Klotski Solver

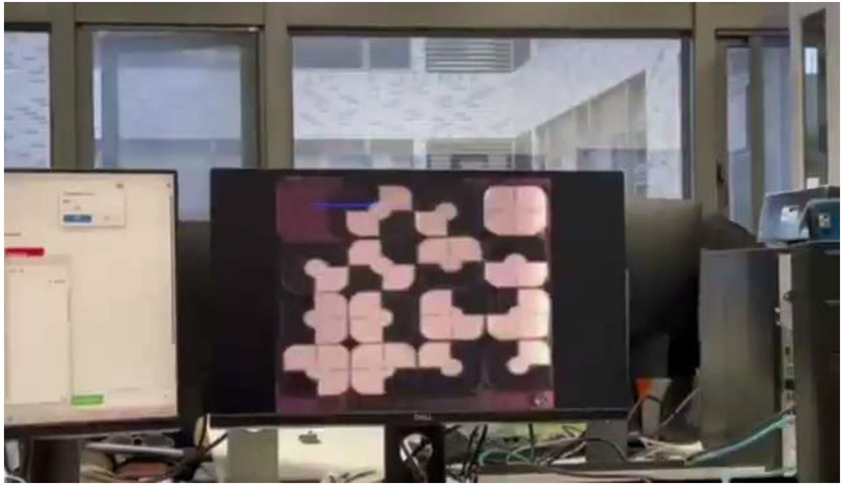
b11901148 李承彥
b11901091 鄭淳芸
b11901003 方嘉麟

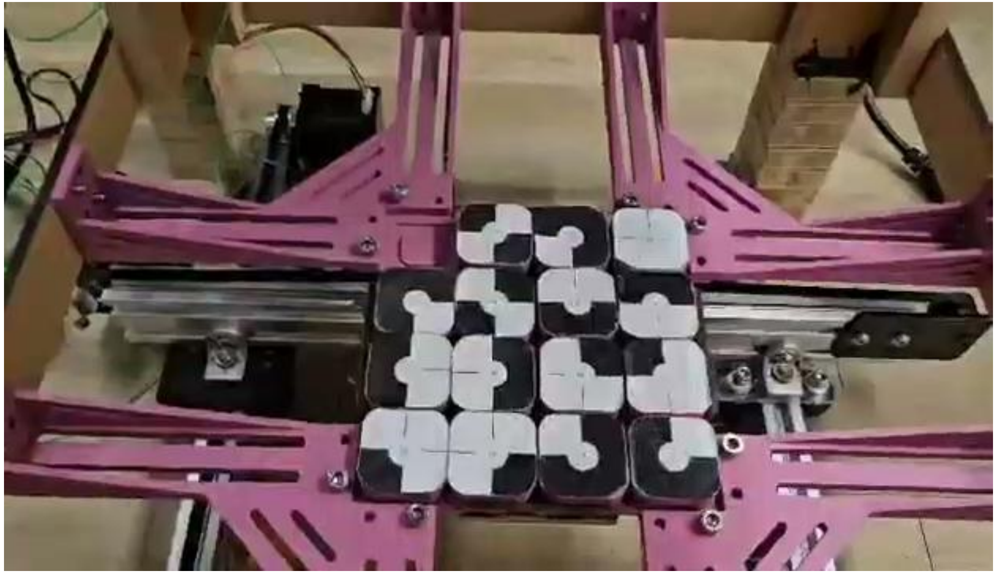
Table of Content

1. Objective
2. Algorithm
3. Architecture
4. Testing
5. Implementation

Objective

Objective





Algorithm

Overview

1	2		

Overview

1	2		
			4

Overview

1	2		3
			4

Overview

1	2		3
			4

Overview

1	2	3	4

Overview

1	2	3	4
5	6		

Overview

1	2	3	4
5	6		
			8

Overview

1	2	3	4
5	6		7
			8

Overview

1	2	3	4
5	6		7
			8

Overview

1	2	3	4
5	6	7	8

Overview

1	2	3	4
5	6	7	8
			13

Overview

1	2	3	4
5	6	7	8
9			13

Overview

1	2	3	4
5	6	7	8
9	13		

Overview

1	2	3	4
5	6	7	8
9			
13			

Overview

1	2	3	4
5	6	7	8
9			
13			14

Overview

1	2	3	4
5	6	7	8
9			
13	10		14

Overview

1	2	3	4
5	6	7	8
9			
13	10	14	

Overview

1	2	3	4
5	6	7	8
9	10		
13	14		

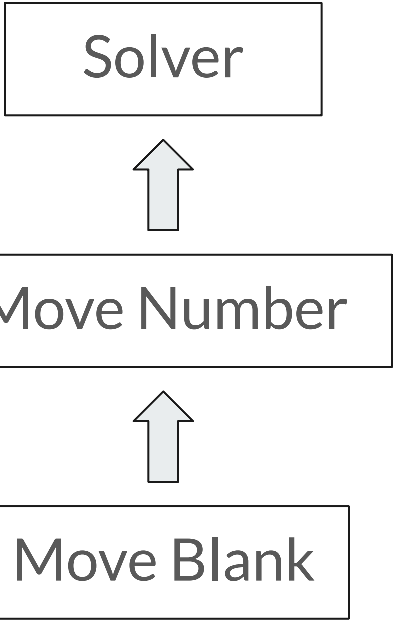
Overview

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Problem Formulation

- Instead of moving blocks, considering to move the blank.
- The blank has 4 possible moves.
- Divide and conquer by bottom-up method.

Bottom-Up Method



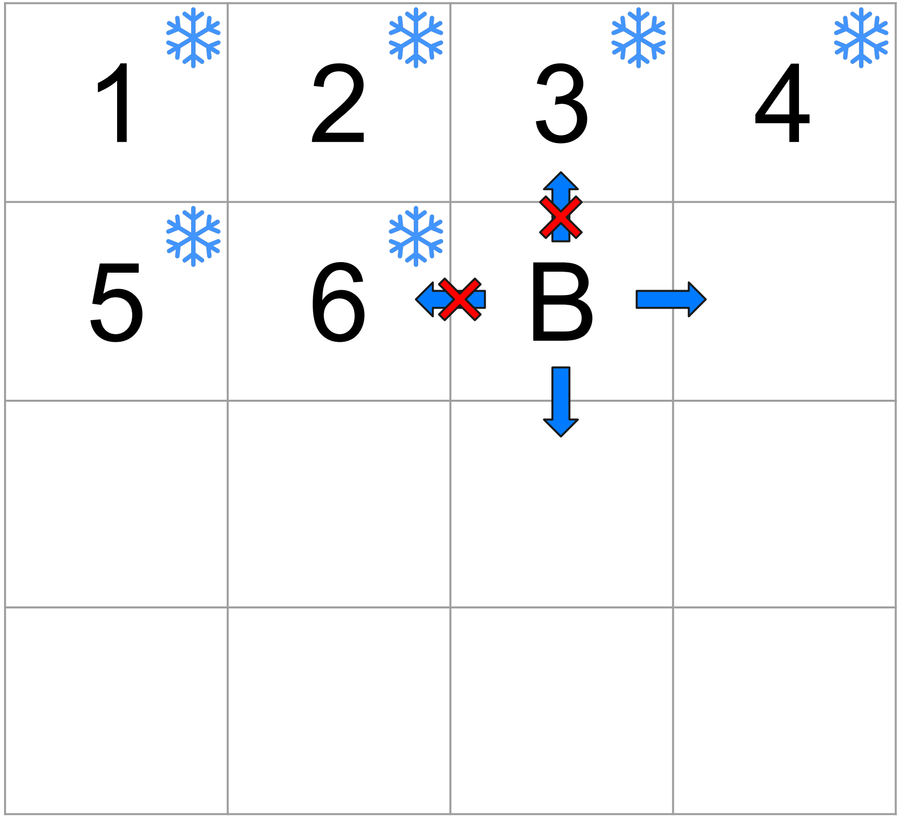
Solver

Move Number

Move Blank

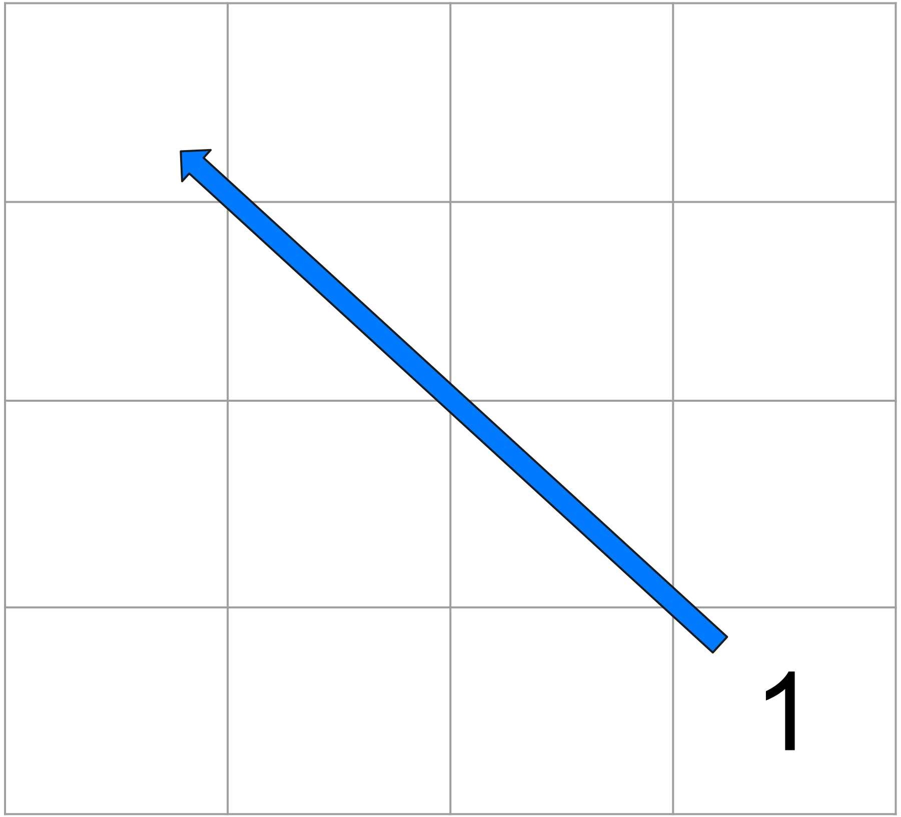
Move Blank

- 4 possible directions.
- Blocks at correct positions can’t not be moved.
  -> Using mask to freeze blocks.



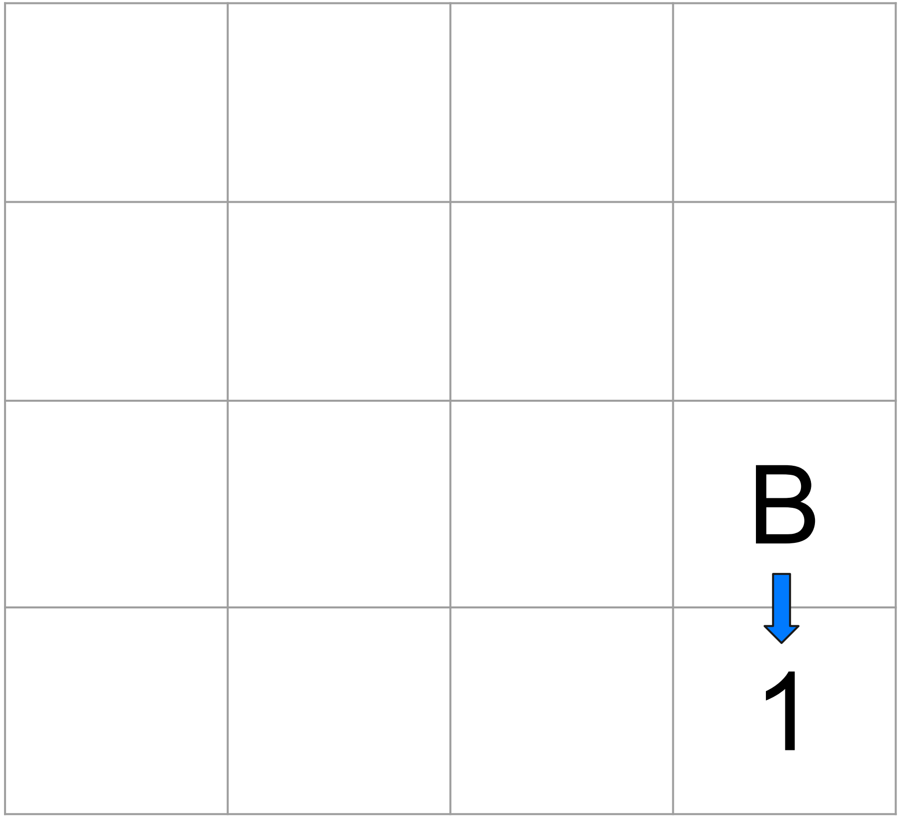
Move Number

- Reuse “Move Blank”.



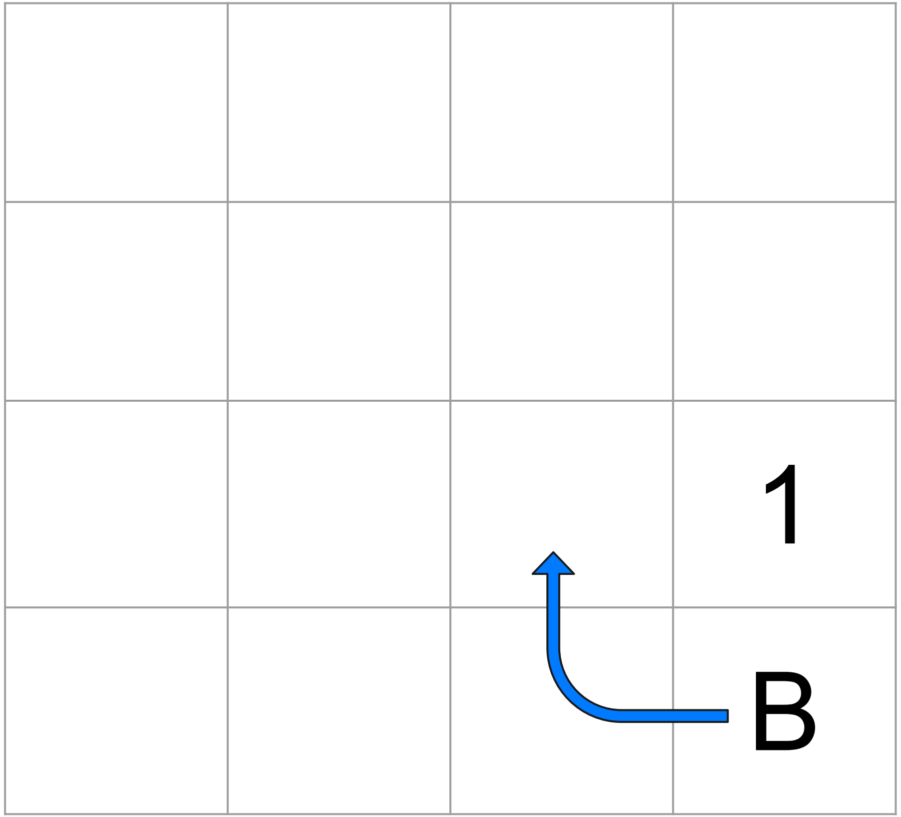
Move Number

- Reuse “Move Blank”.



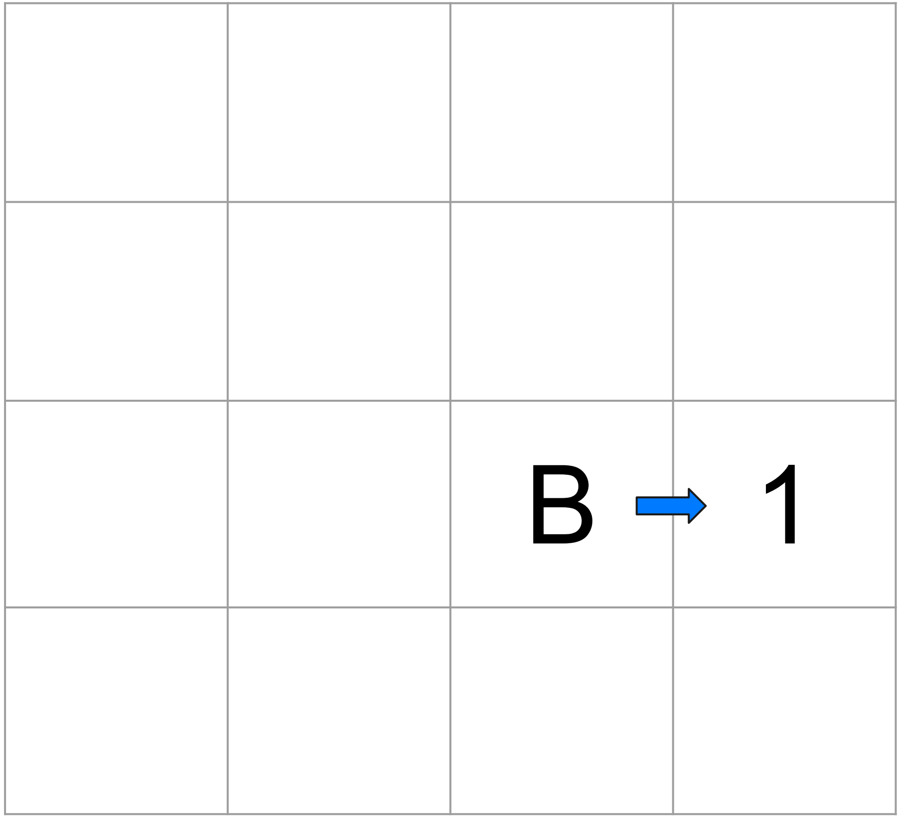
Move Number

- Reuse “Move Blank”.



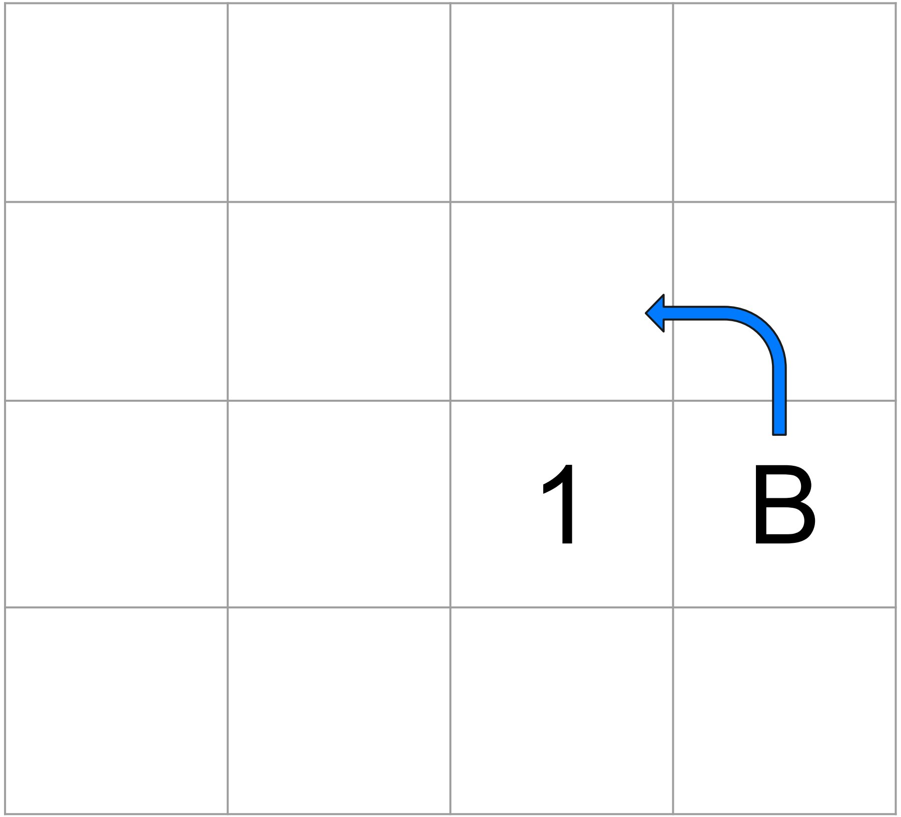
Move Number

- Reuse “Move Blank”.



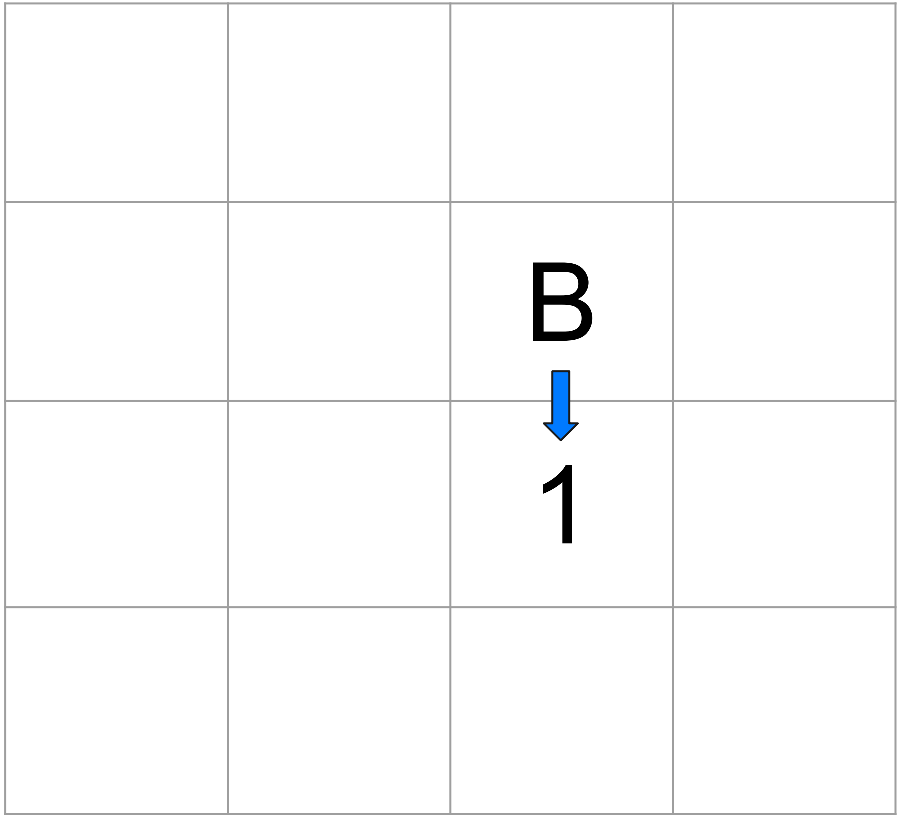
Move Number

- Reuse “Move Blank”.



Move Number

- Reuse “Move Blank”.



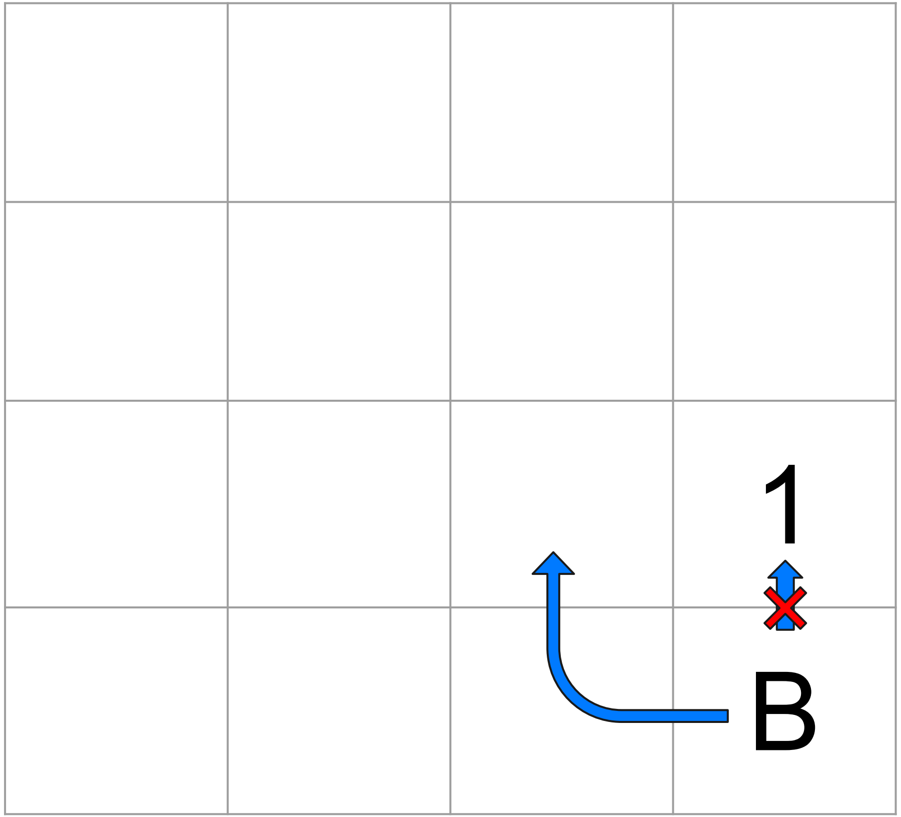
Move Number

- Reuse “Move Blank”.

		1	
		B	

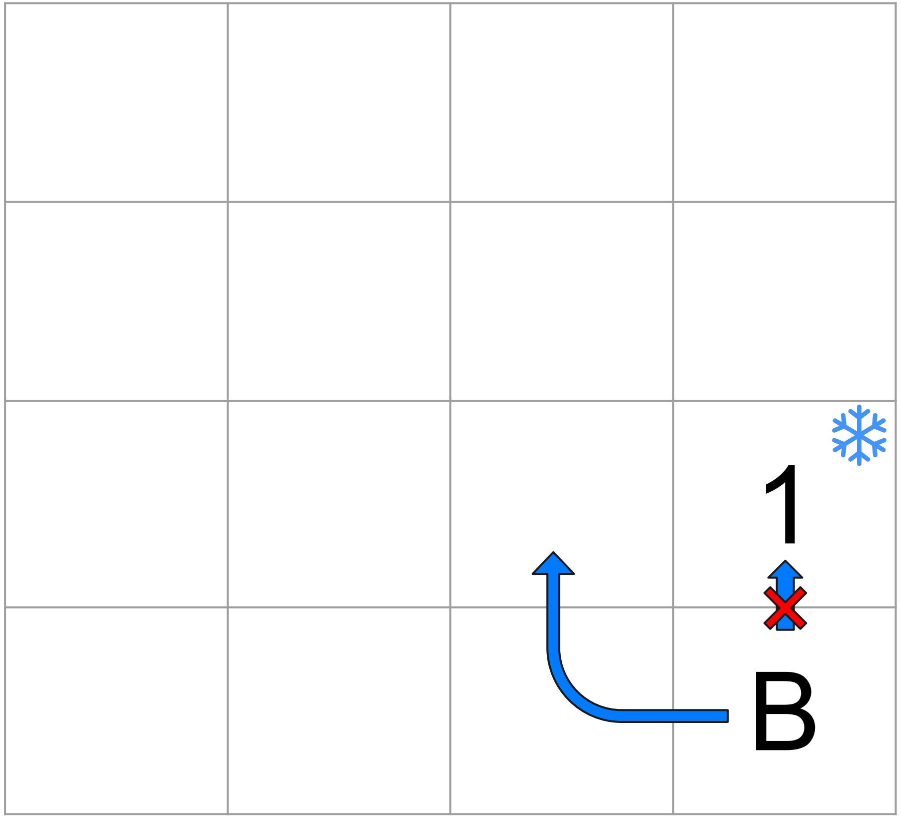
Move Number: Issue 1

- The blank may move through The target number



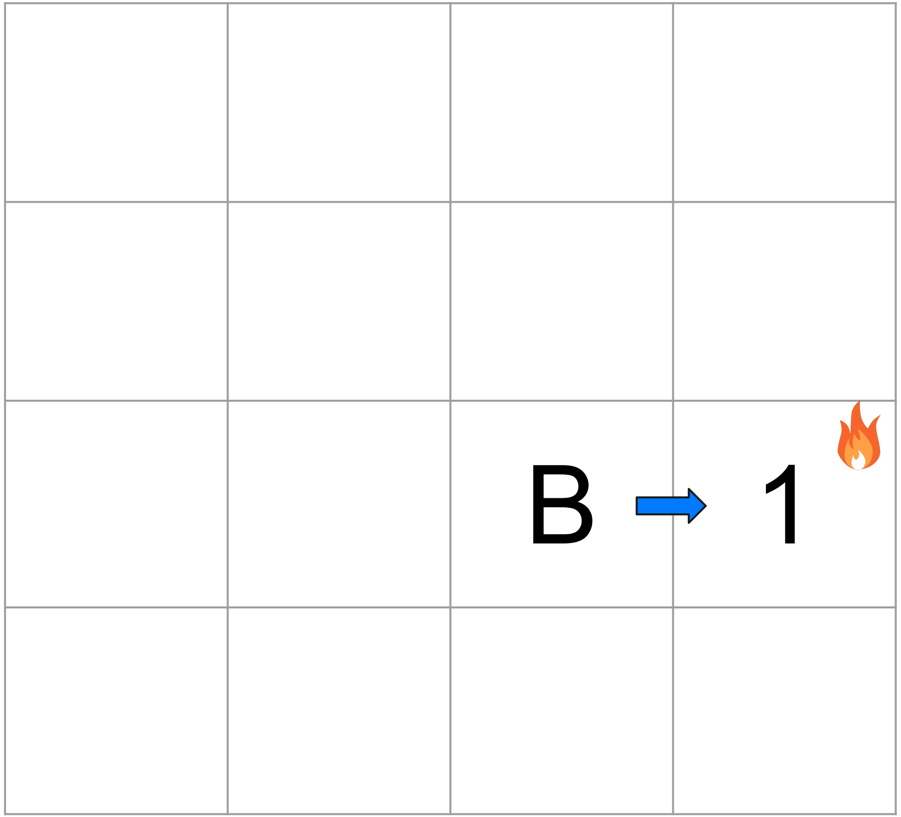
Move Number: Sol 1

- Freeze the target.



Move Number: Sol 1-1

- Freeze the target.
  Then unfreeze it.



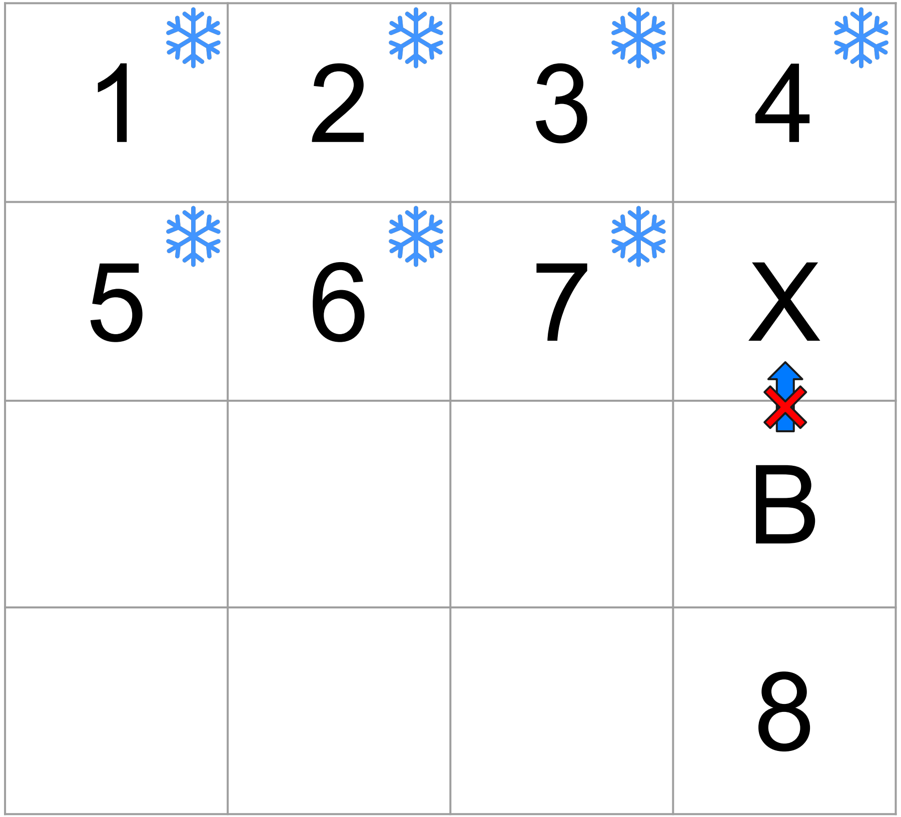
Move Number: Sol 1-1

- Freeze the target.
  Then unfreeze it.

		1	B

Move Number: Issue 2-1

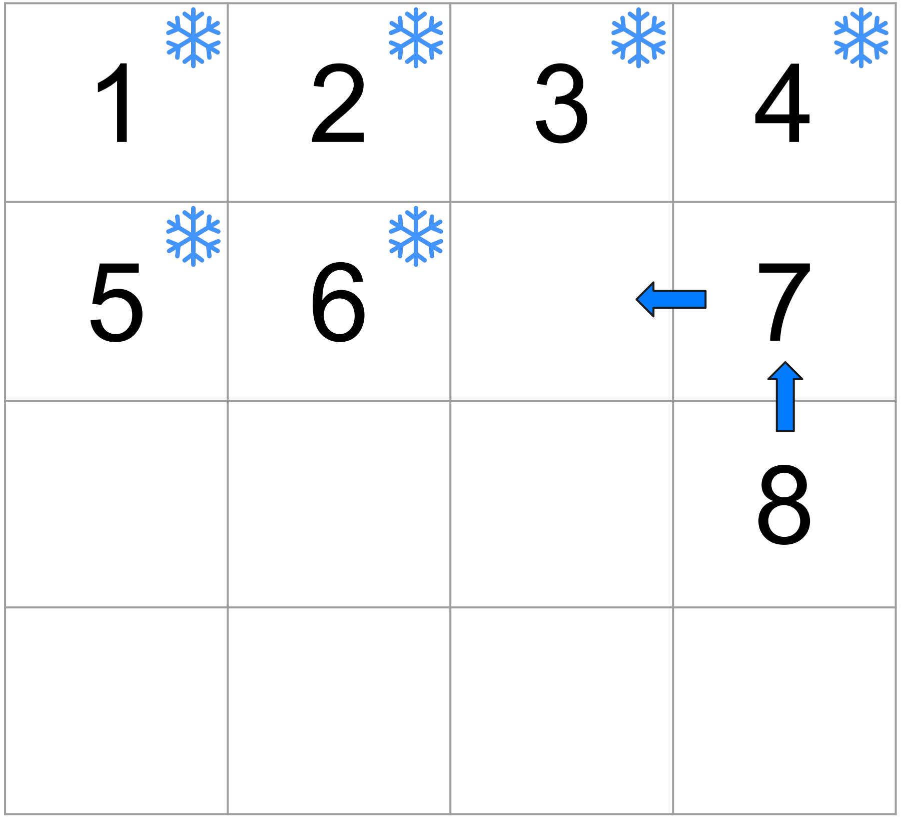
- Mask may hinder other blocks.



1 ❄	2 ❄	3 ❄	4 ❄
5 ❄	6 ❄	7 ❄	X
			B
			8

Move Number: Sol 2-1

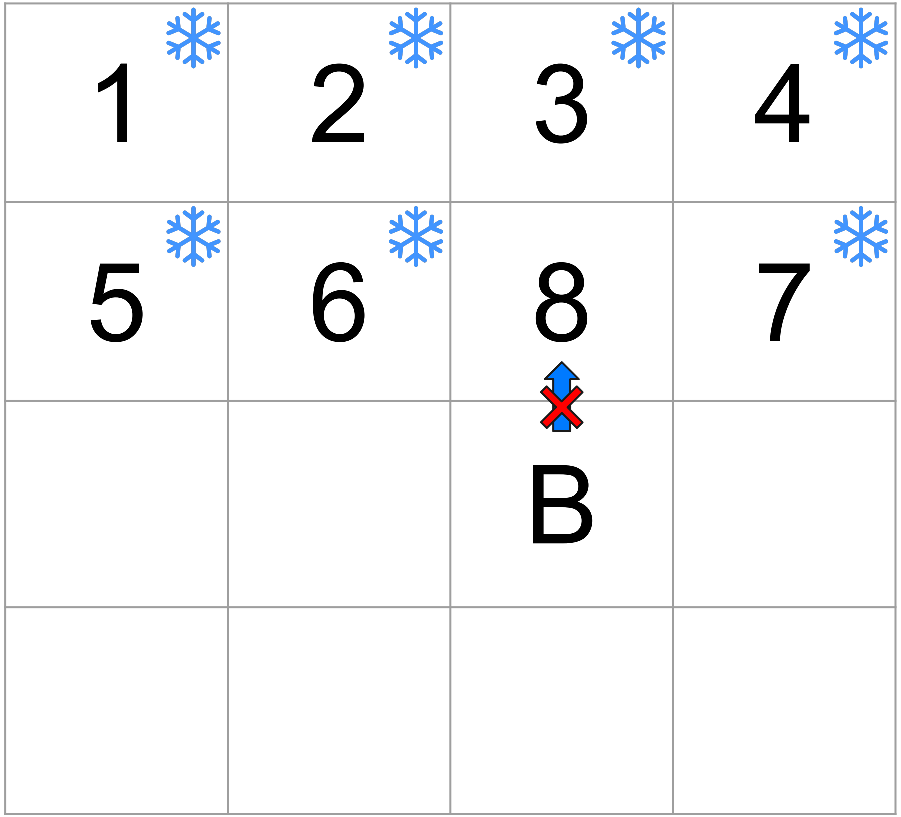
- Preserve space for others.



1 ❄	2 ❄	3 ❄	4 ❄
5 ❄	6 ❄	←	7
			↑ 8

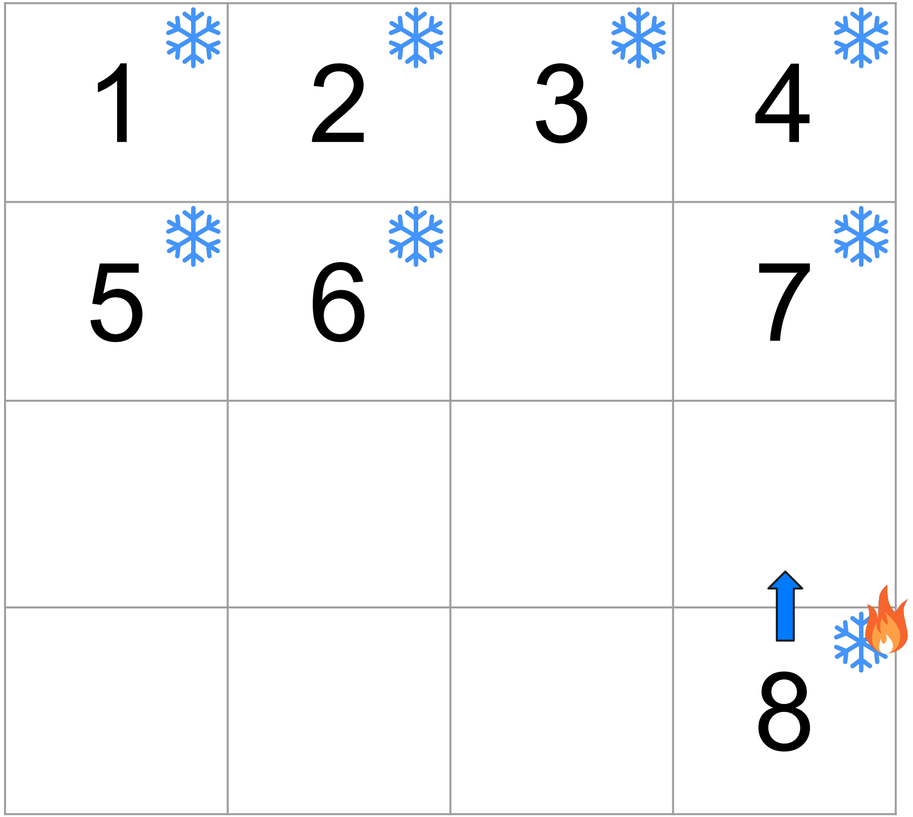
Move Number: Issue 2-2

- Blocks may be captured.



Move Number: Sol 2-2

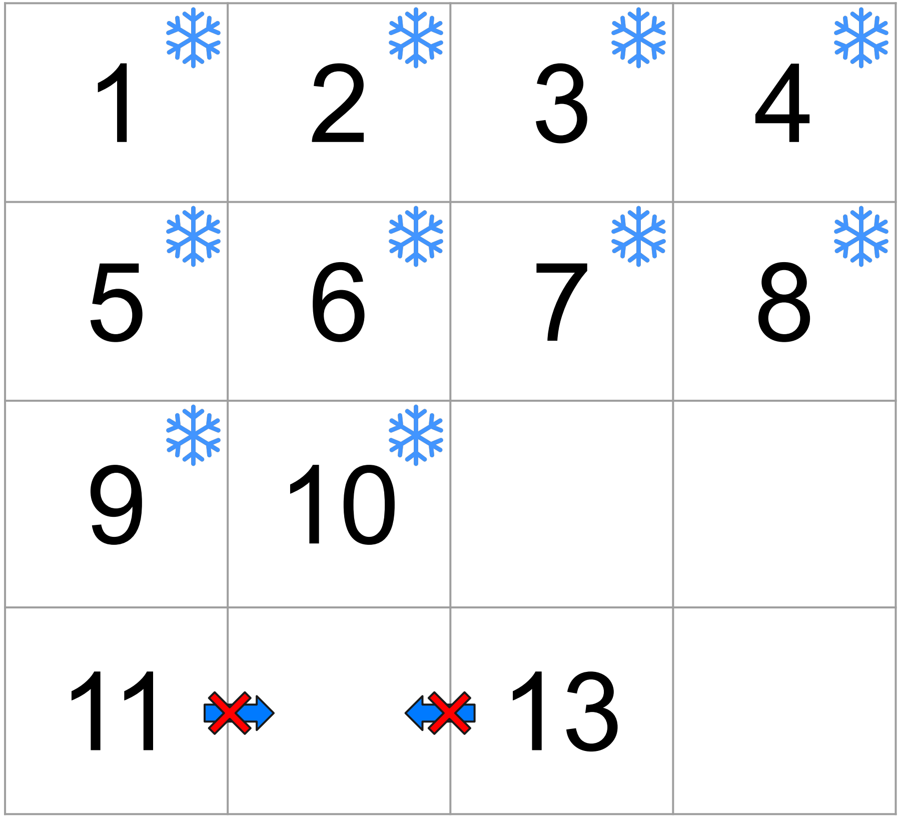
- Prevent the target blocks to be captured.



1 ❄	2 ❄	3 ❄	4 ❄
5 ❄	6 ❄		7 ❄
			⬆ ❄🔥 8

Move Number: Issue 3

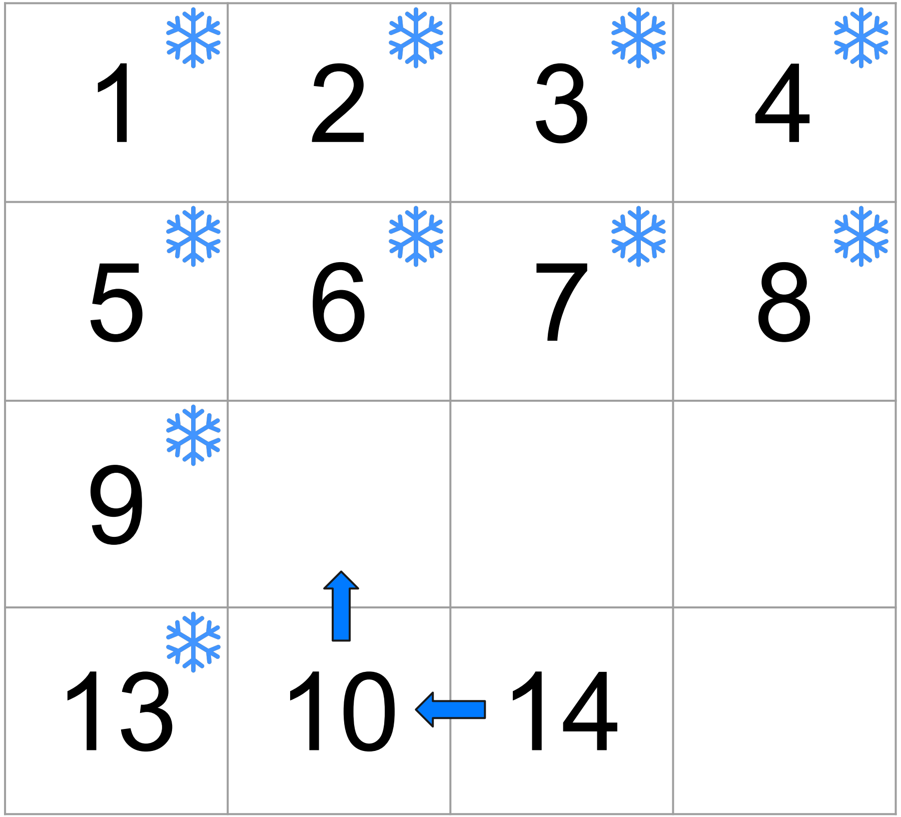
- At least two not frozen rows are needed for moving numbers.



1 ❄	2 ❄	3 ❄	4 ❄
5 ❄	6 ❄	7 ❄	8 ❄
9 ❄	10 ❄		
11 ↮		↮ 13	

Move Number: Sol 3

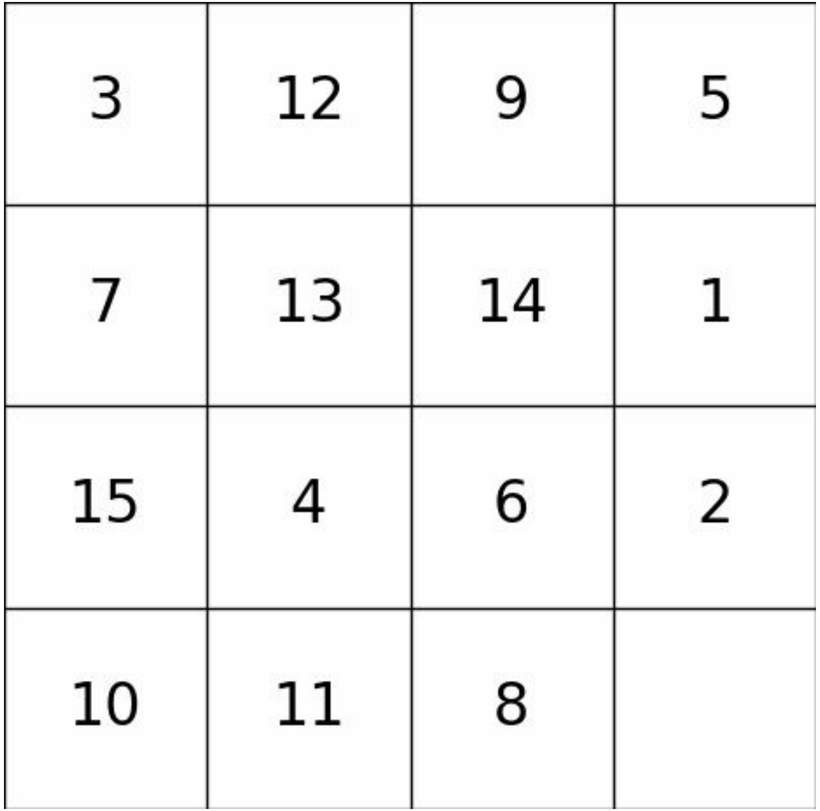
- Sort columns first.



1 ❄	2 ❄	3 ❄	4 ❄
5 ❄	6 ❄	7 ❄	8 ❄
9 ❄	↑		
13 ❄	10 ←	14	

Solver

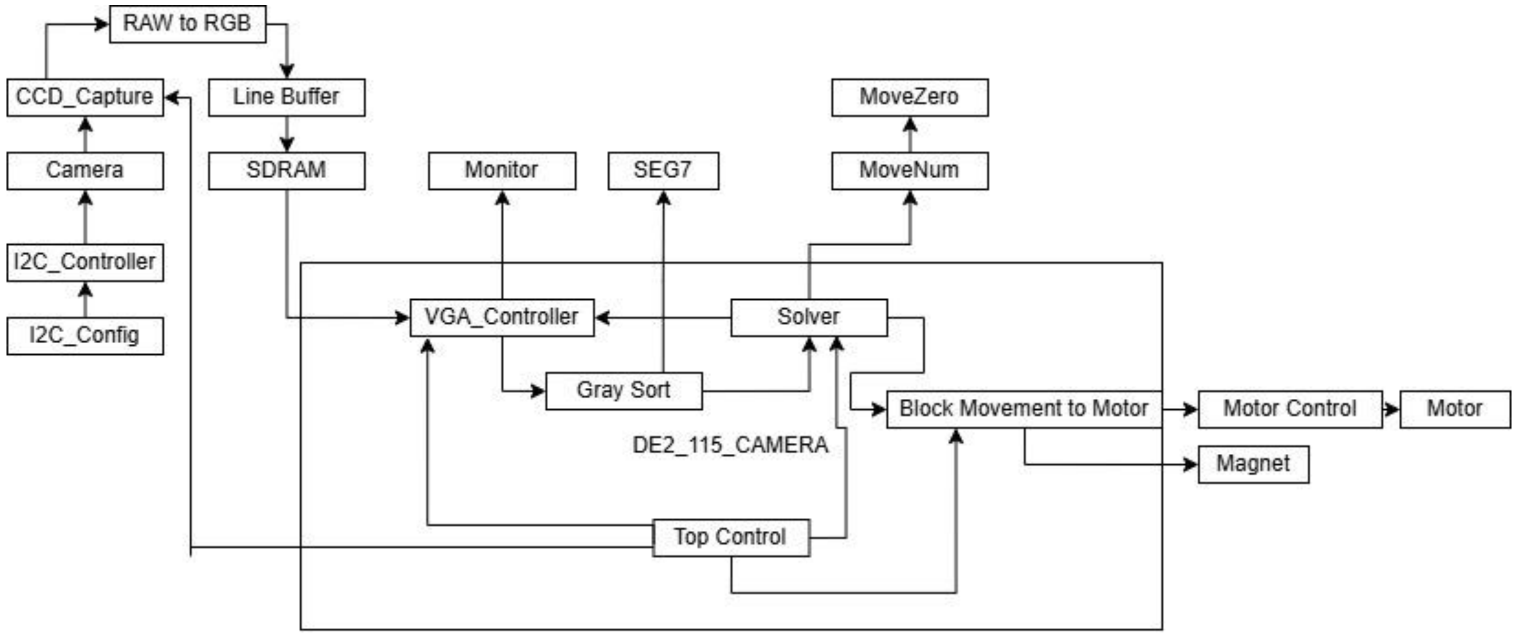
- From 1 to 15.
- Reuse “Move Number”.



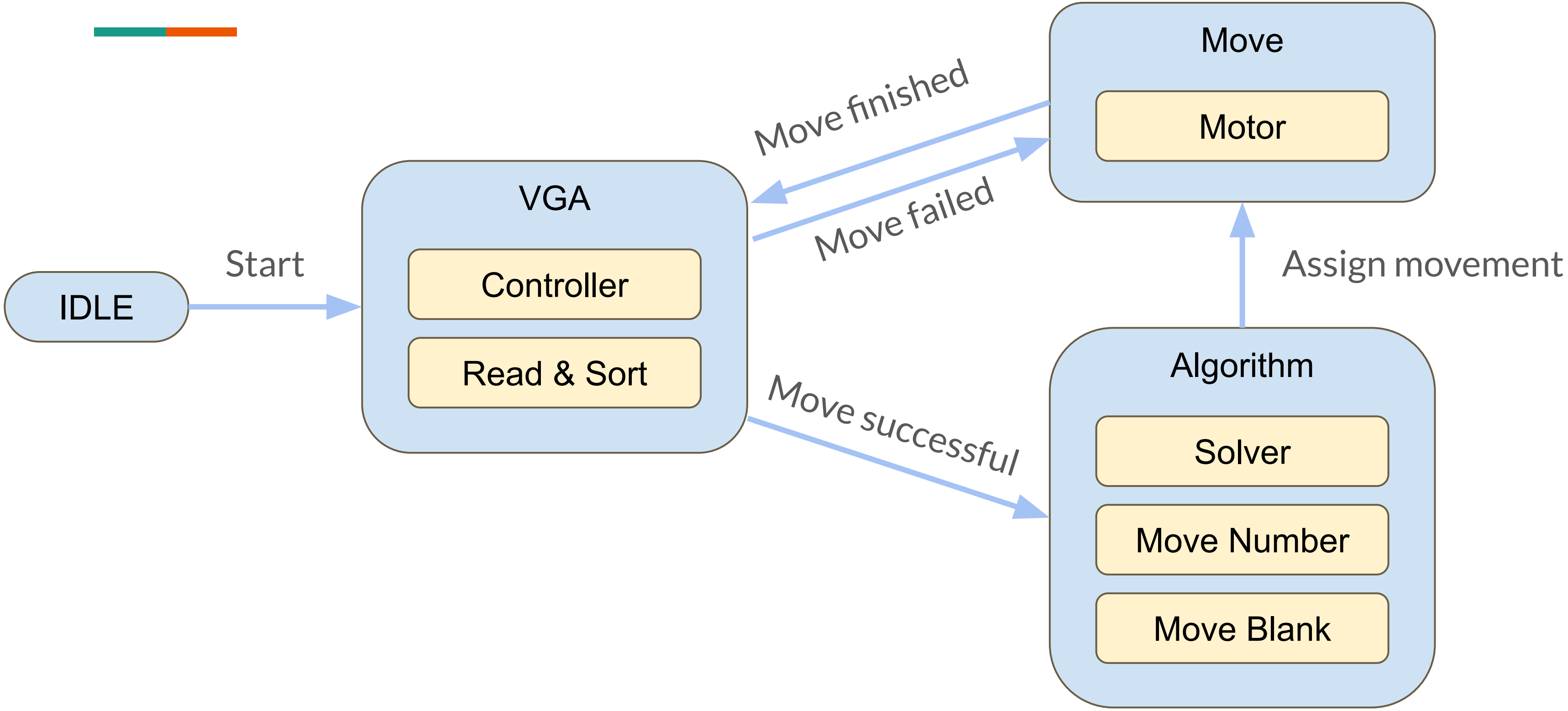
3	12	9	5
7	13	14	1
15	4	6	2
10	11	8	

Architecture

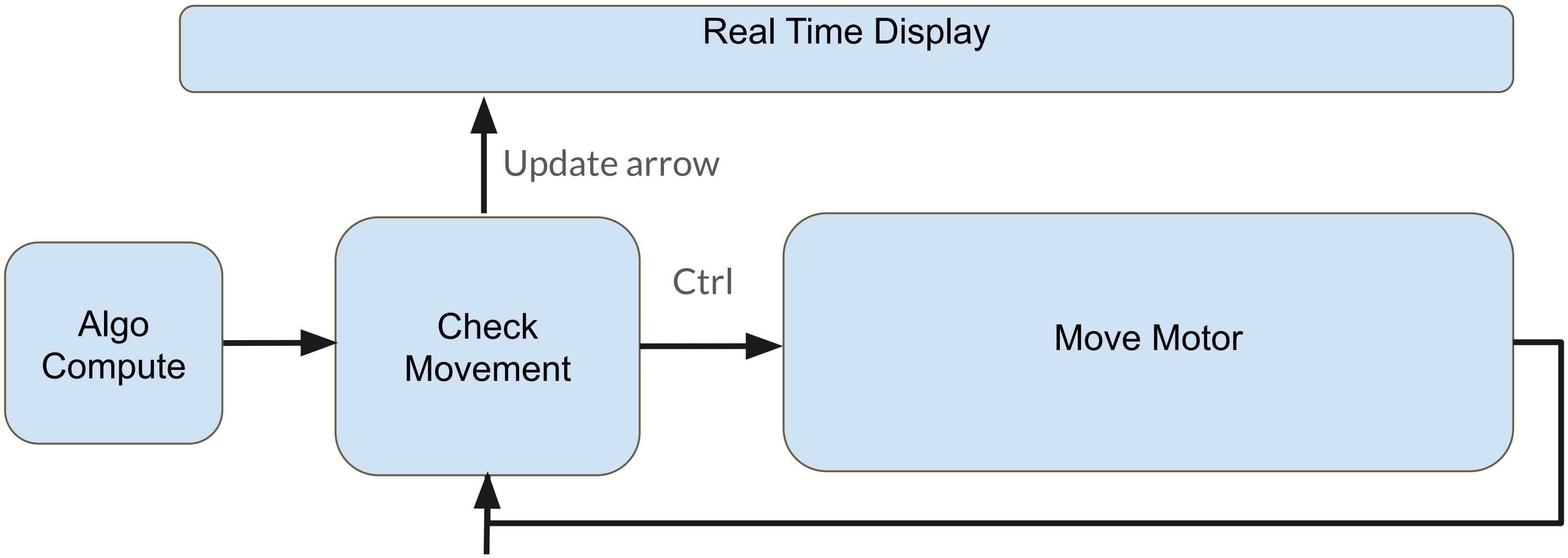
Architecture



Finite State Machine



Time Flow


Real Time Display
Update arrow
Algo Compute
Check Movement
Ctrl
Move Motor

Testing

Workflow

Python

Use python to simulate the algorithm.

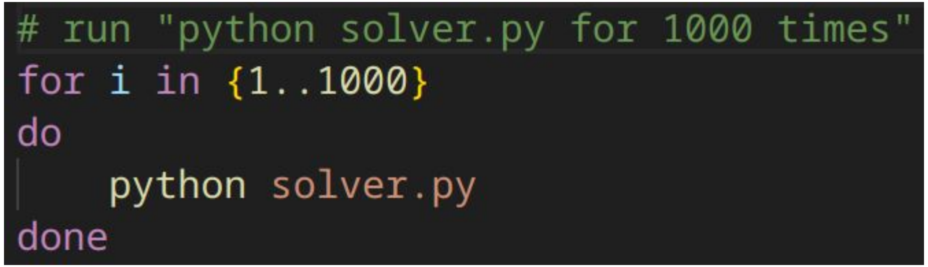
Submodule

Use several testbenches to check the function.

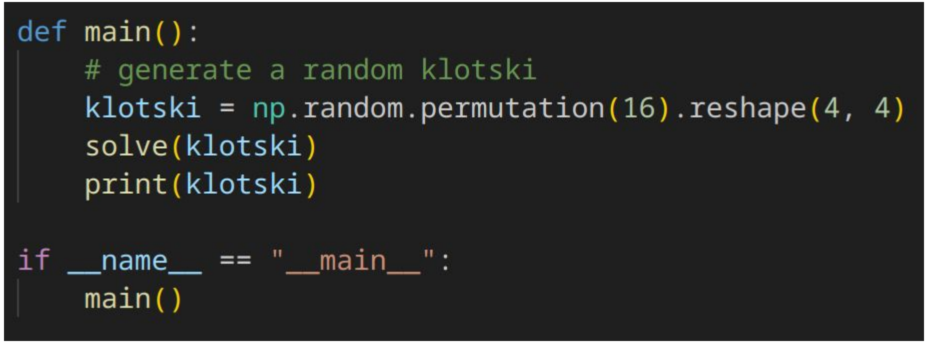
Top Module

FPGA

Python approach



```
# run "python solver.py for 1000 times"
for i in {1..1000}
do
    python solver.py
done
```

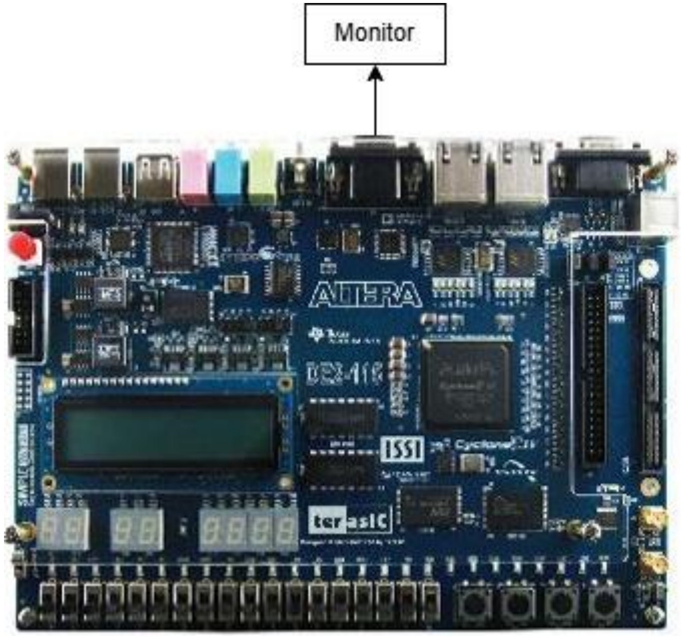


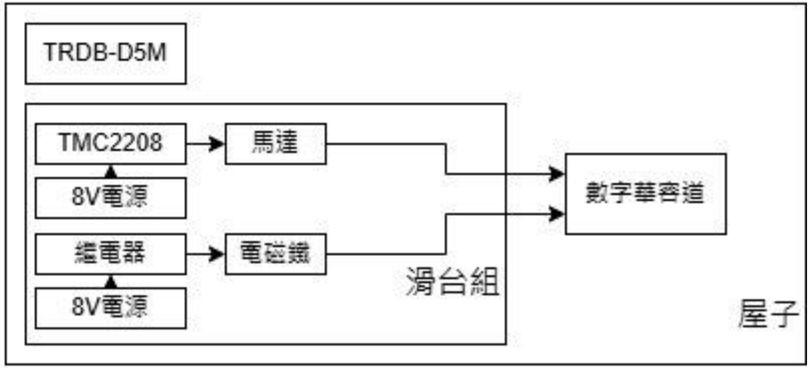
```
def main():
    # generate a random klotski
    klotski = np.random.permutation(16).reshape(4, 4)
    solve(klotski)
    print(klotski)

if __name__ == "__main__":
    main()
```

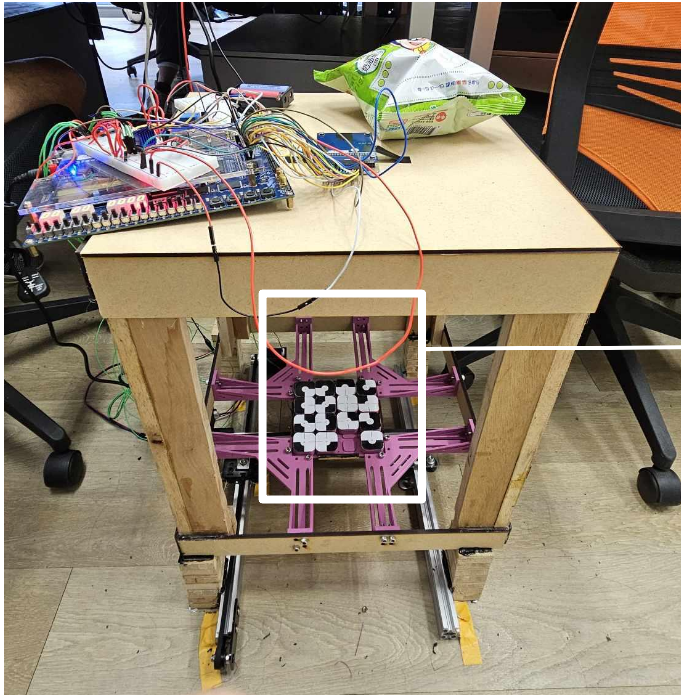
Implementation

Hardware Resources



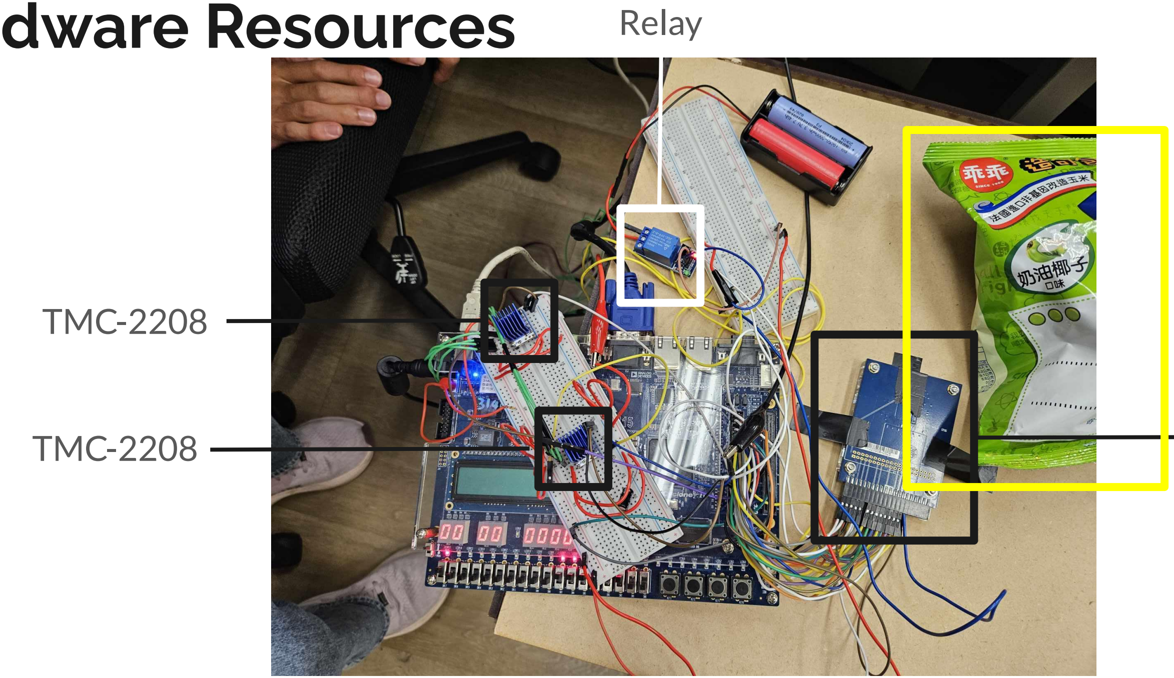


Hardware Resources



Klotski

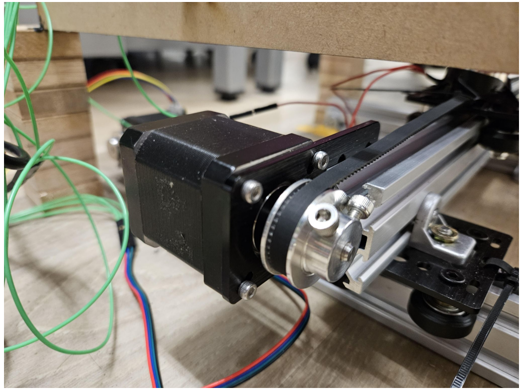
Hardware Resources

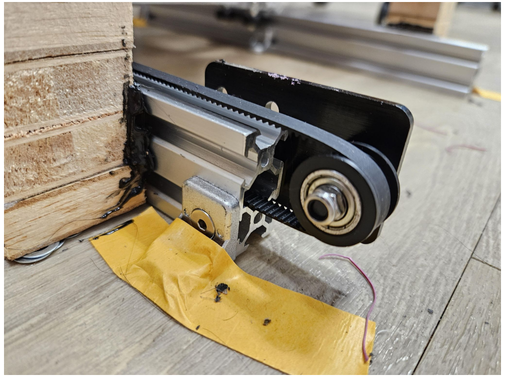


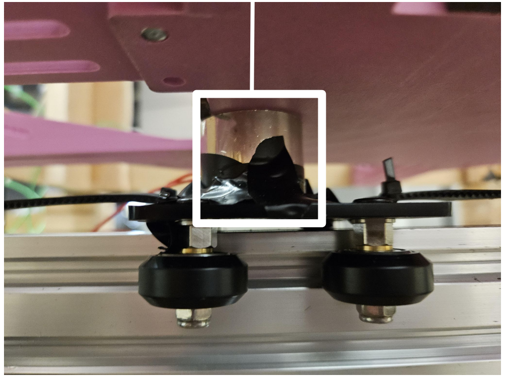
27 廿七

宜：祭祀、修造
忌：齋醮、冠笄

Hardware Resources







電磁鐵

Function of Motor

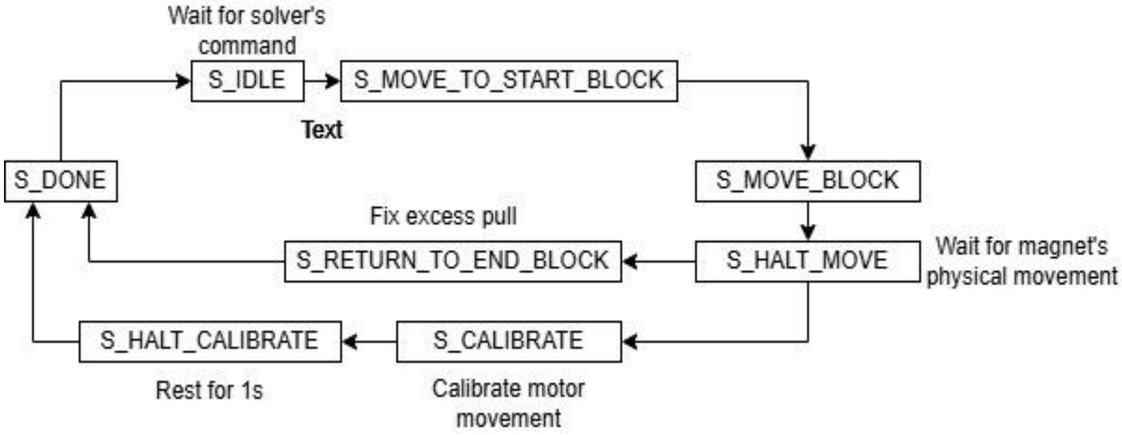
1. Receive start block & end block from solver

2. Route Planning(avoid intervention between different movement)

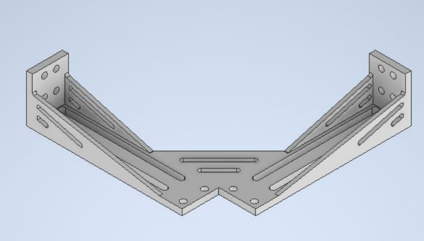
3. Signal conversion

4. Excess tow & Rest

5. Automatic movement calibration

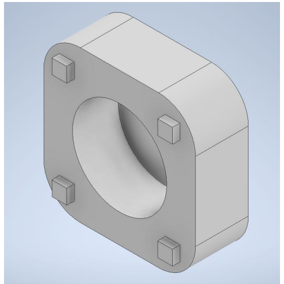


klotski



Issue 1: Rotation of cube when moving

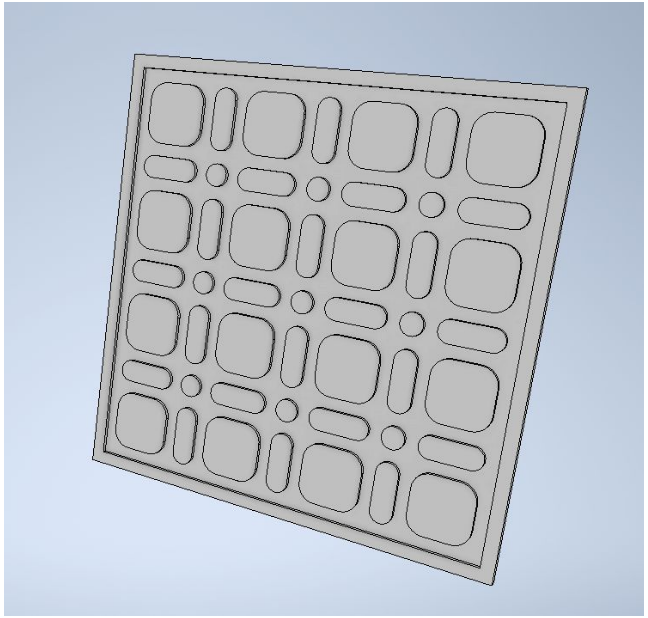
Solve 1: Moving along the rail



Issue 2: Cubes stuck with each other

Solve 2: Corner rounded,

Sense whether a movement is correctly executed



Block Sensing

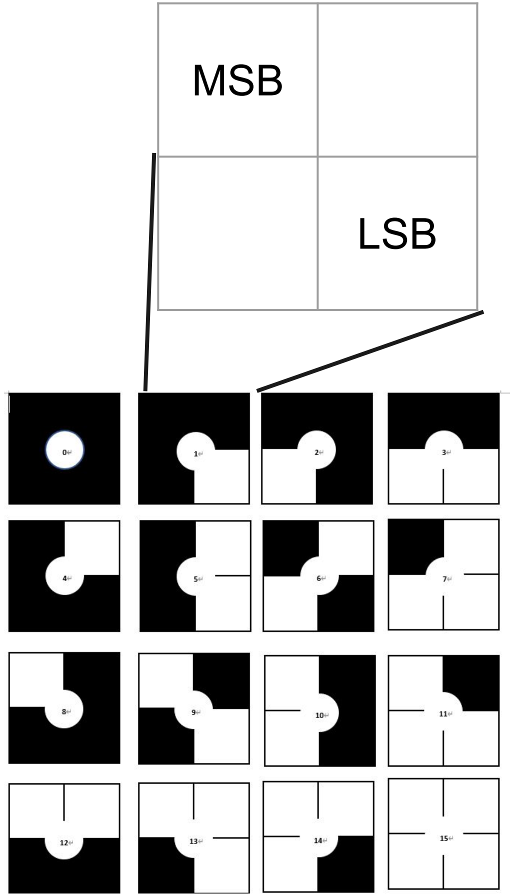
- Issue:

  Relative color is sensitive to relative brightness

- Use black and white to encode

  4 sample points in a session to increase accuracy

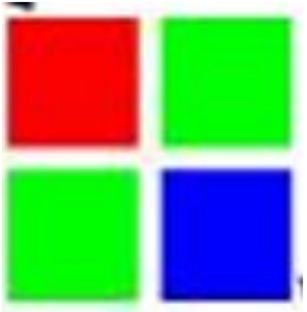
- Shift before Addition to reduce required register

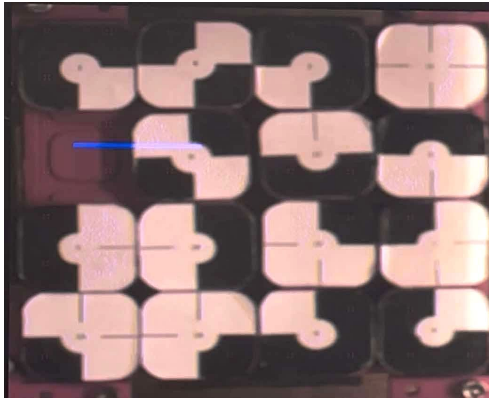


Block Sensing

- Use black and white to decode
- Bayer pattern -> RGB

  Read **Green signal** to represent the block
- Use **blue arrow**, and thus no interference





Novelty

- We proposed the algorithm for klotski.
- To handle failed moves, we use VGA to check if the position of the number is correct after every move.

參考資料

- DE2-115 User Manual：https://www.terasic.com.tw/attachment/archive/502/DE2_115_User_manual.pdf
- TRDB-D5M User Manual ：Https://www.terasic.com.tw/attachment/archive/281/TRDB_D5M_UserGuide.pdf
- A4988驱动NEMA步进电机(42步进电机)http://www.taichi-maker.com/homepage/reference-index/motor-reference-index/arduino-a4988-nema-stepper-motor/
- Arduino控制電磁鐵繼電器正確使用｜智慧工廠專題 https://www.youtube.com/watch?v=A4OWrq6dQNA
- Real-Time-Image-Video-Dehazing https://github.com/DCLAB08/Real-Time-Image-Video-Dehazing/tree/main
- alhaitham-hardware https://github.com/aabdelfattah/alhaitham-hardware
- [Arduino 範例] 繼電器(Relay)的使用 https://blog.jmaker.com.tw/arduino-relay/