_0) available)
- Other avail P val…

Steps complete:

- Lay out input B
- Lay out input A

Idle

Idle

Idle

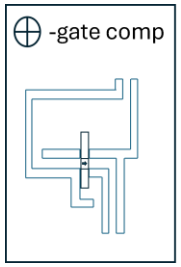
	Component: Input tracks			
A	•		•	•
B			•	•

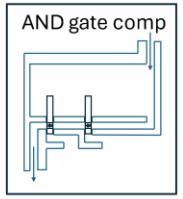
Component: Job Board
12. Record XOR result -> 0; place grain in Row 1 ‘read ready’ as value = 0 means no grain in ‘value’ column (assigned) (result 1 would place grain in ‘value’ and only then read-ready)
13. Clear XOR input 1 (assigned)
14. Clear XOR input 2 (assigned)

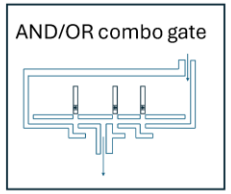
Job 12

Job 14

Job 13



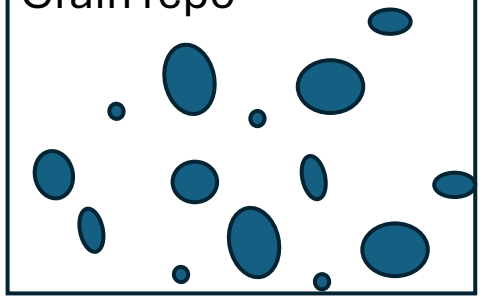




Component: Memory Table

HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready
P_0		•							•		•	

Component:
Grain repo



Example of Task progression:

Board snapshot 9 (calculation in progress)

Steps available:

- Other avail P val…
- P_0 read-ready; calculate G_0

Steps complete:

- Lay out input B
- Lay out input A
- Calculate P_0 (pair (A_0 , B_0) available)

Component: Job Board
14. Clear XOR input 2 (assigned)
15. Reserve…
16. Another job…
17. etc…

	Component: Input tracks			
A	•		•	•
B			•	•

Idle

Job y

Job x

Job z

Job 14

Component: Memory Table

HUMAN READABLE	G	P	C	S	4	3	2	1	0	Value	Reserved	Read ready
P_0		•							•		•	•

Component: Grain repo