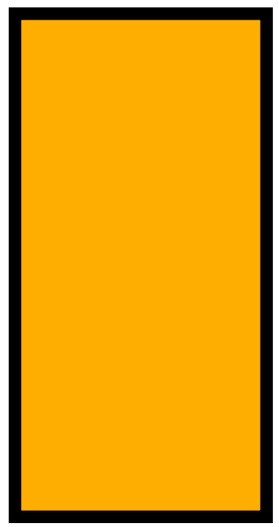
Platformer game

Technical storyboards

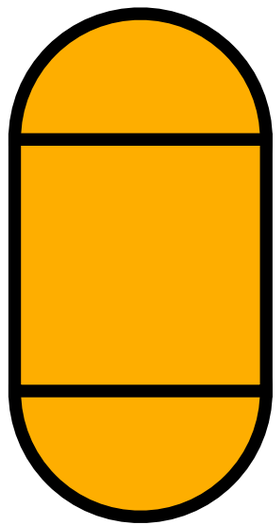
Malcolm Ryan 2024

Avatar

Collider choice



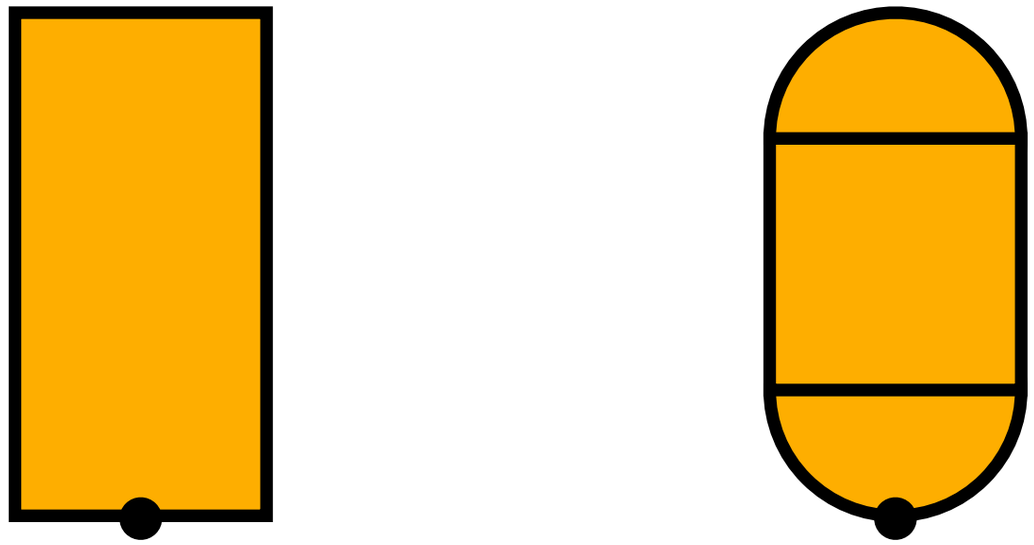
Box



Capsule

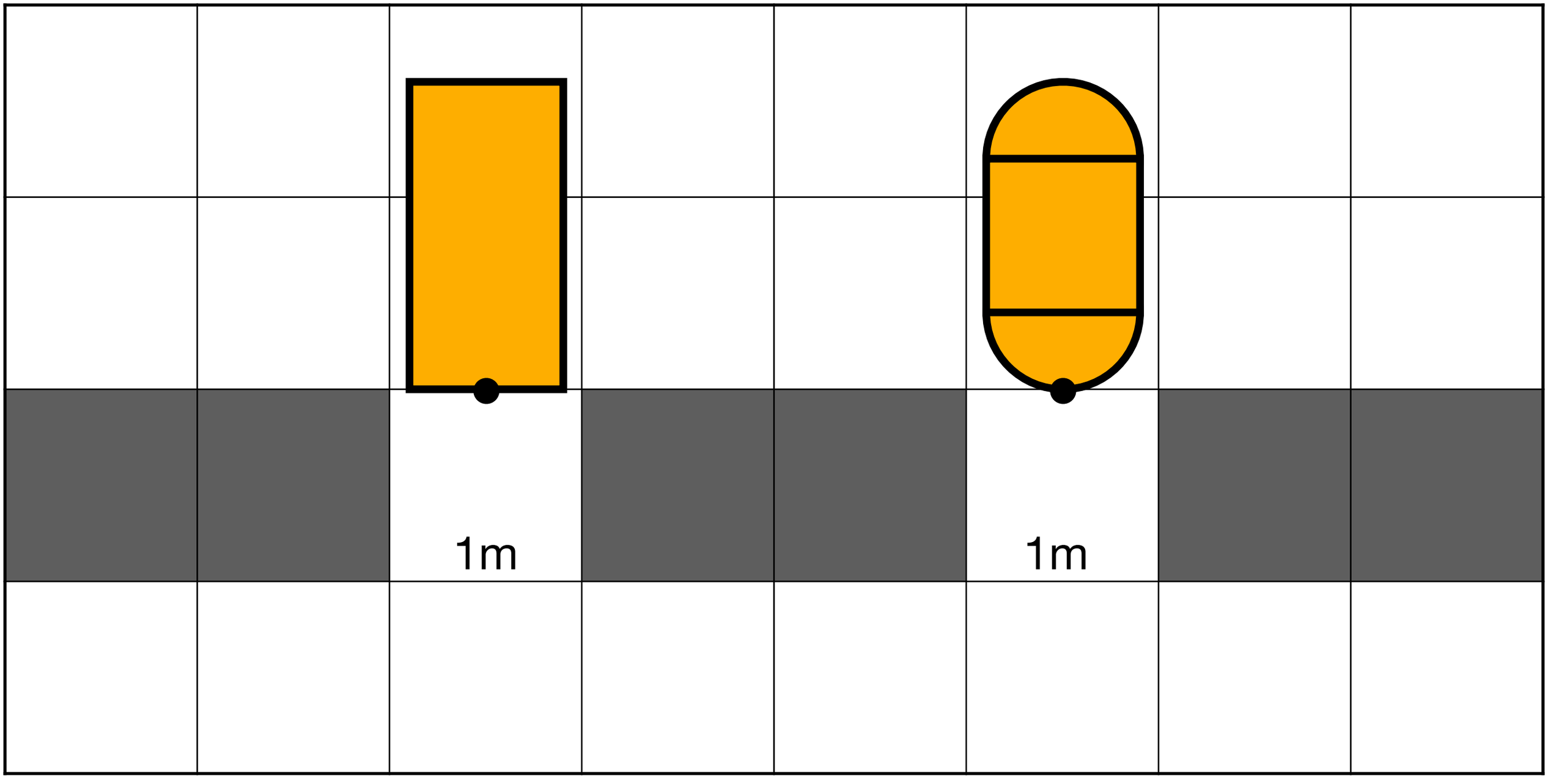
Avatar

Pivot point



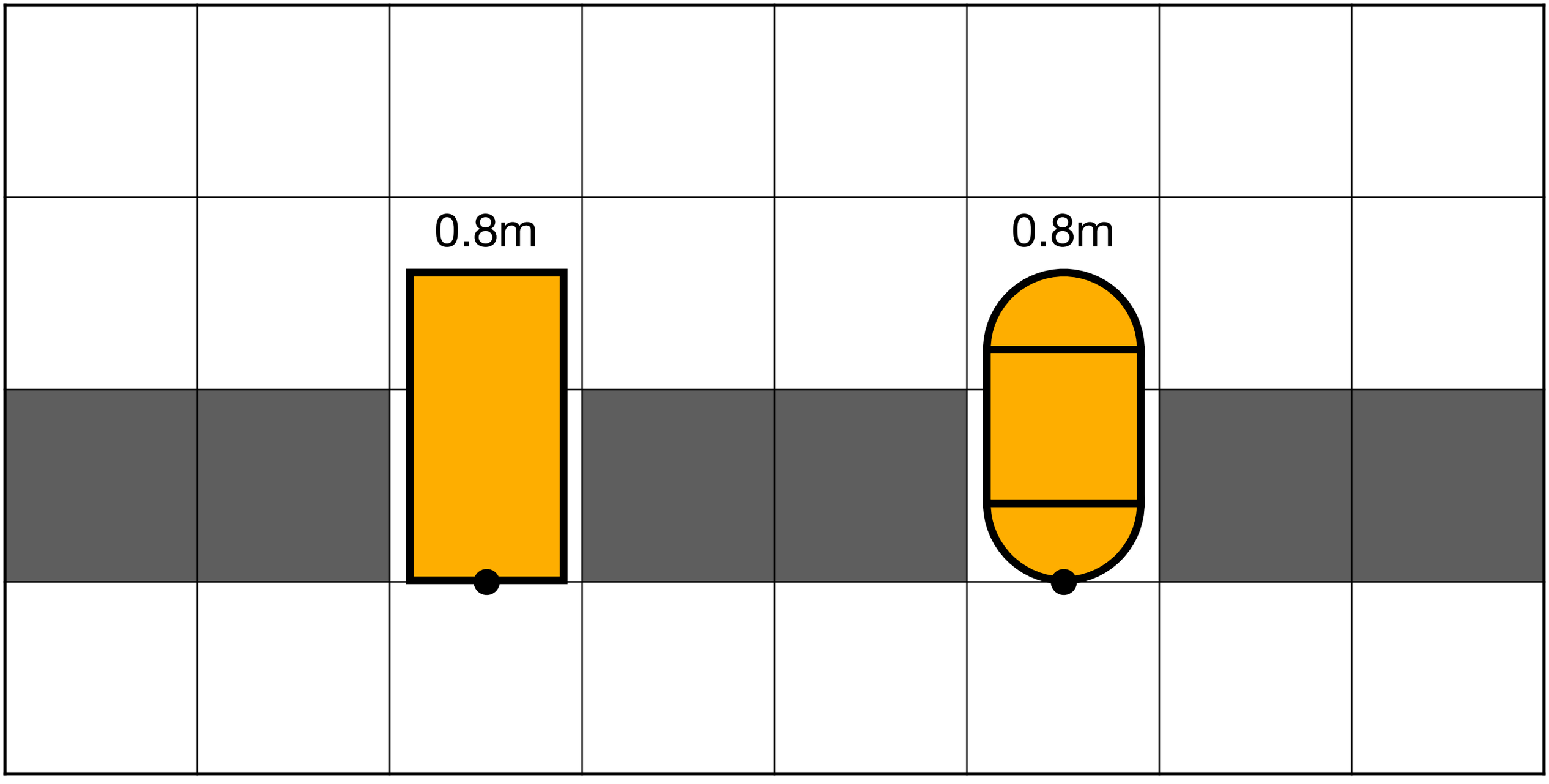
Avatar

Size



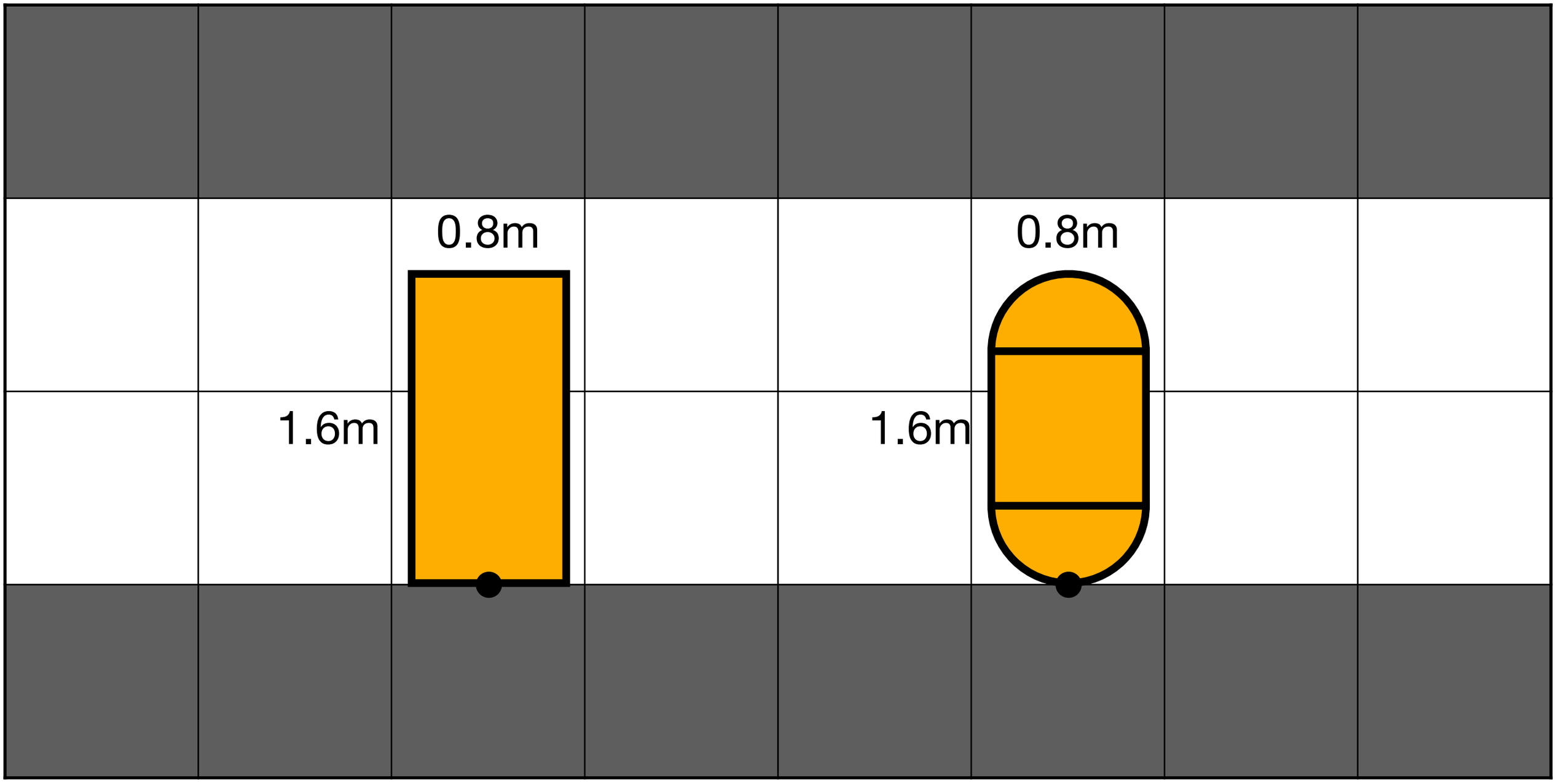
Avatar

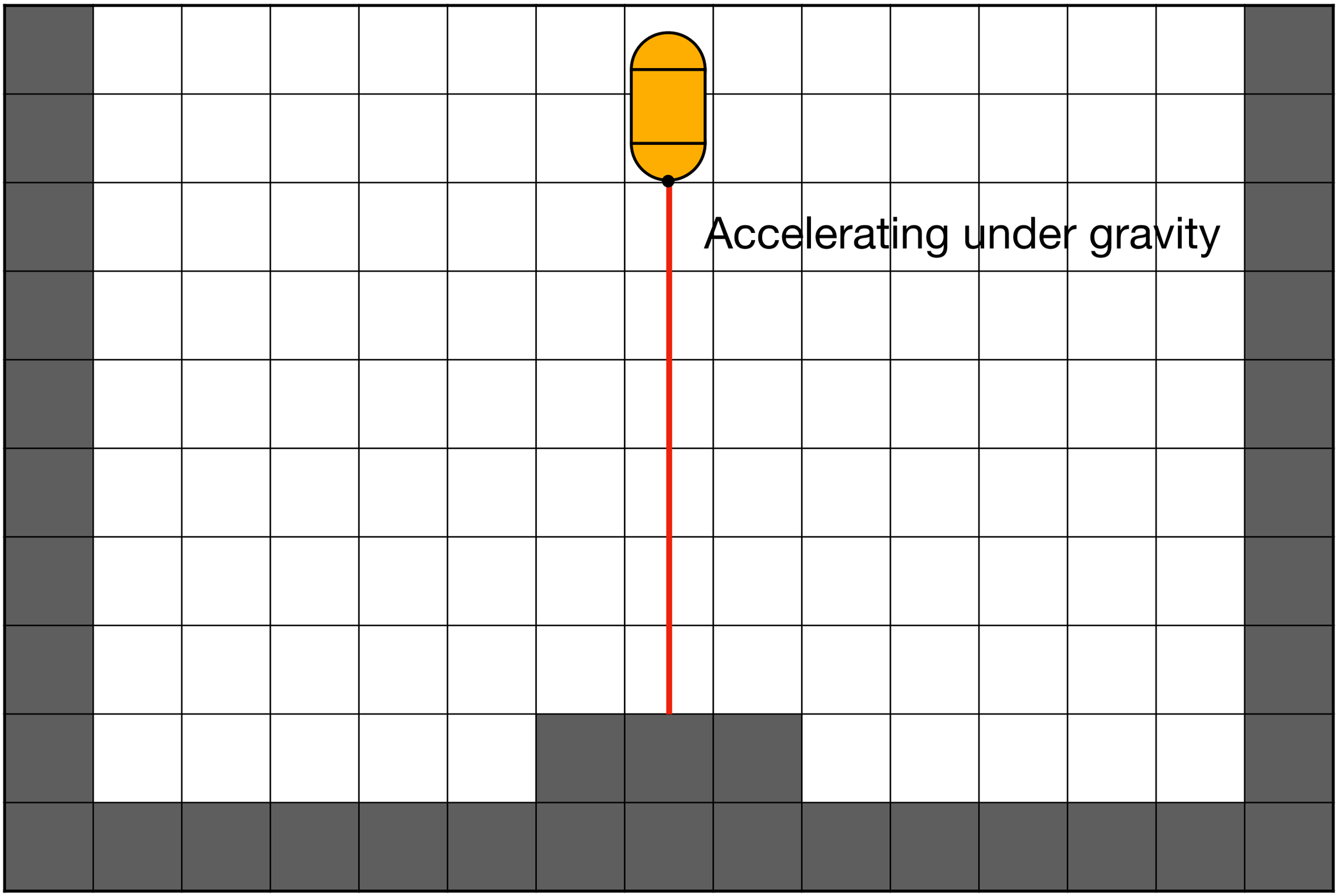
Size

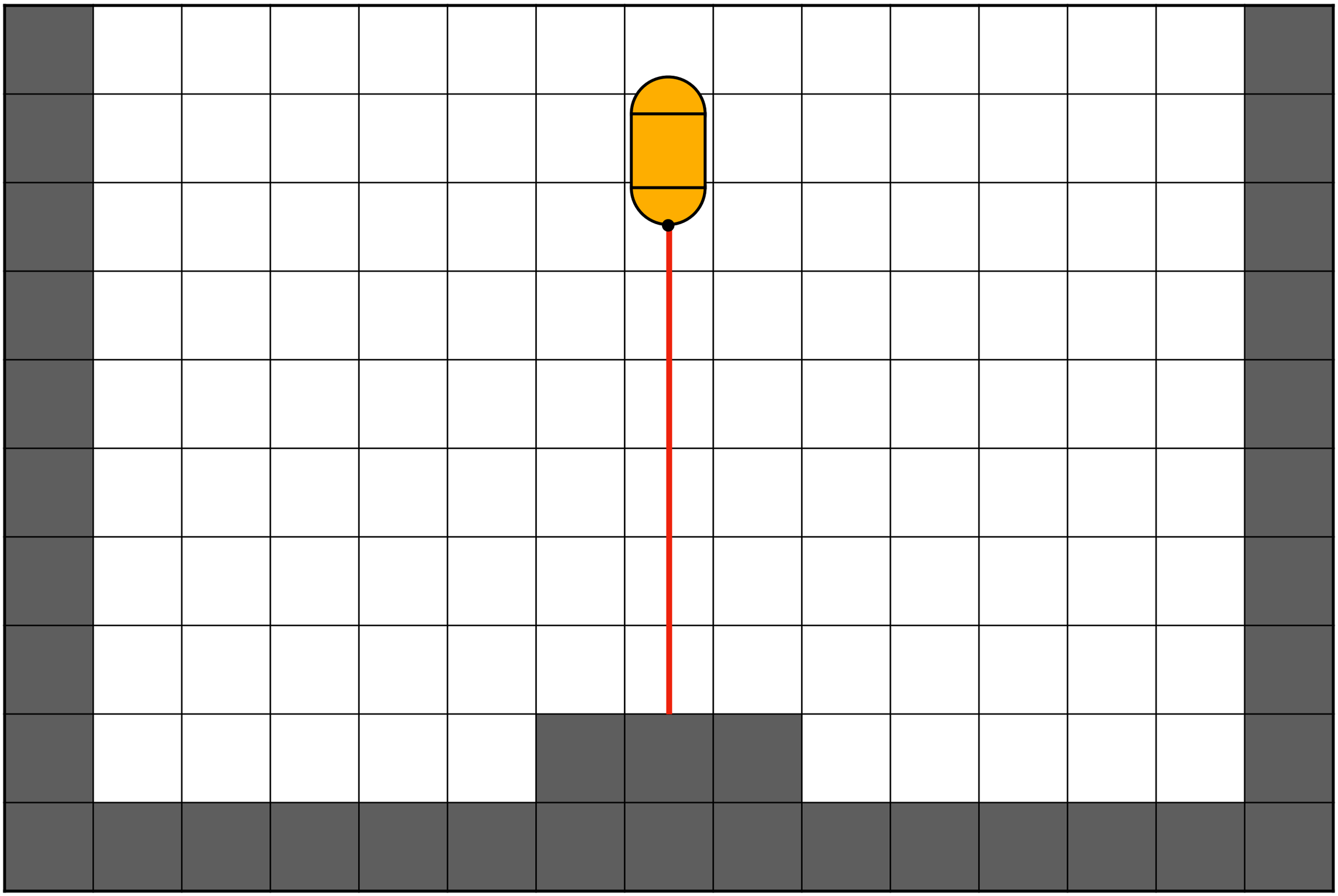


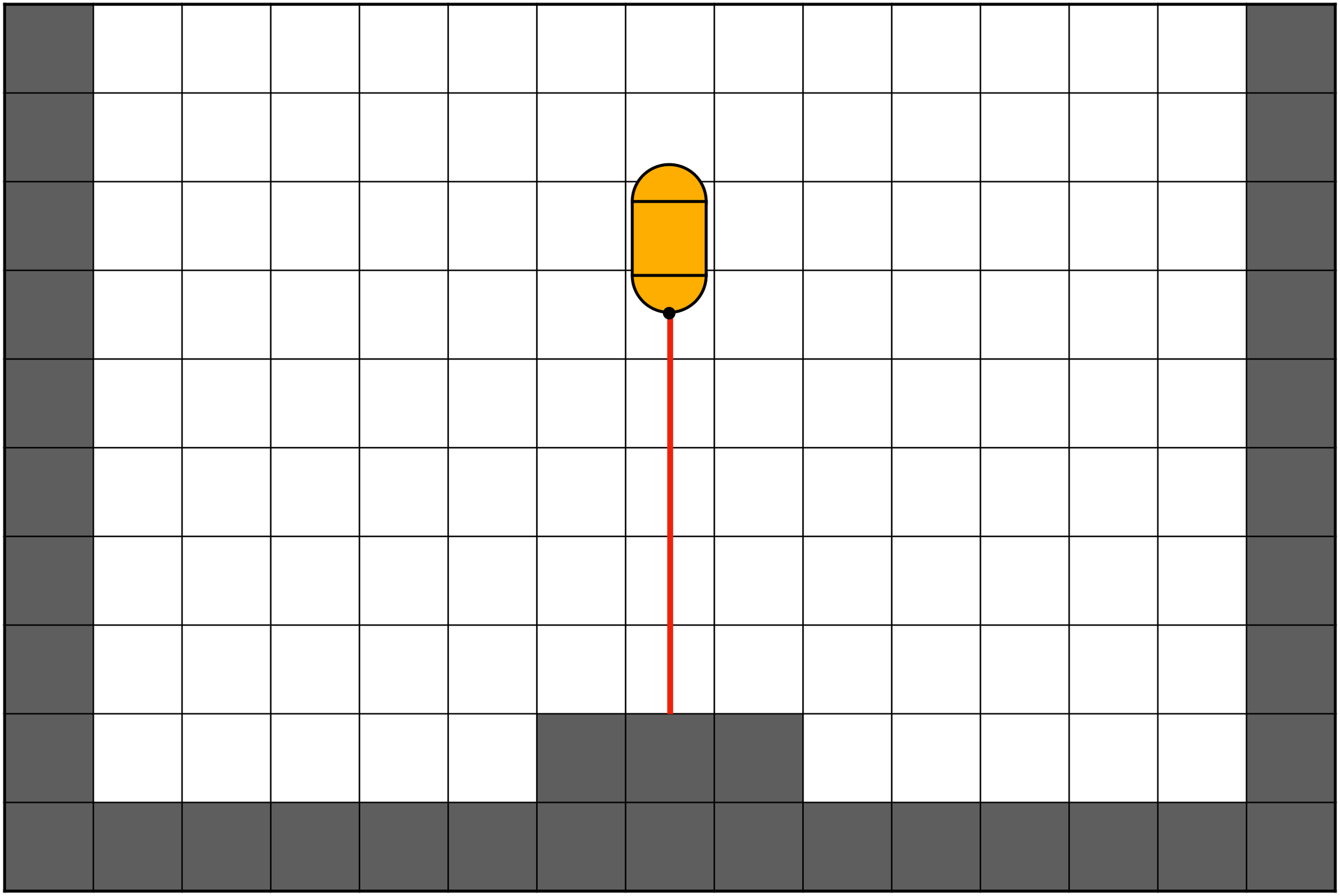
Avatar

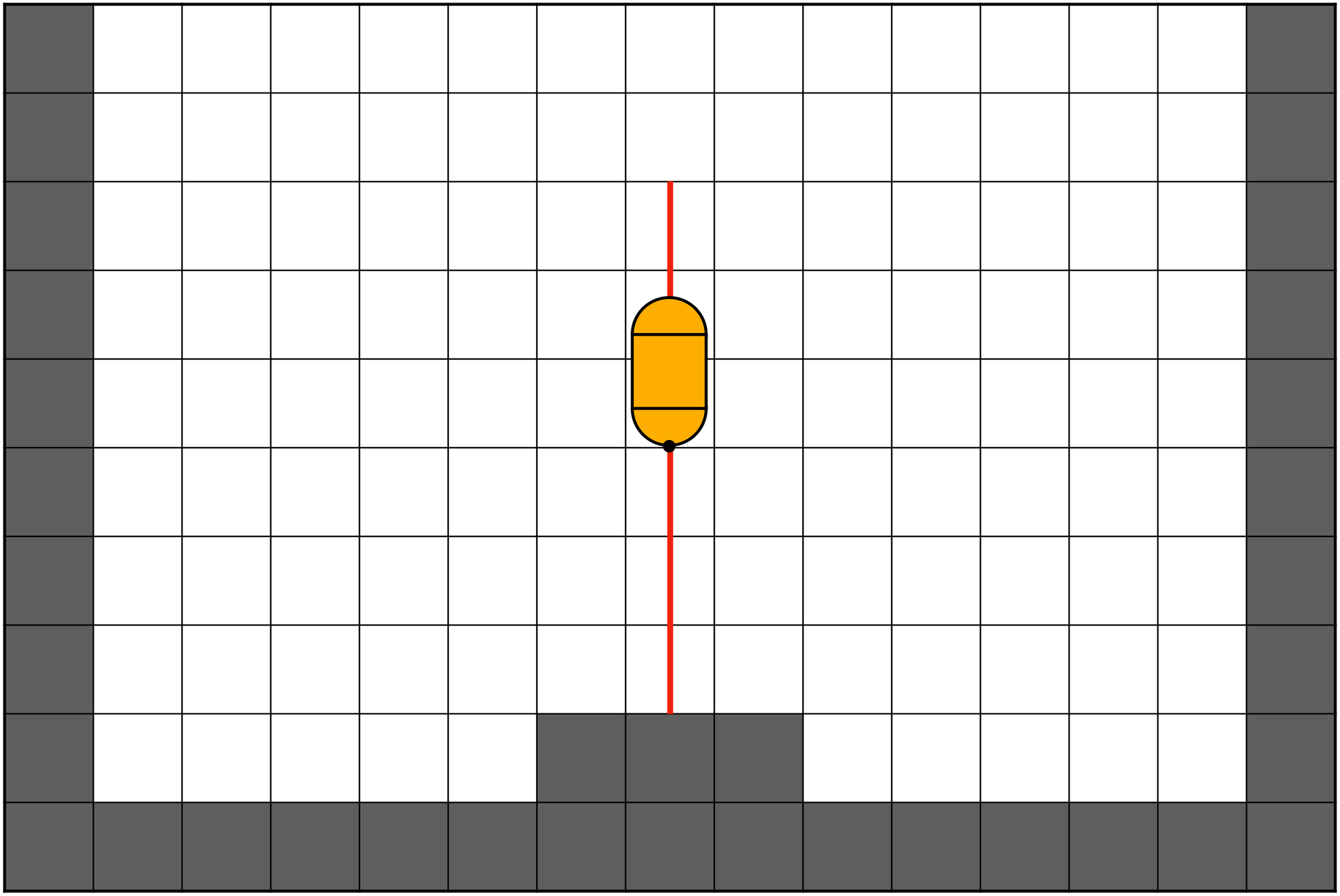
Size

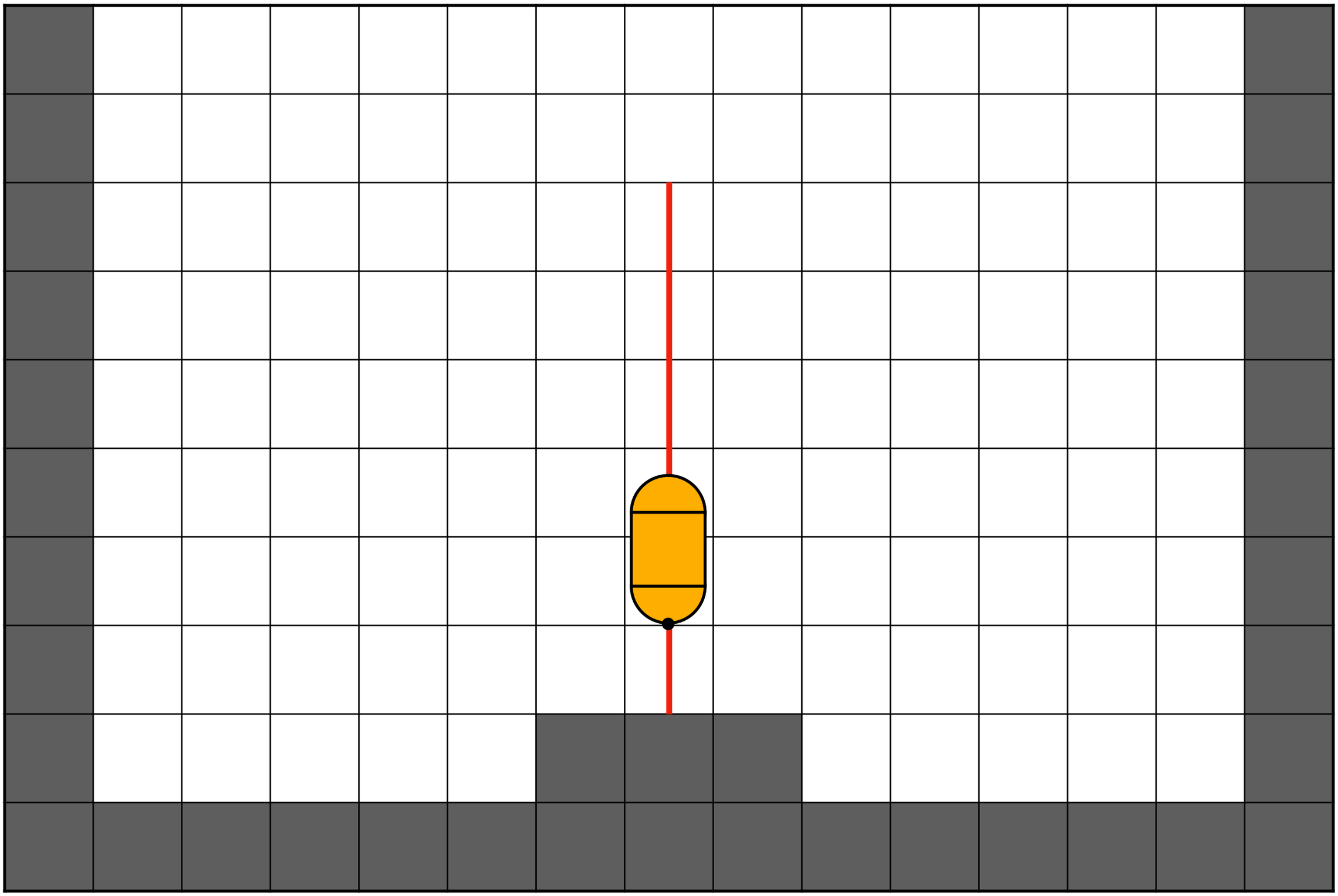


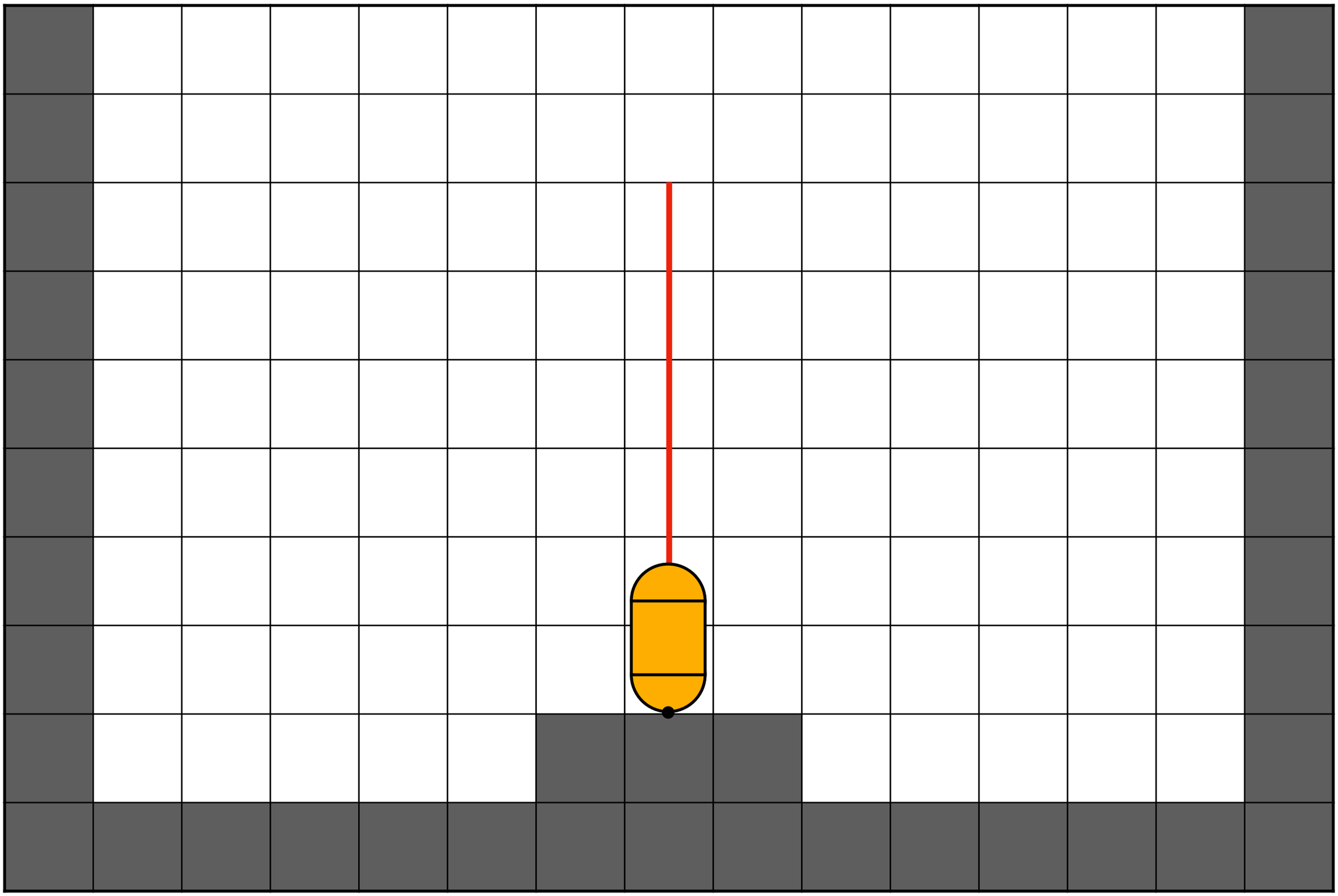

Accelerating under gravity

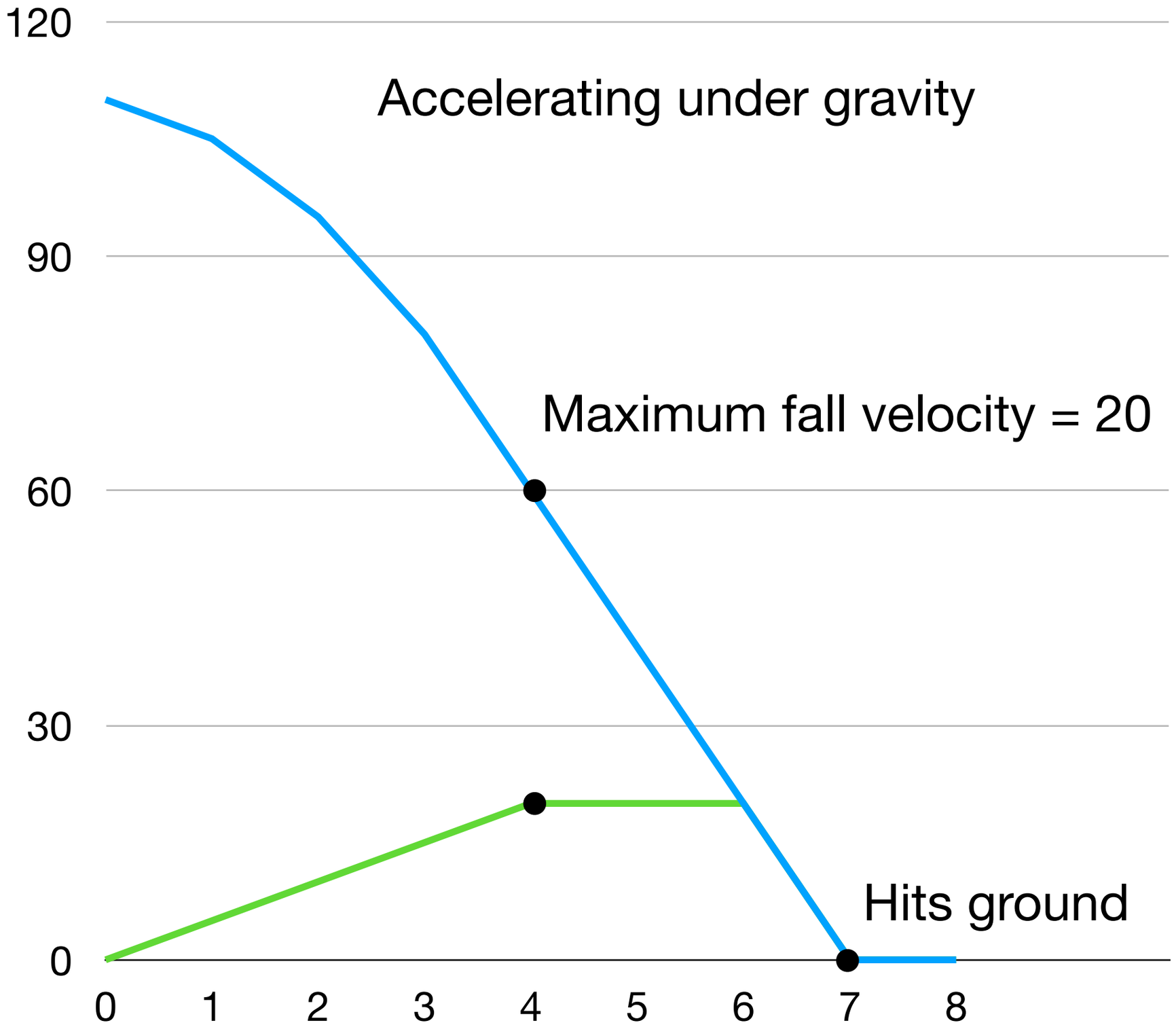




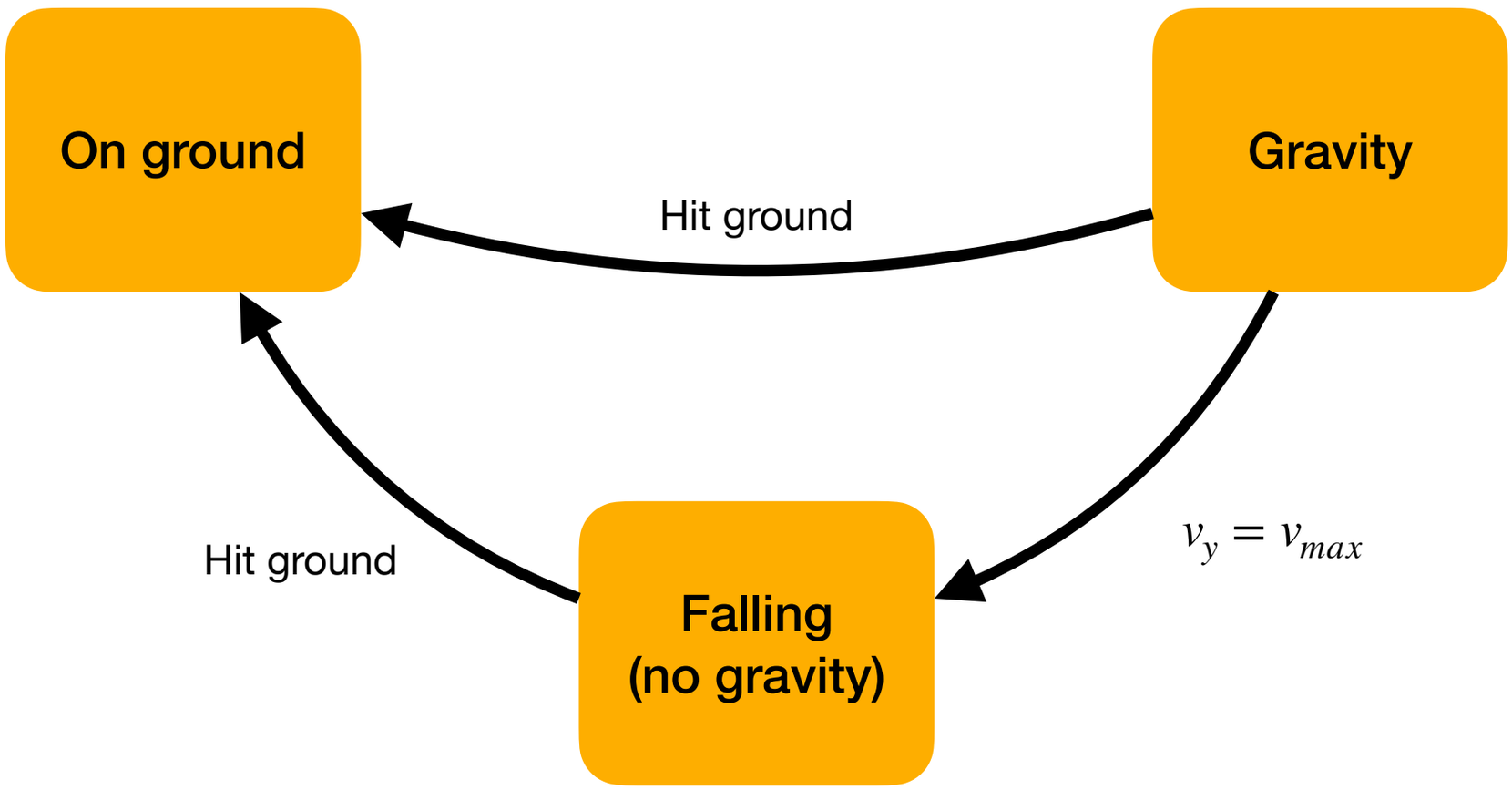


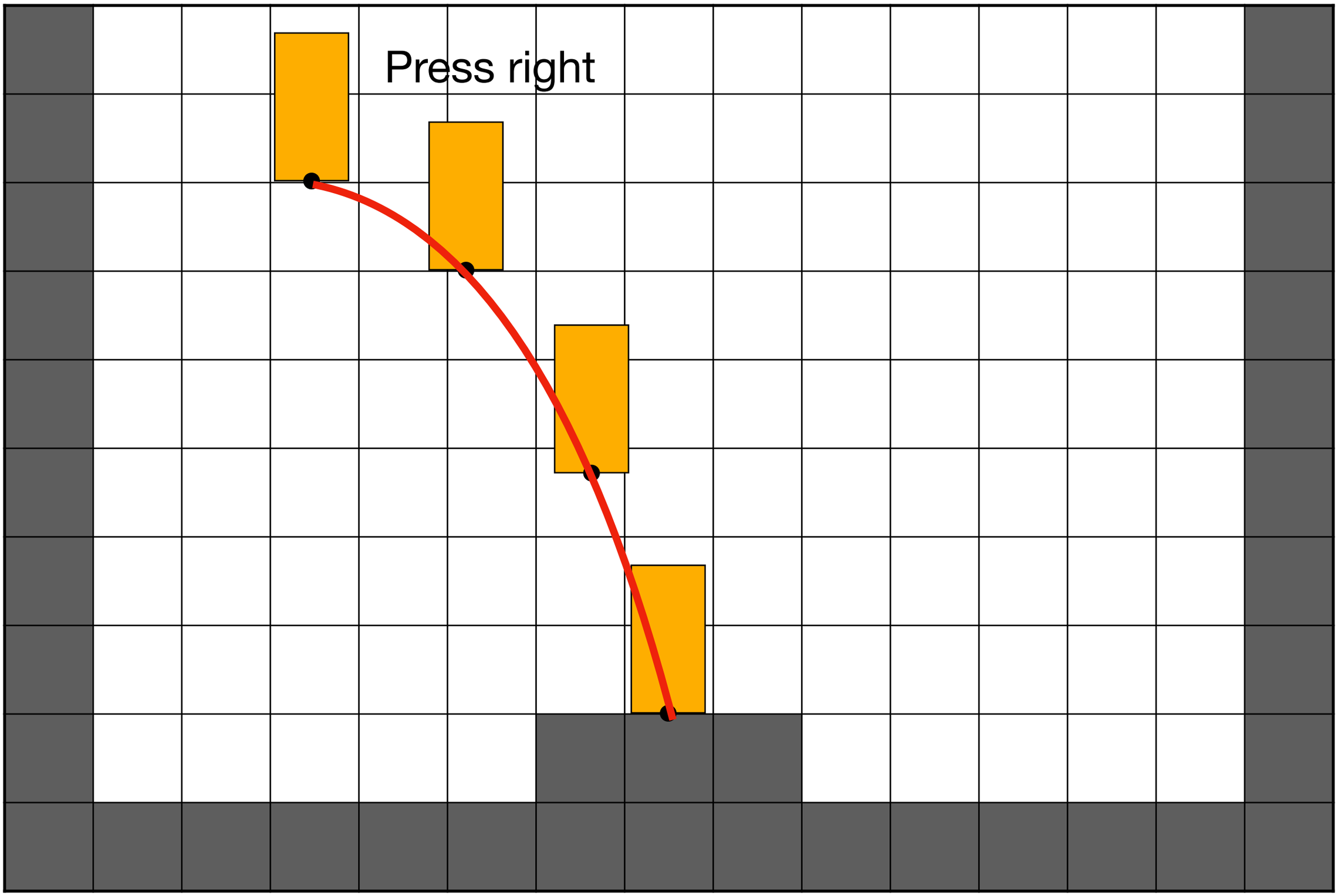


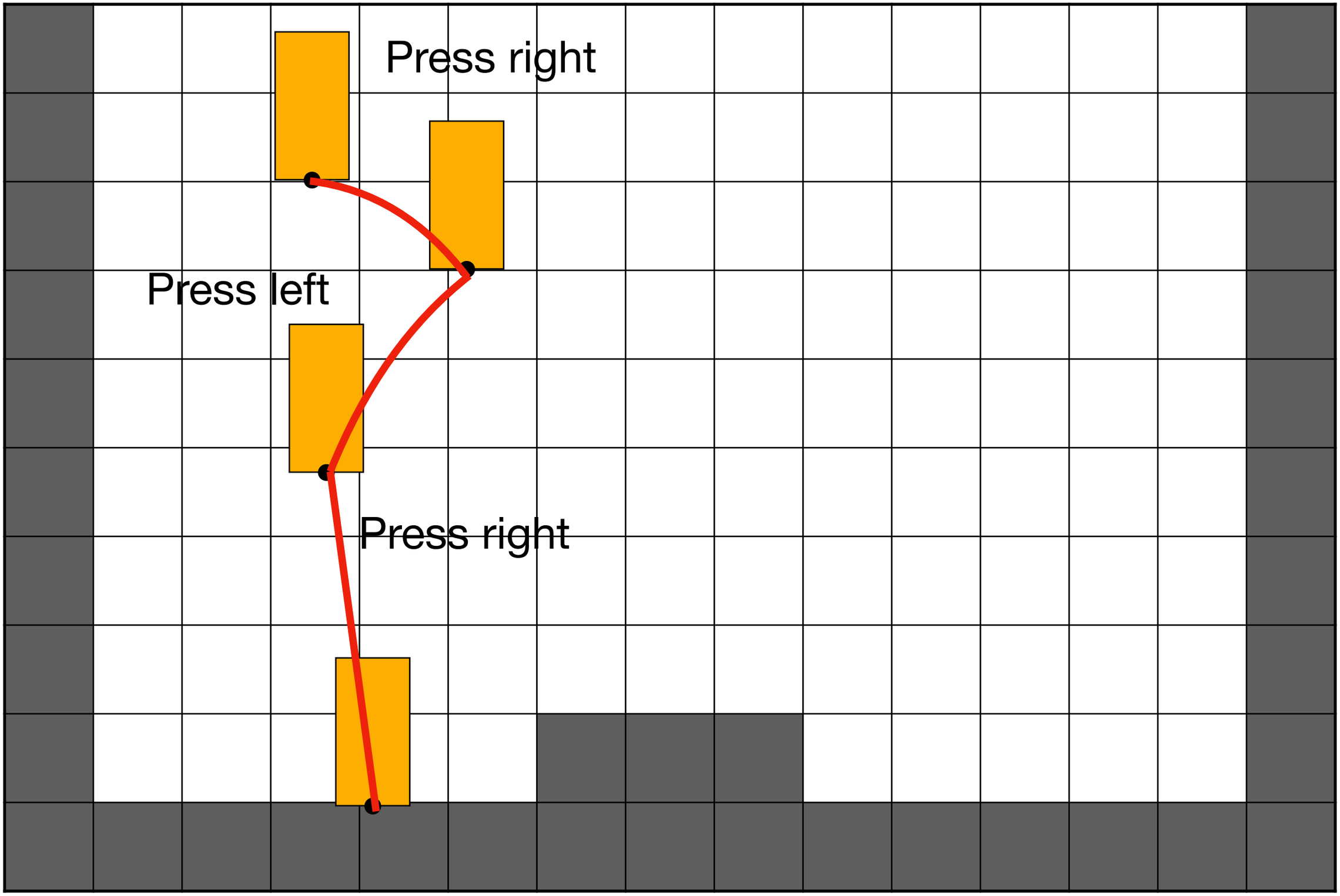



120
90
60
30
0
Accelerating under gravity
Maximum fall velocity = 20
Hits ground
0
1
2
3
4
5
6
7
8

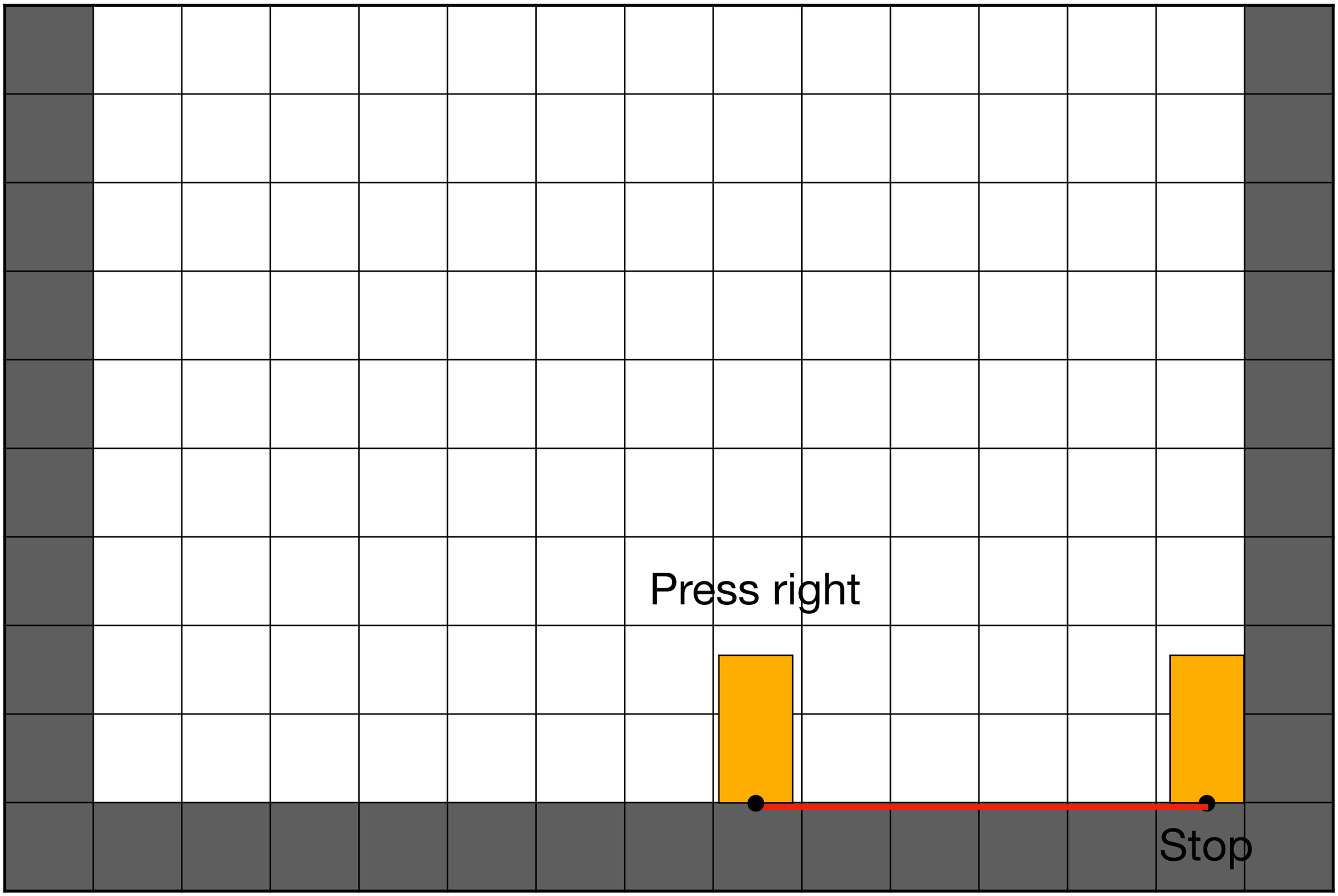
State machine




Press right

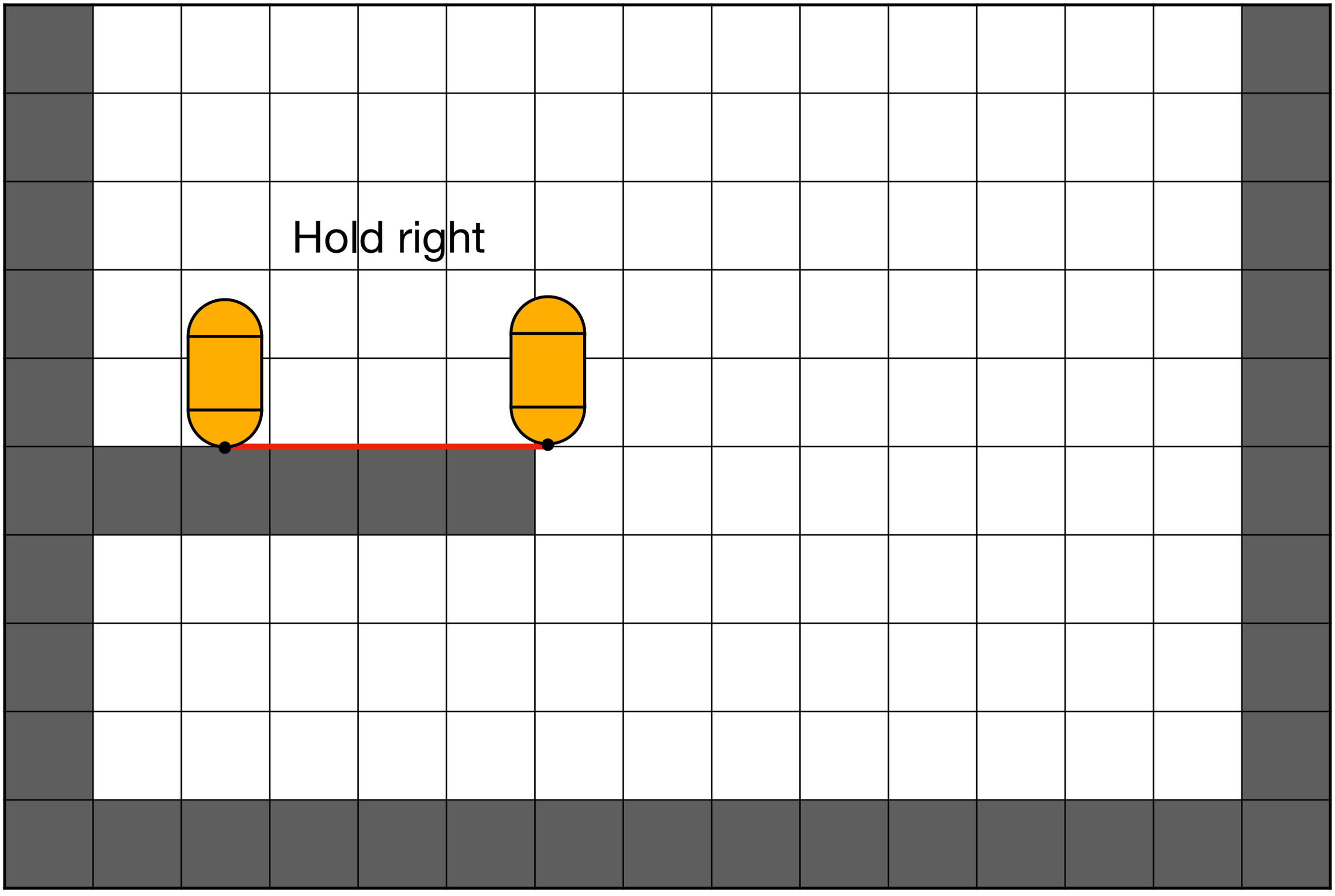

Press right
Press left
Press right

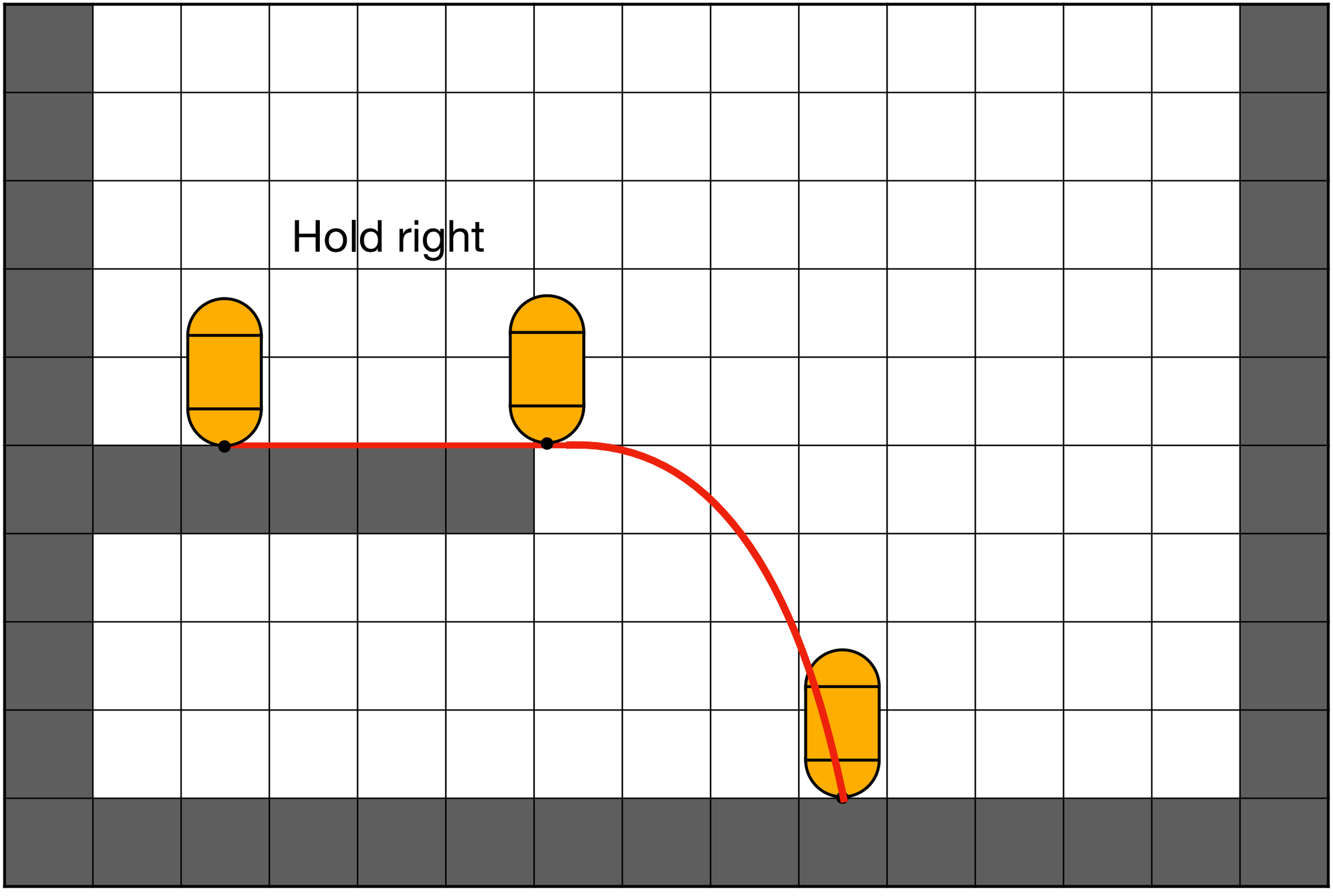
(or almost)

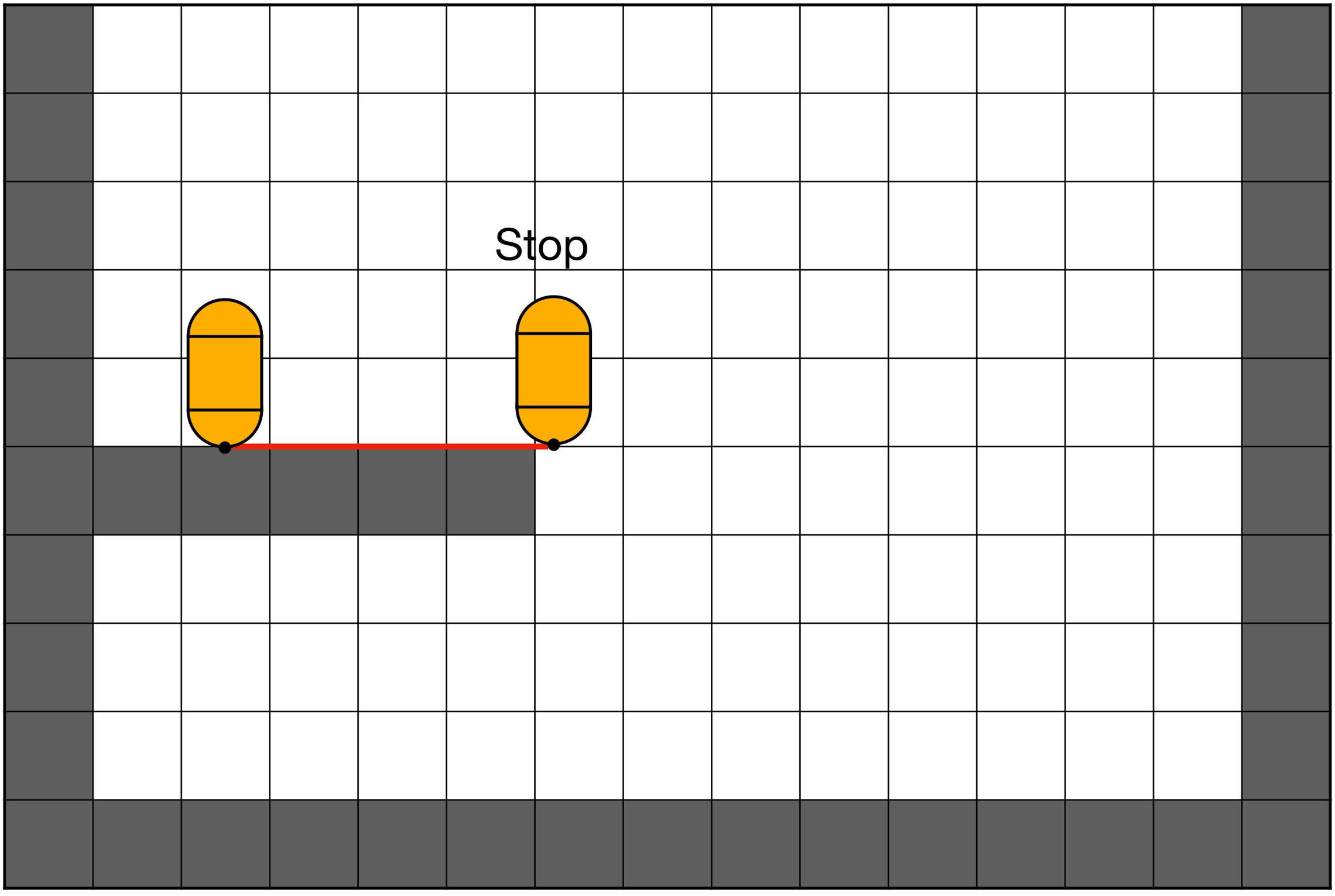

Press right
Stop

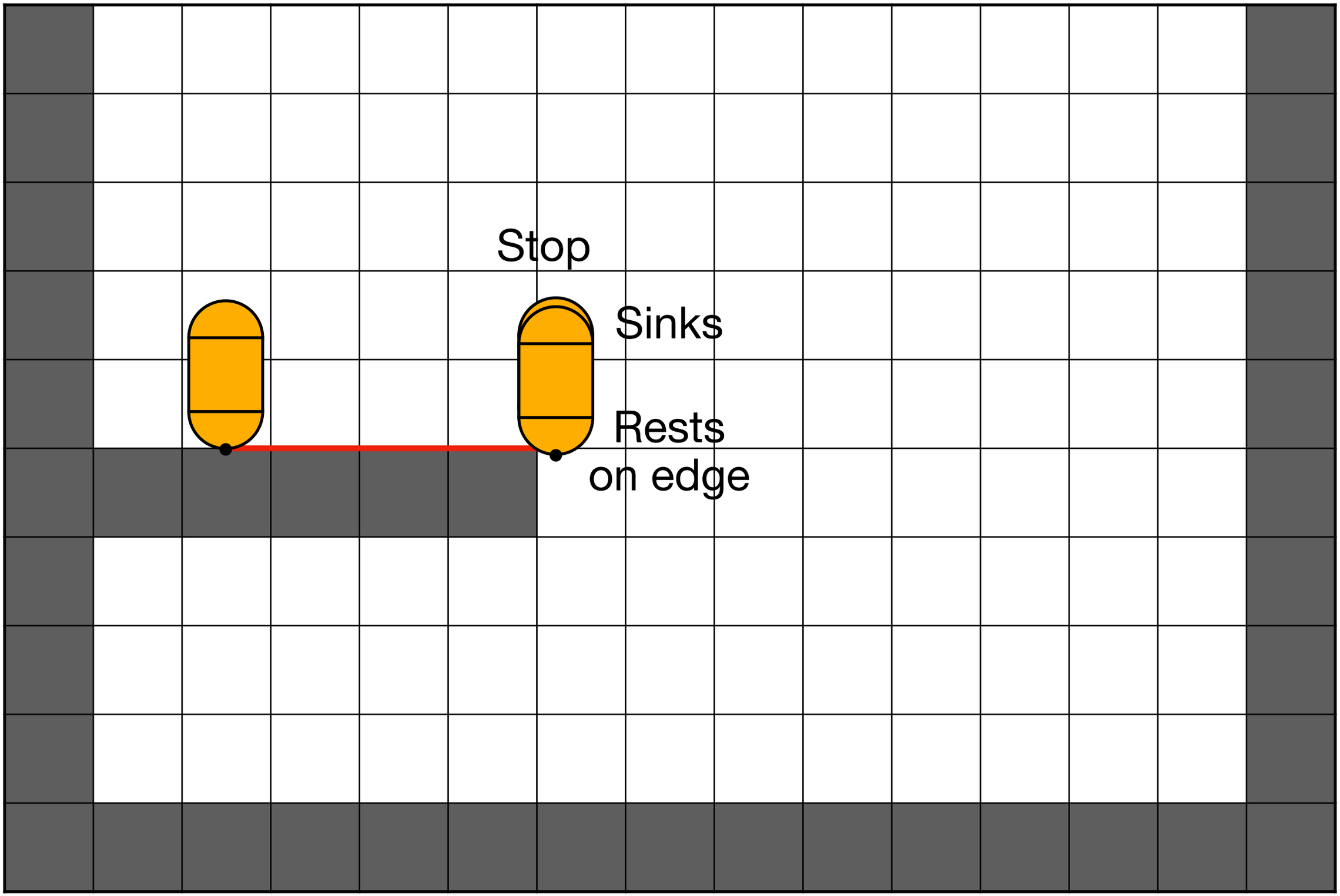
General rules

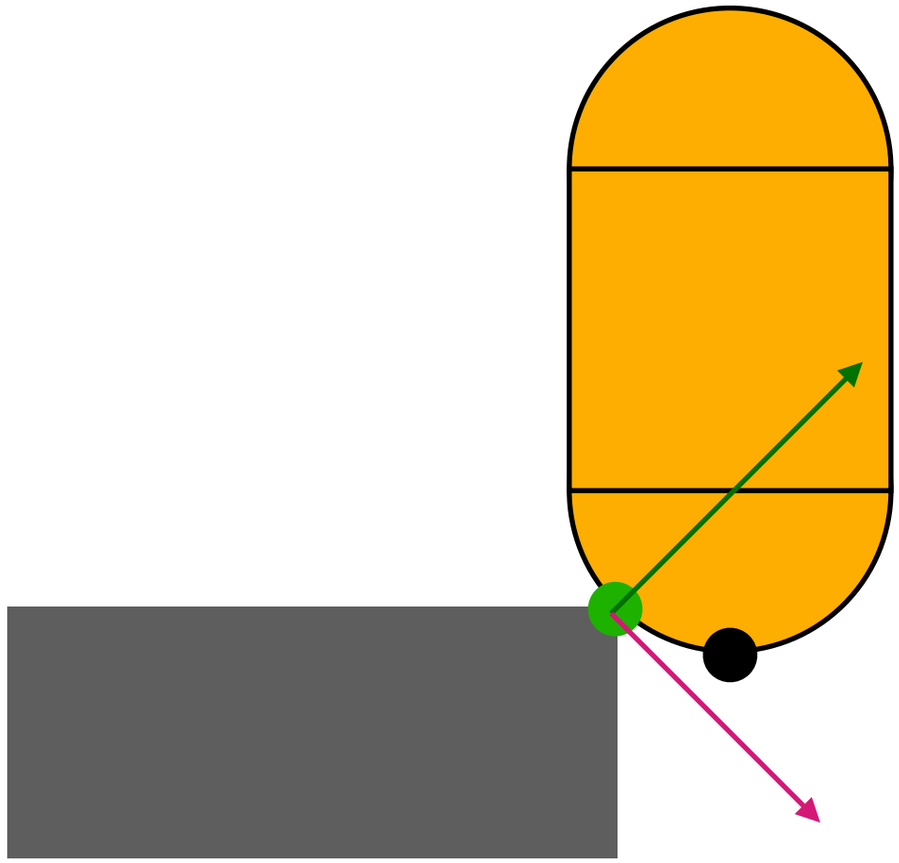
- Changes in horizontal velocity should be (almost) instant
- Horizontal velocity should match input (unless stopped by a wall)
-

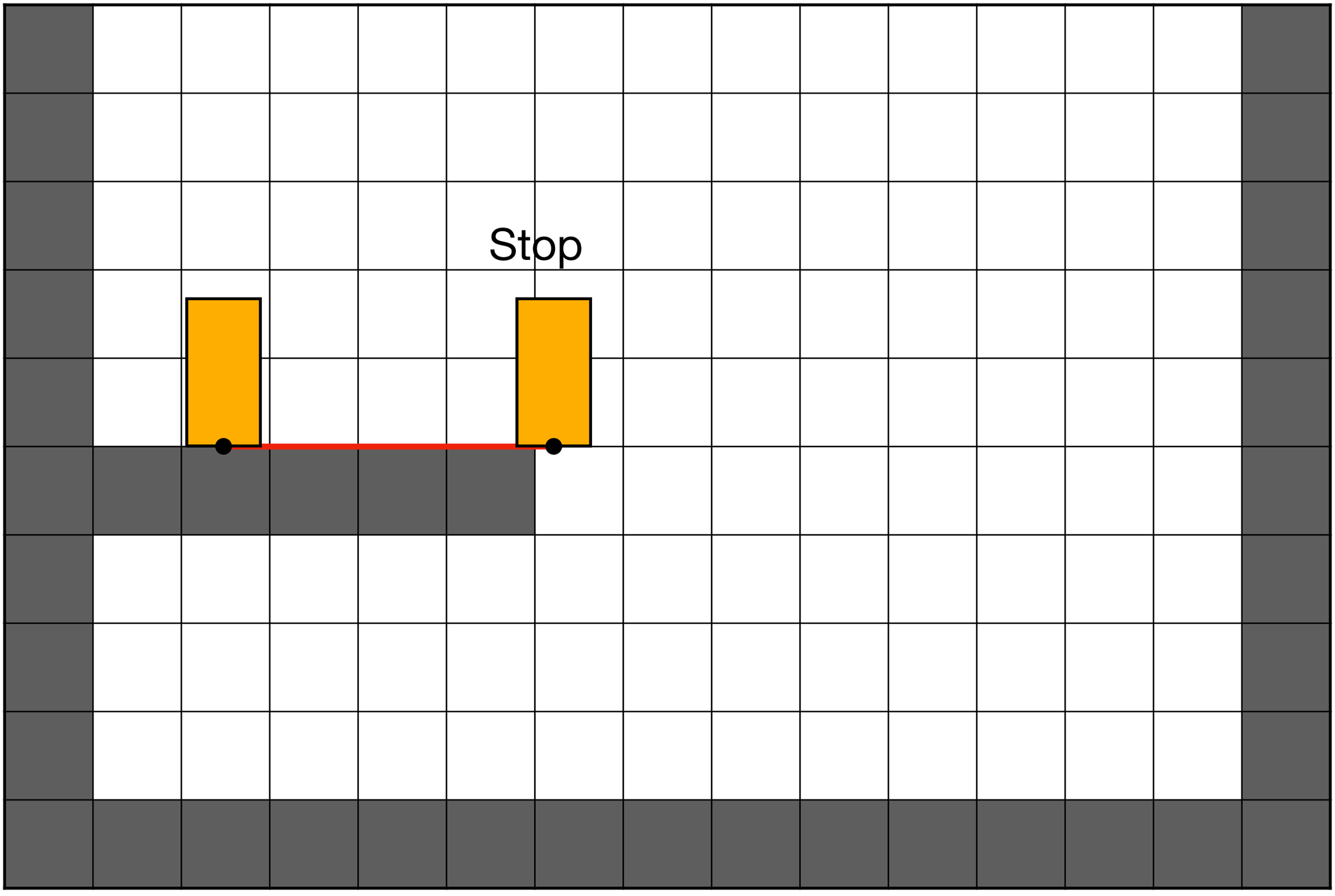

Hold right

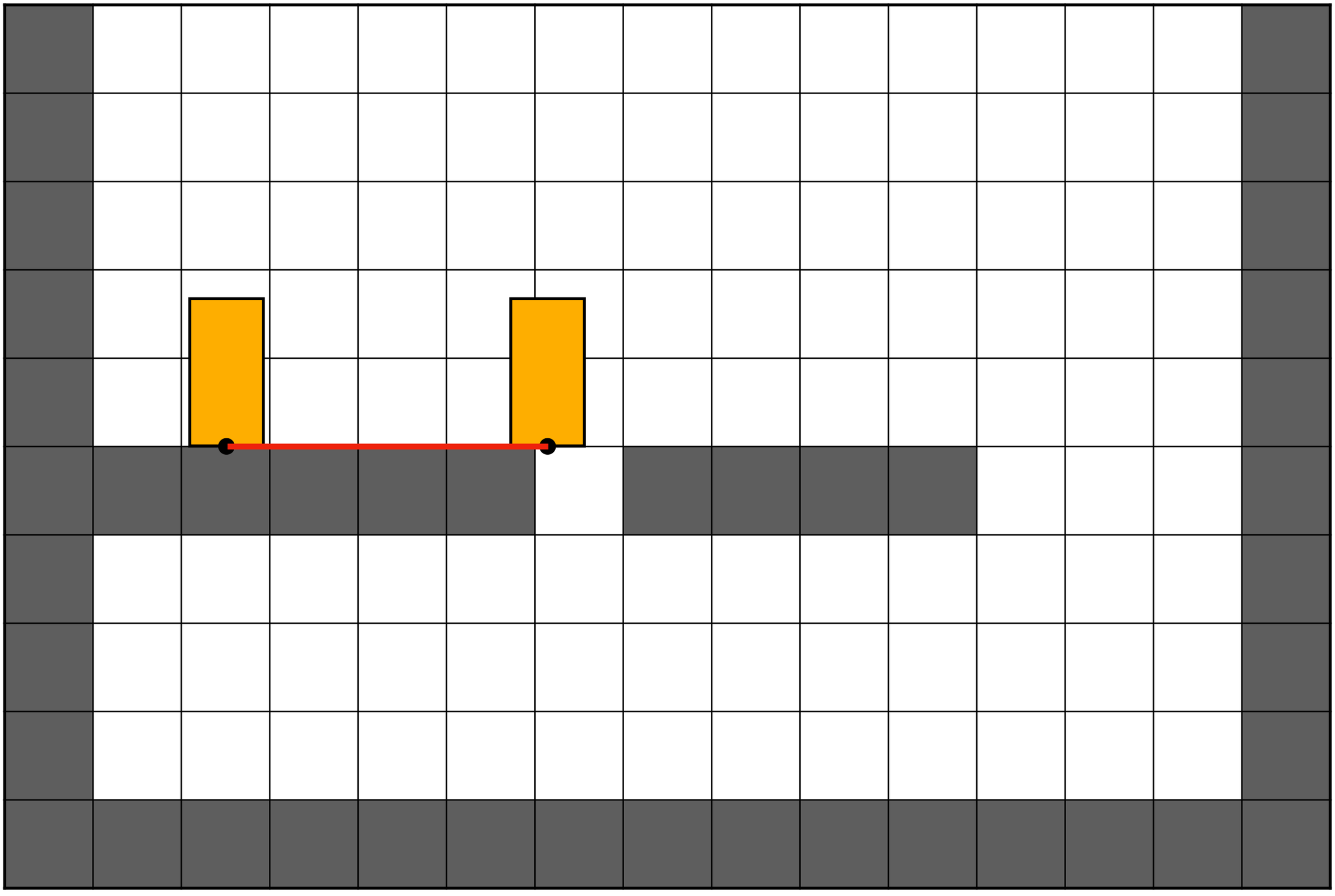

Hold right

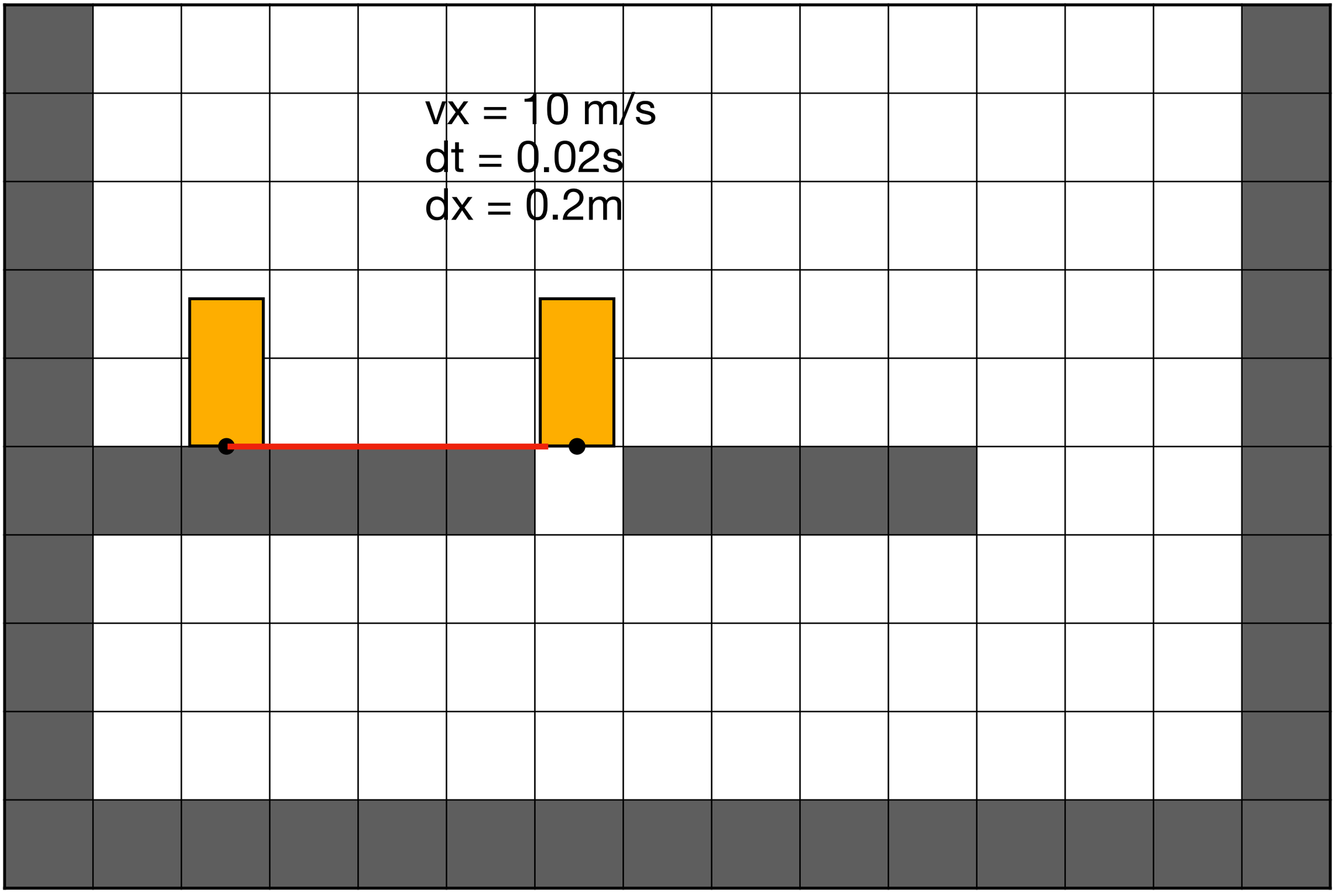

Stop

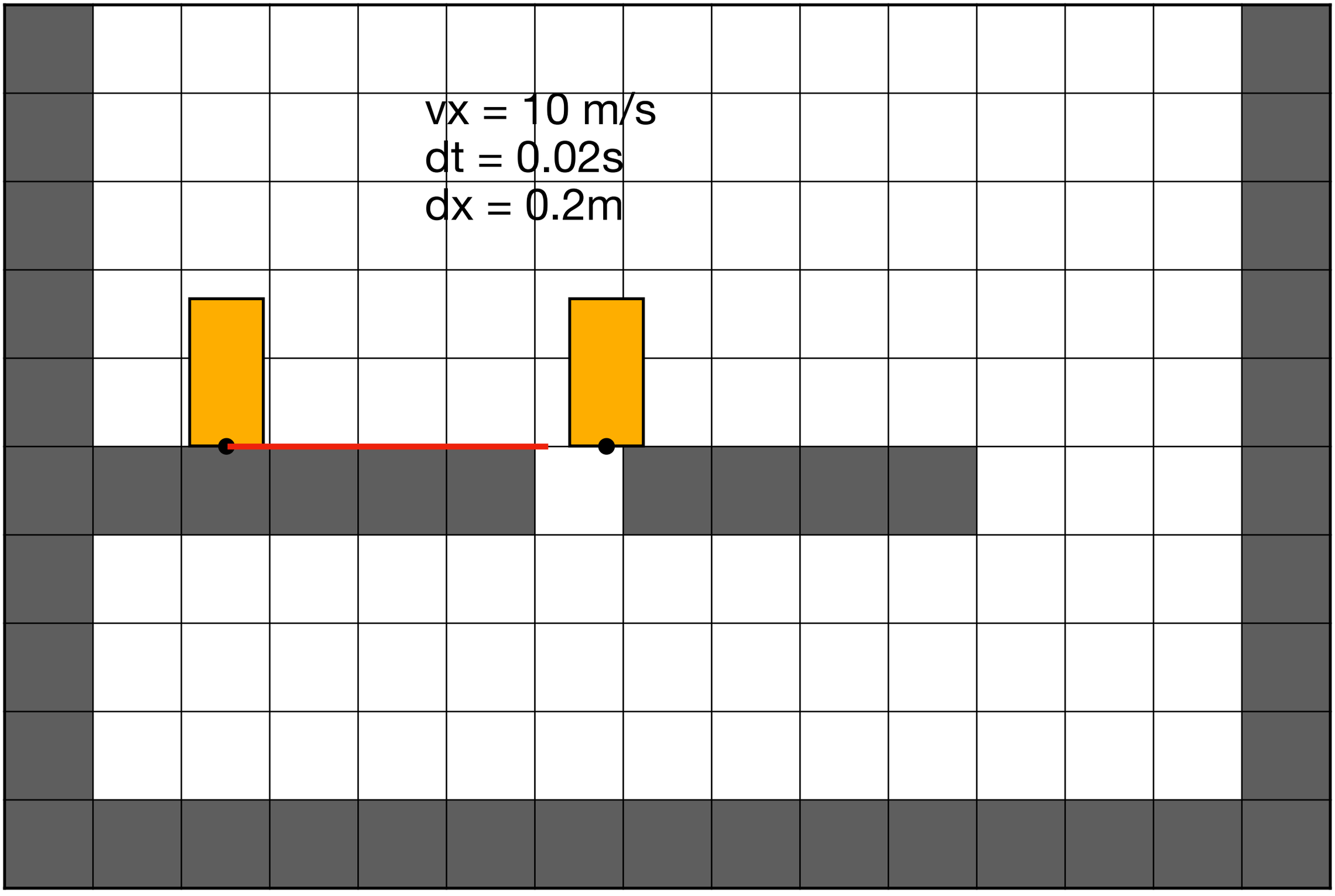

Stop
Sinks
Rests
on edge

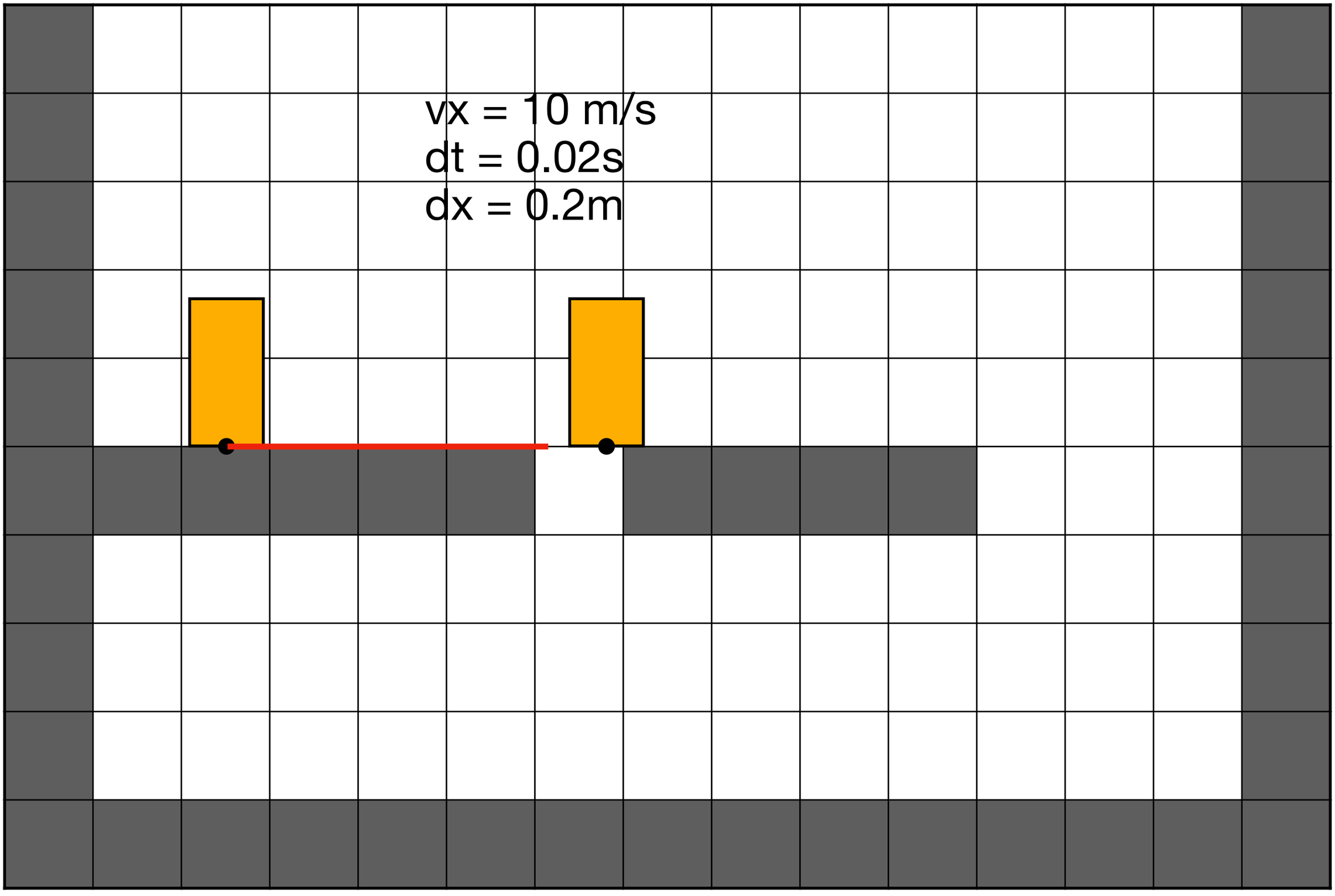


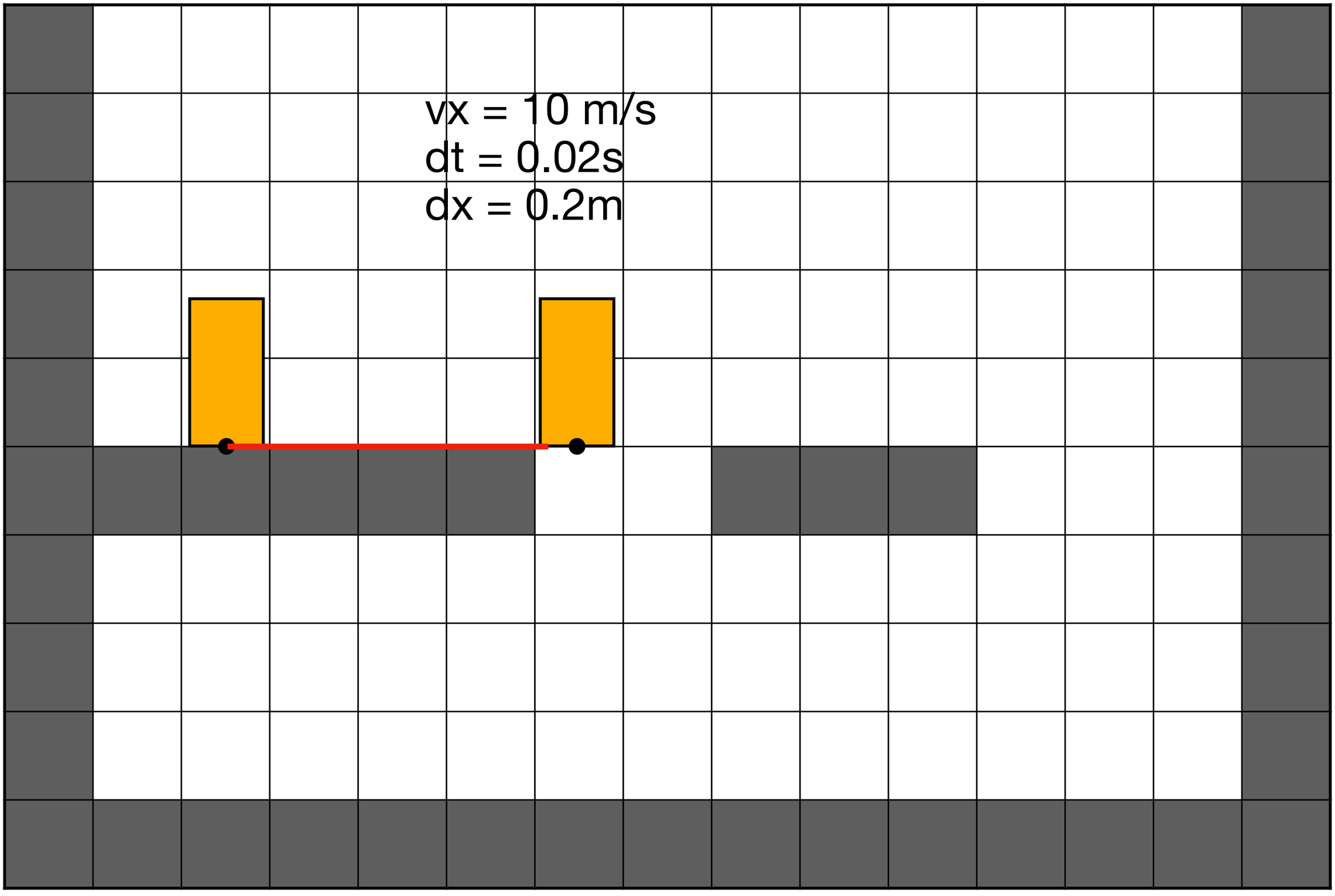

Stop

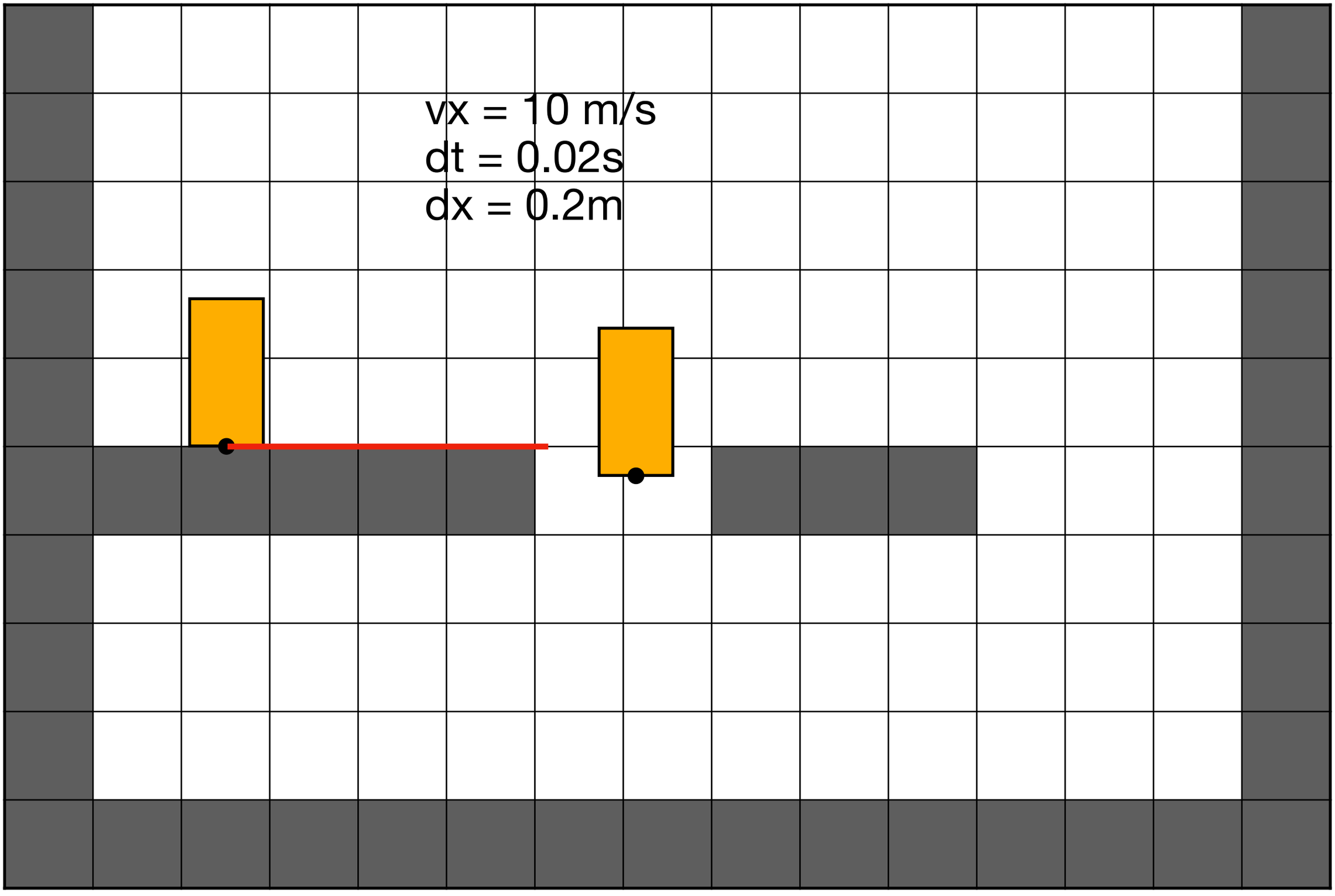


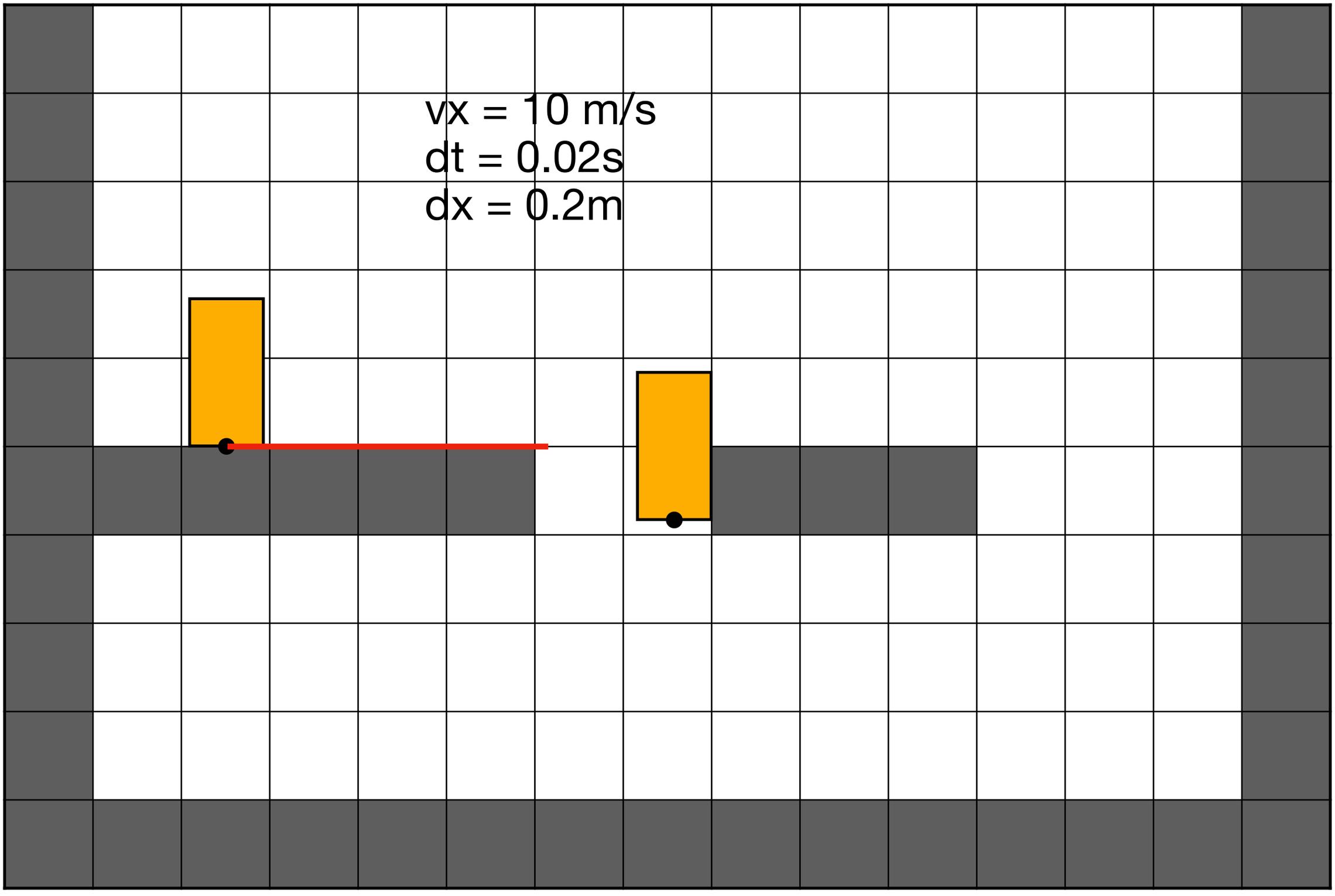

vx = 10 m/s
dt = 0.02s
dx = 0.2m

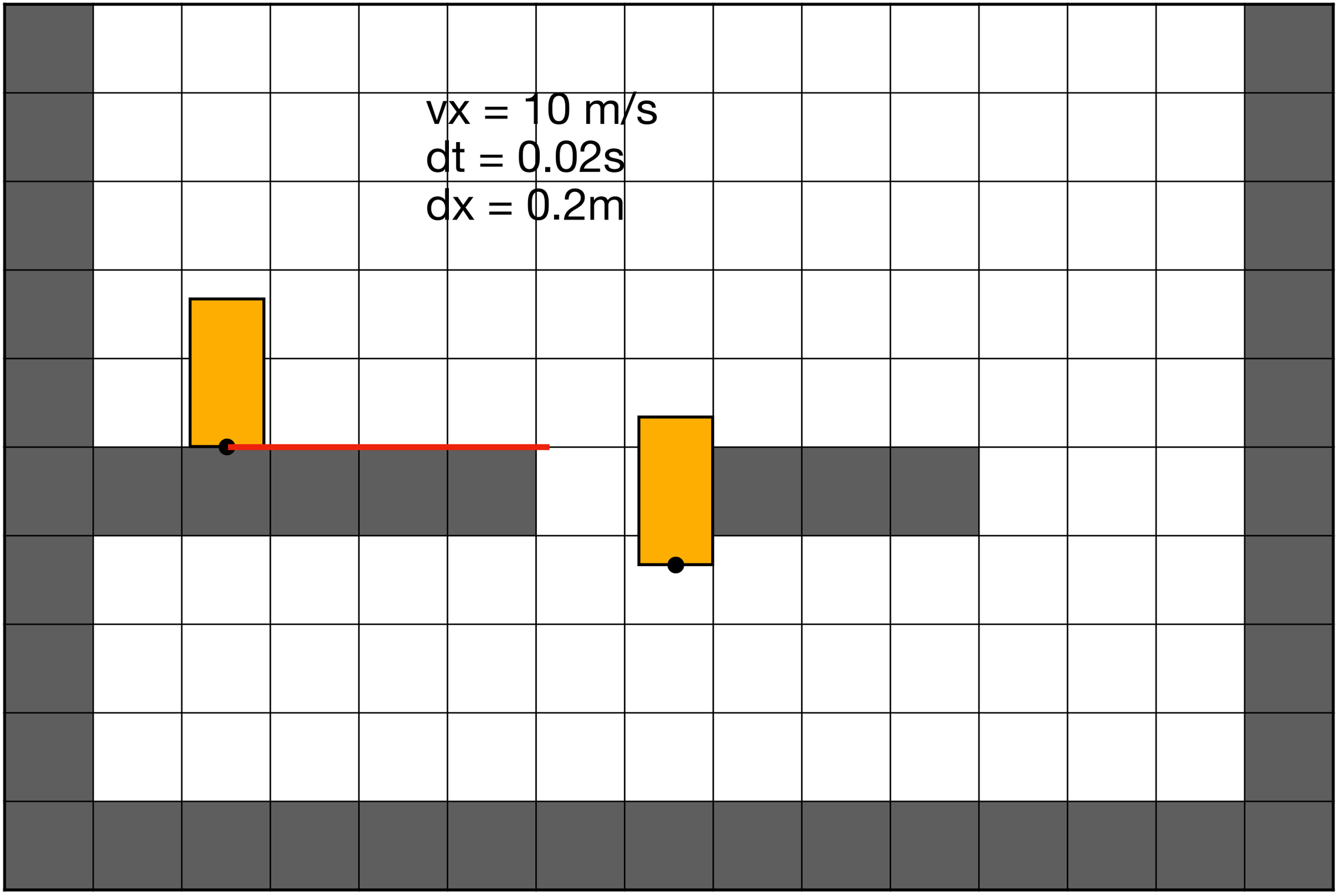

vx = 10 m/s
dt = 0.02s
dx = 0.2m

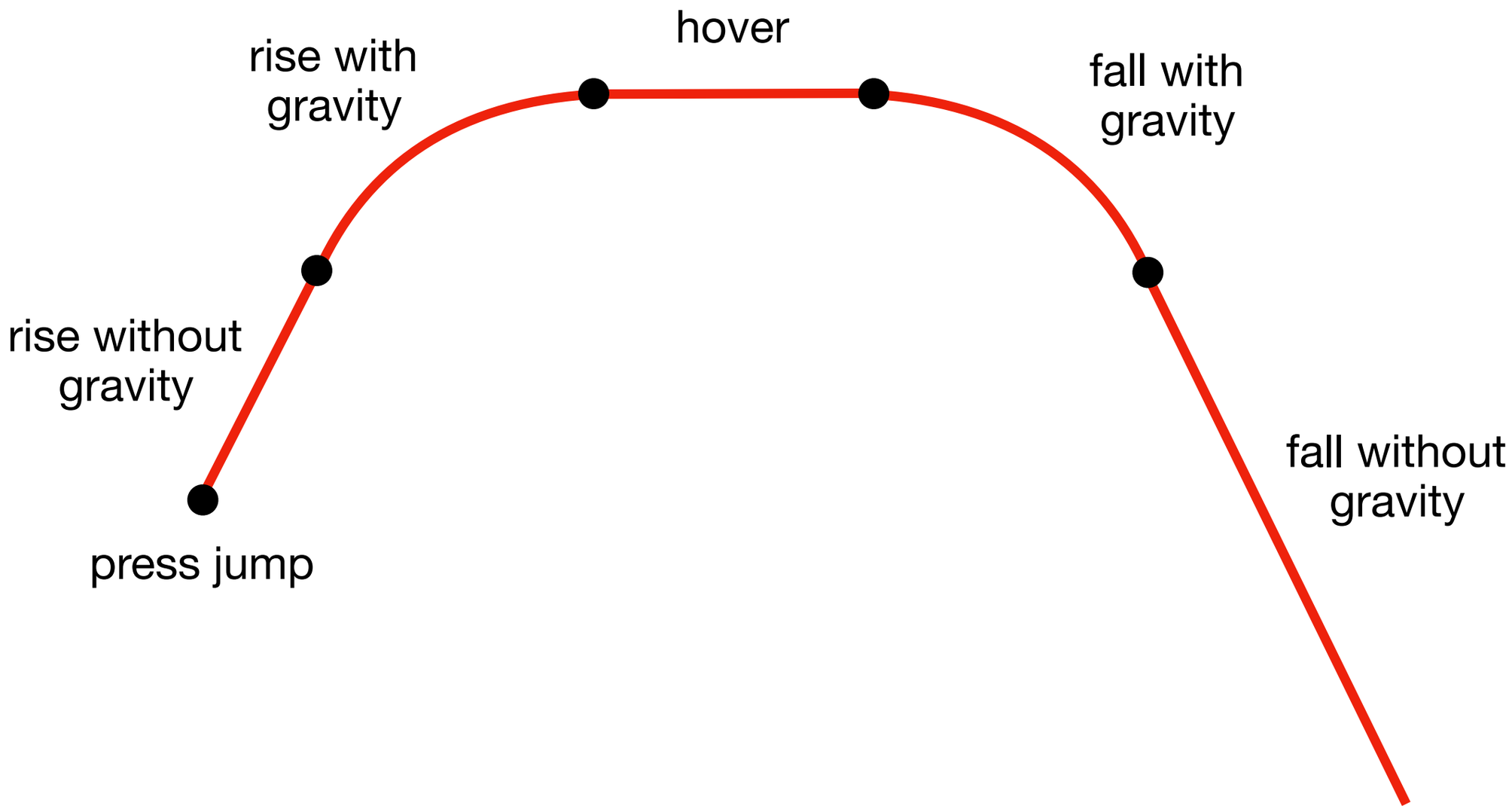

vx = 10 m/s
dt = 0.02s
dx = 0.2m


vx = 10 m/s
dt = 0.02s
dx = 0.2m

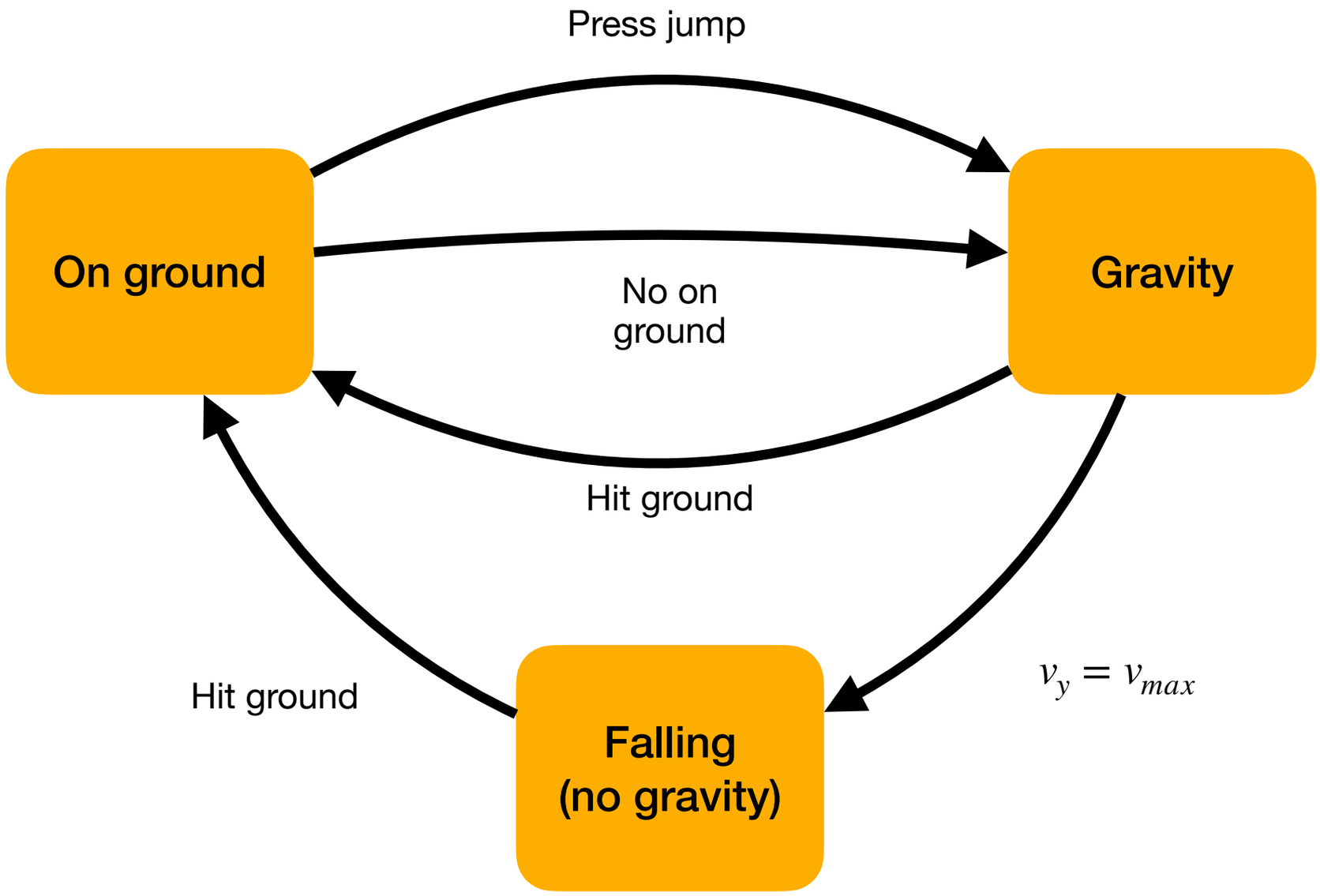

vx = 10 m/s
dt = 0.02s
dx = 0.2m


vx = 10 m/s
dt = 0.02s
dx = 0.2m

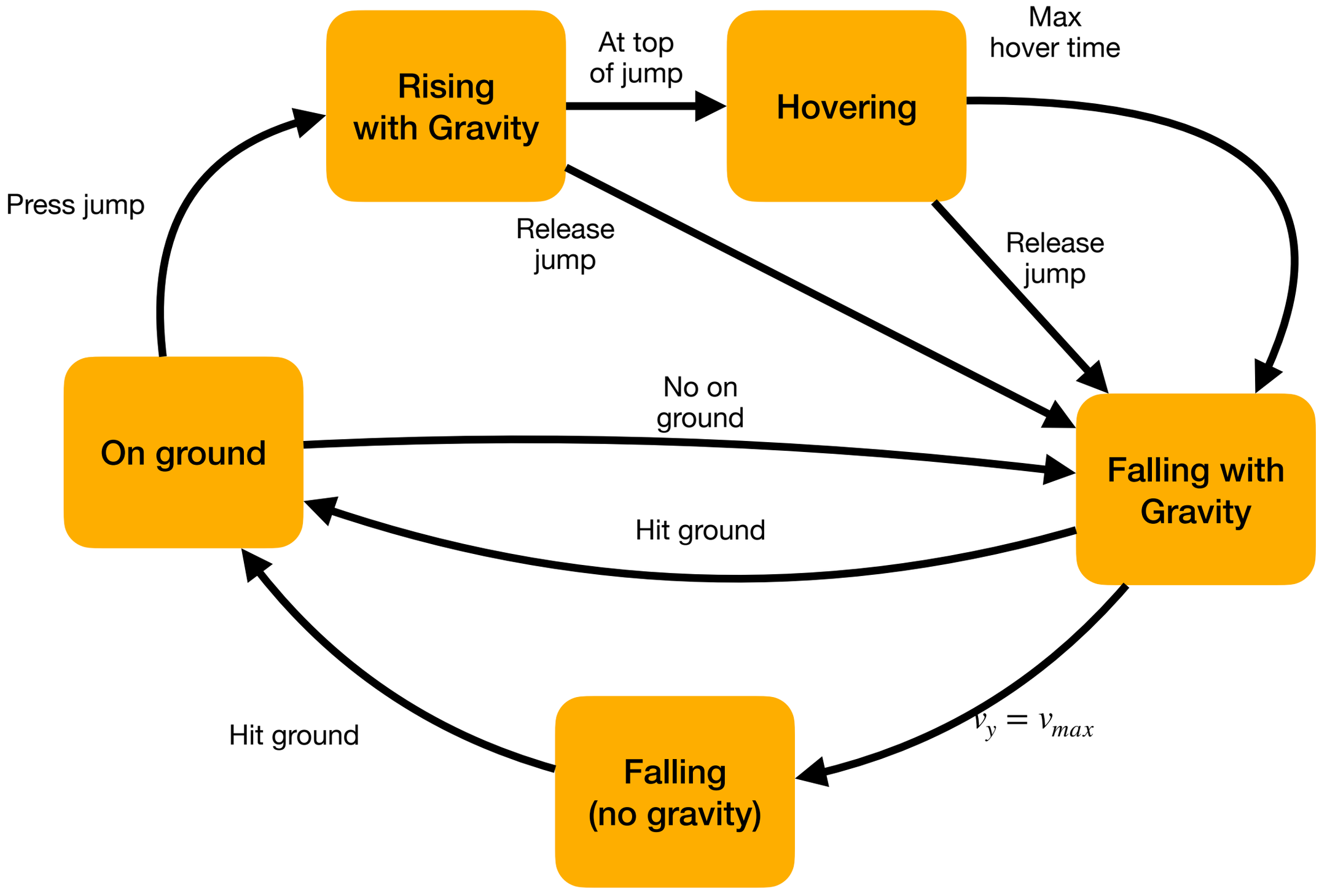

vx = 10 m/s
dt = 0.02s
dx = 0.2m

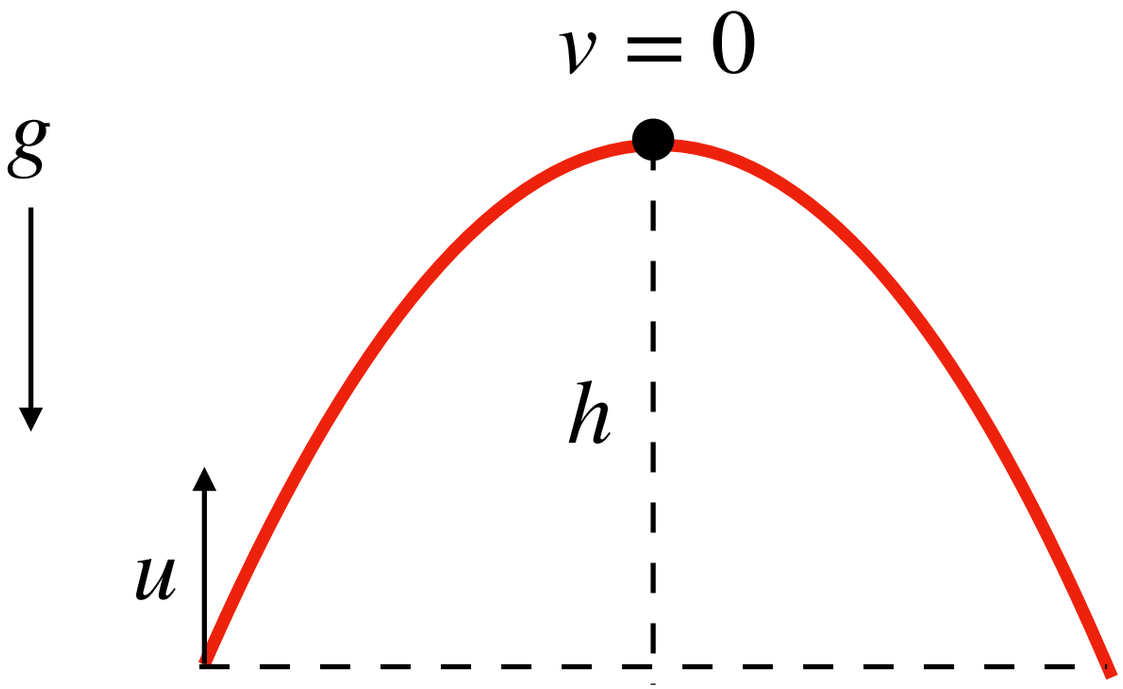

hover
rise with gravity
fall with gravity
rise without gravity
fall without gravity
press jump

State machine



State machine





$$v = u + at$$

$$s = ut + \frac{1}{2}at^2$$

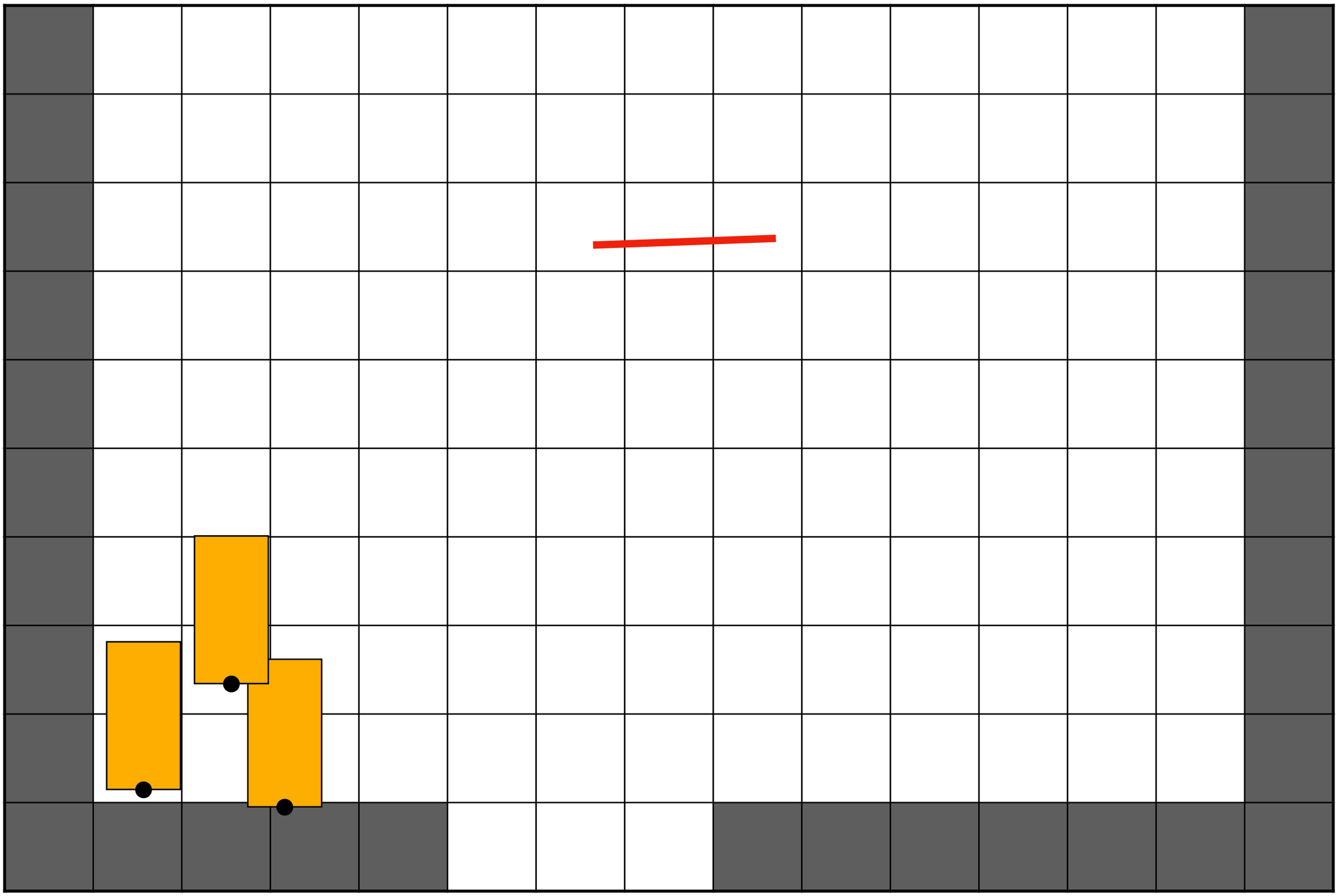
$$v^2 = u^2 + 2as$$

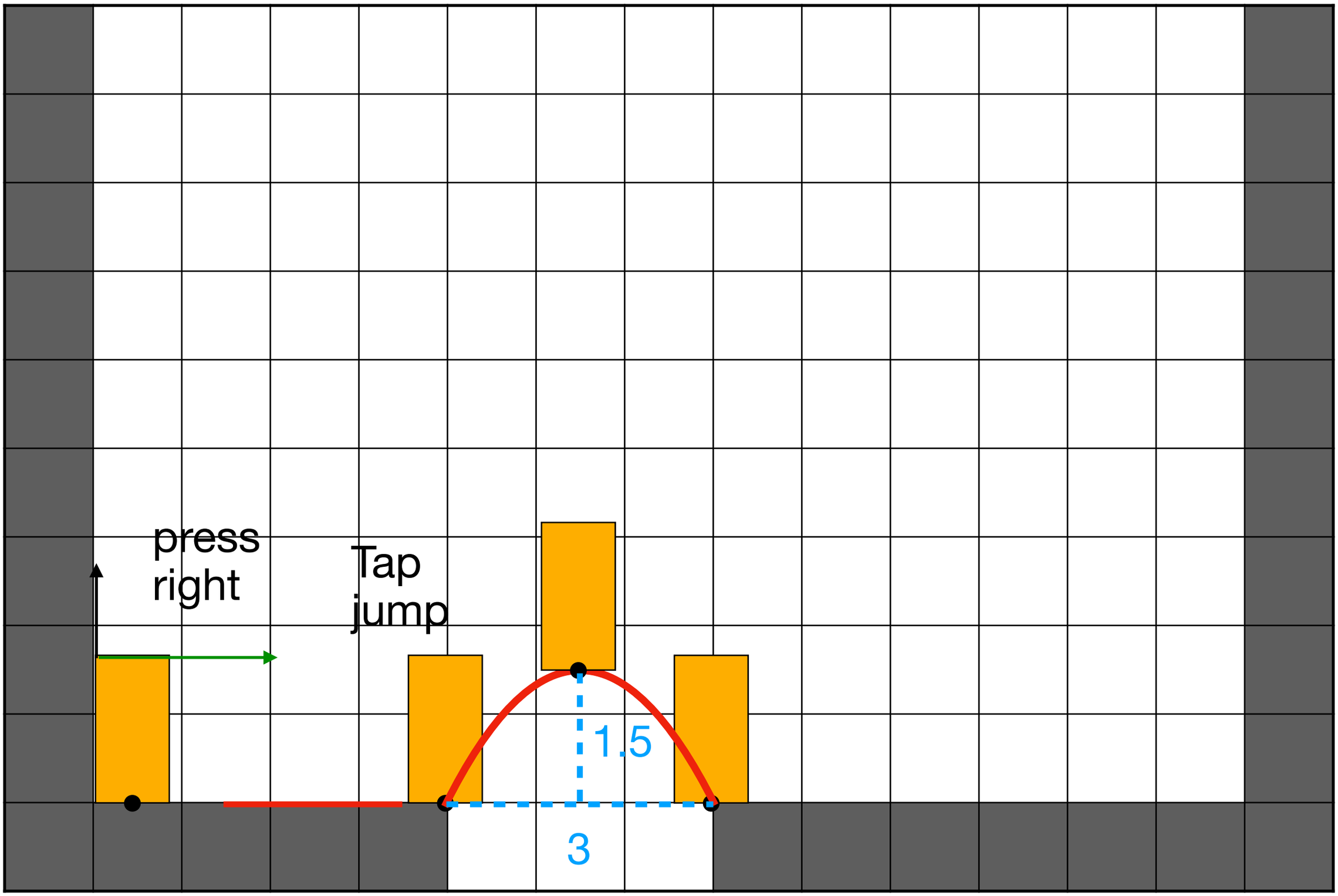
$$v^2 = u^2 + 2gh$$

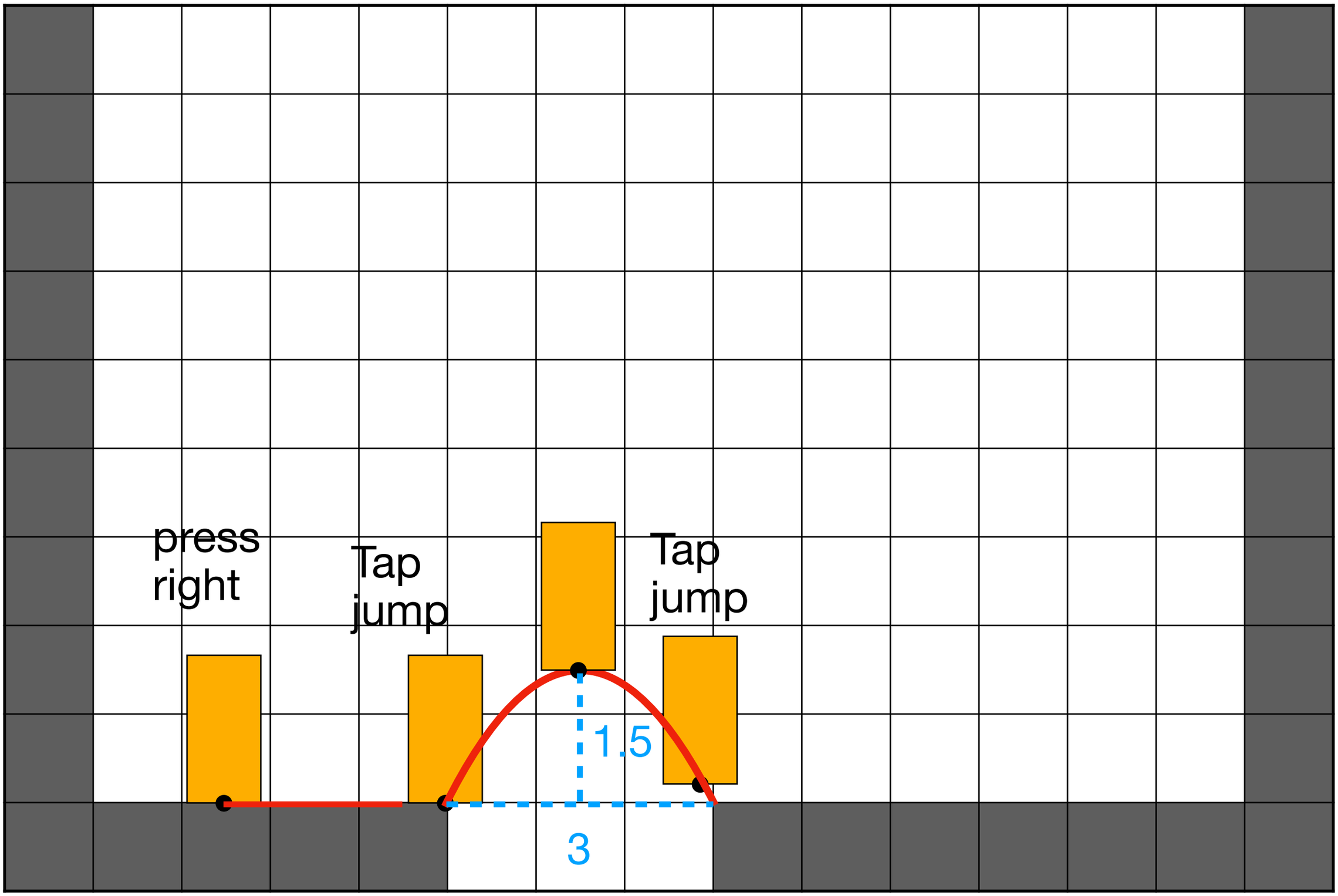
$$0 = u^2 + 2gh$$

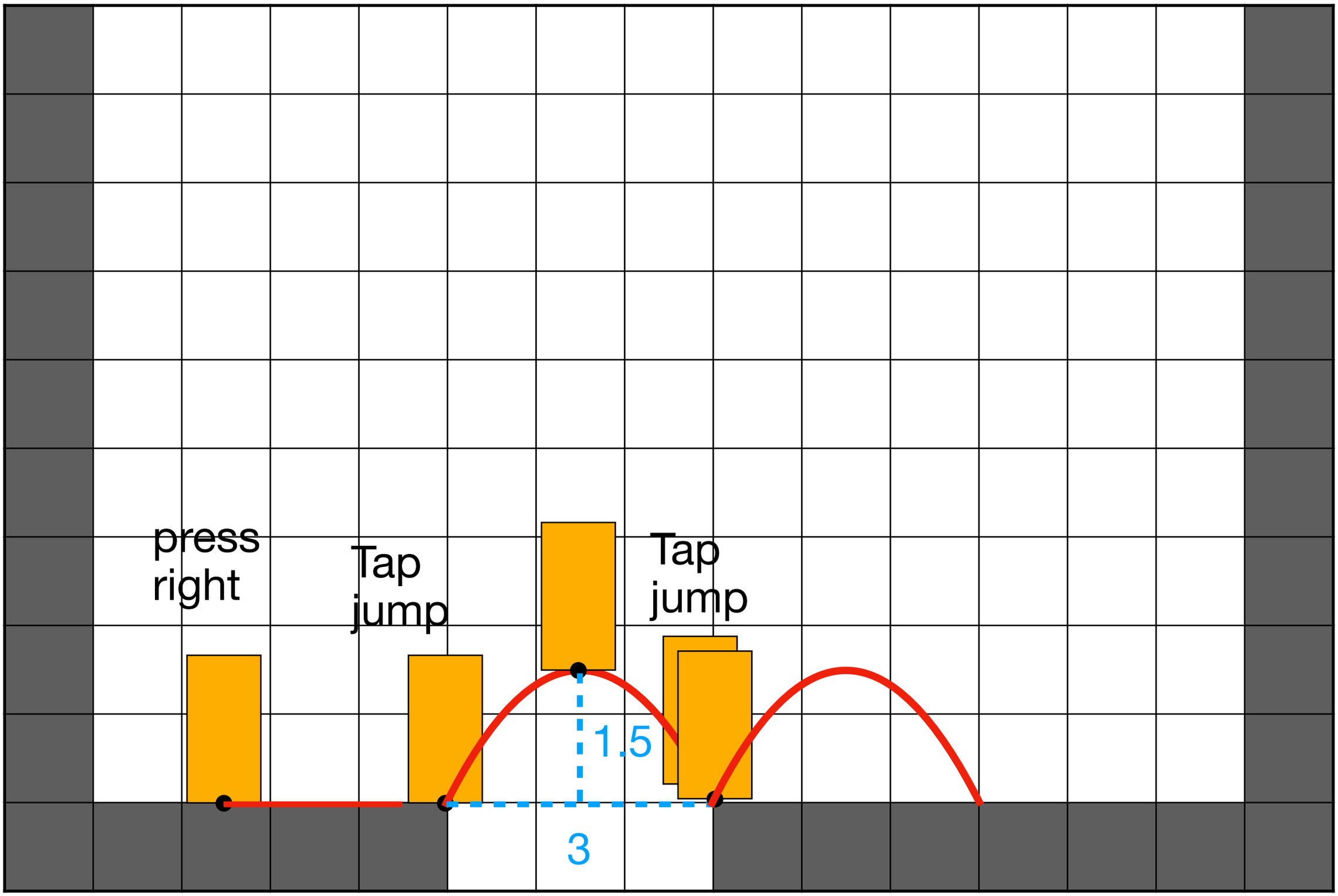
$$u^2 = -2gh$$

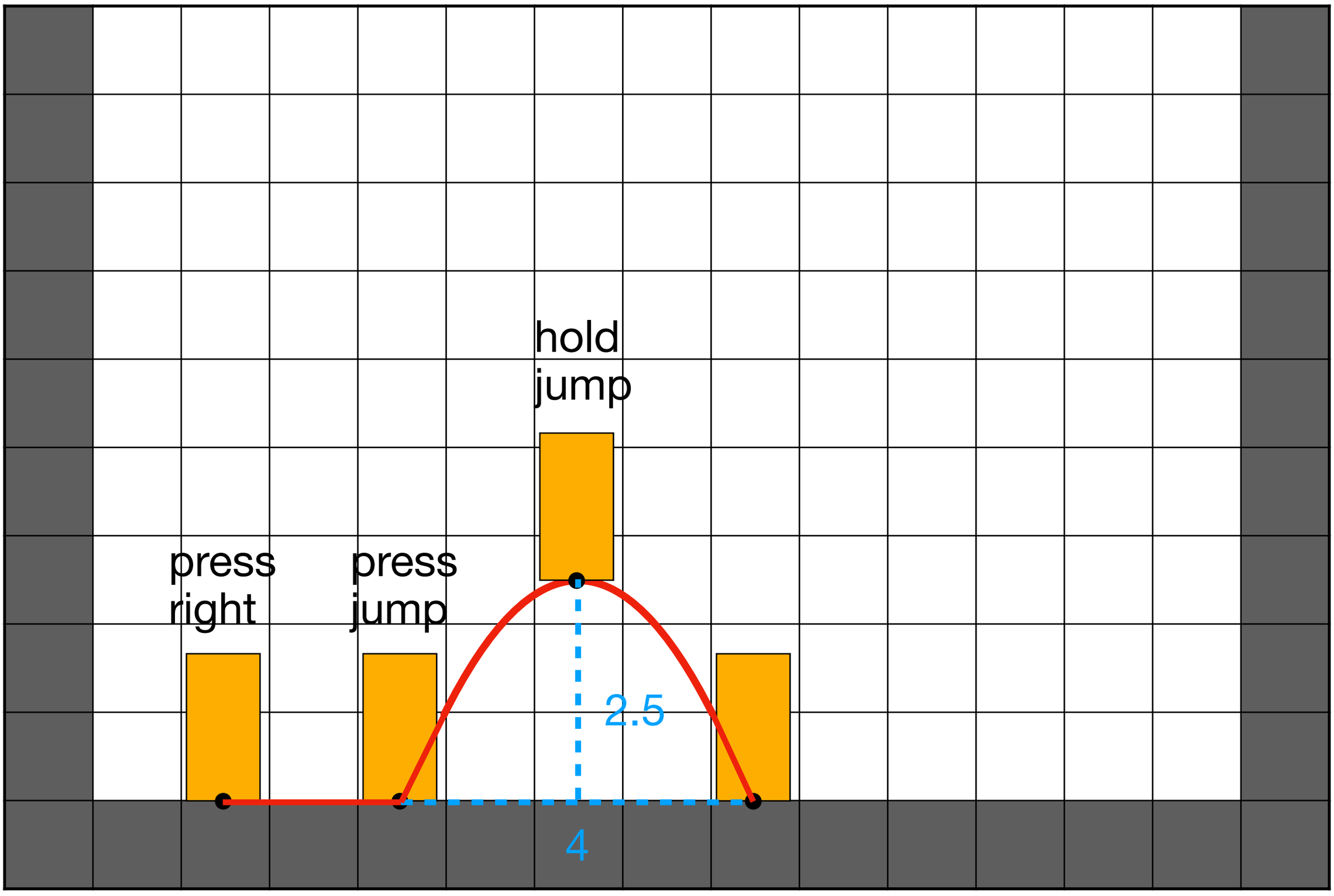
$$u = \sqrt{-2gh}$$




press
right
Tap
jump
1.5
3


press
right
Tap
jump
Tap
jump
1.5
3


press
right
Tap
jump
Tap
jump
1.5
3


hold
jump
press
right
press
jump
2.5
4

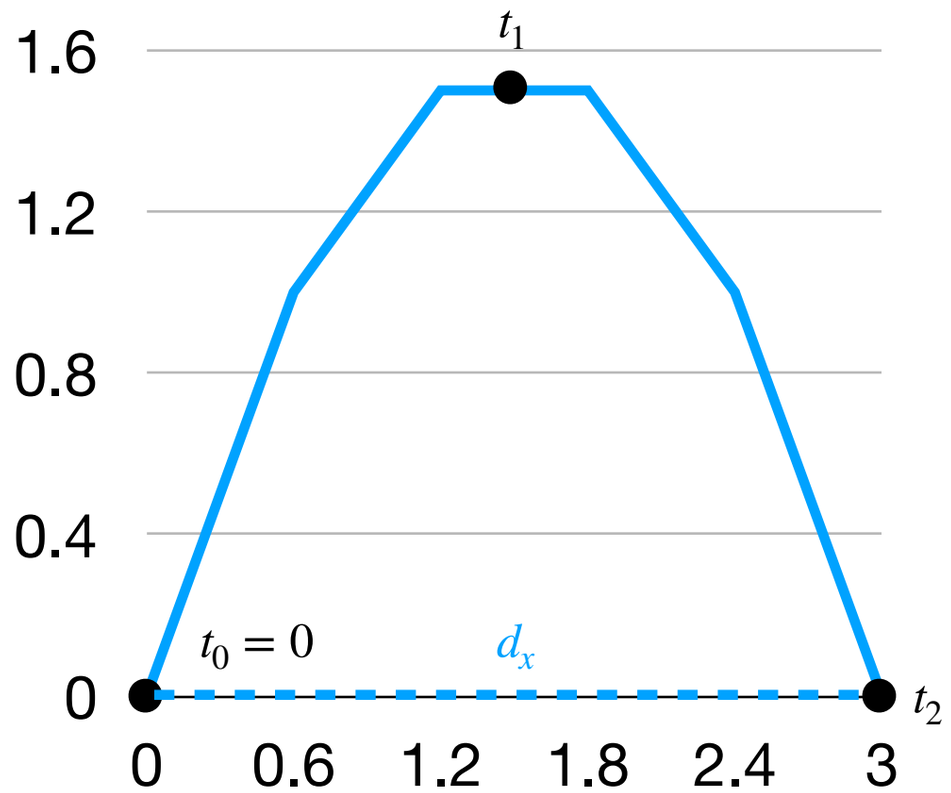
Some math

Short jump

v_x = horizontal speed.

d_x = horizontal jump distance

$t_2 = d_x / v_x$ = jump time



Some math

Short jump

t_1

u_y = initial vertical jump speed

g = acceleration due to gravity

$v_y(t) = u_y + gt$ = vertical speed at time t

$d_y(t)$ = the height at time t

$d_y(t) = u_y t + gt^2/2$

Some math

Short jump

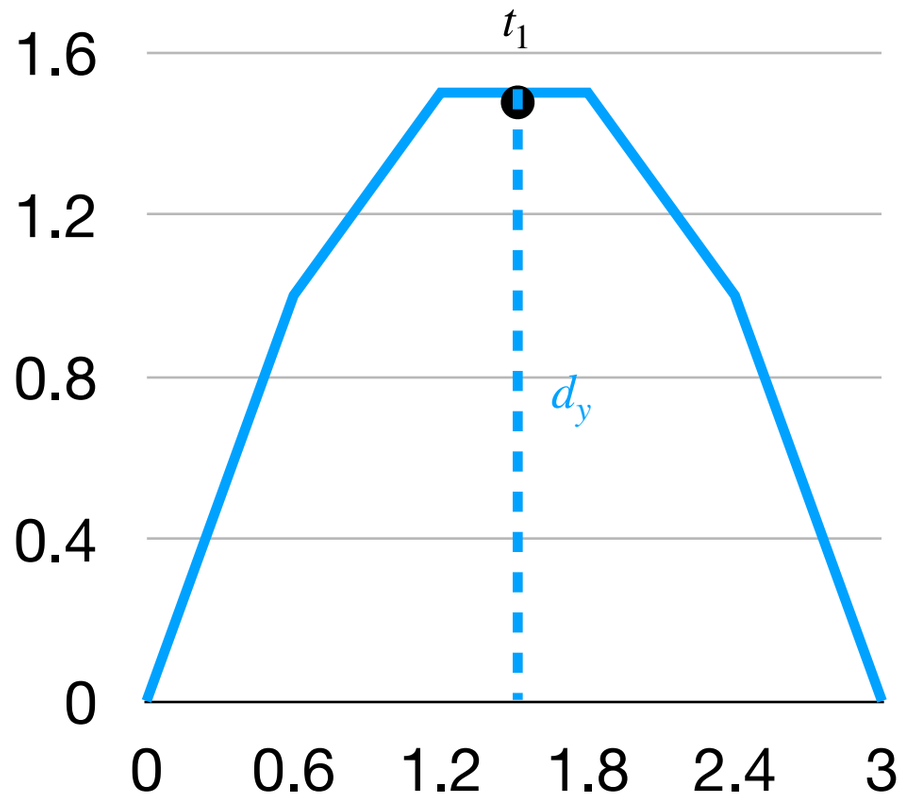
At the highest point of the jump:

$$t_1 = t_2/2$$

$$v_y(t_1) = 0$$

$$d_y(t_1) = h$$

Solve to find u_y and g



Some math

Short jump

$$v_y(t_1) = 0$$

$$u_y + gt_1 = 0$$

$$u_y = -gt_1$$

Some math

Short jump

$$d_y(t_1) = h$$

$$u_y t_1 + g t_1^2/2 = h$$

$$(-g t_1) t_1 + g (t_1)^2/2 = h$$

$$-g (t_1)^2/2 = h$$

$$g = -2h/t_1^2$$

$$u_y = 2h/t_1$$

Some math

Short jump

Plugging in our desired values:

$$v_x = 5, d_x = 3, h = 1.5$$

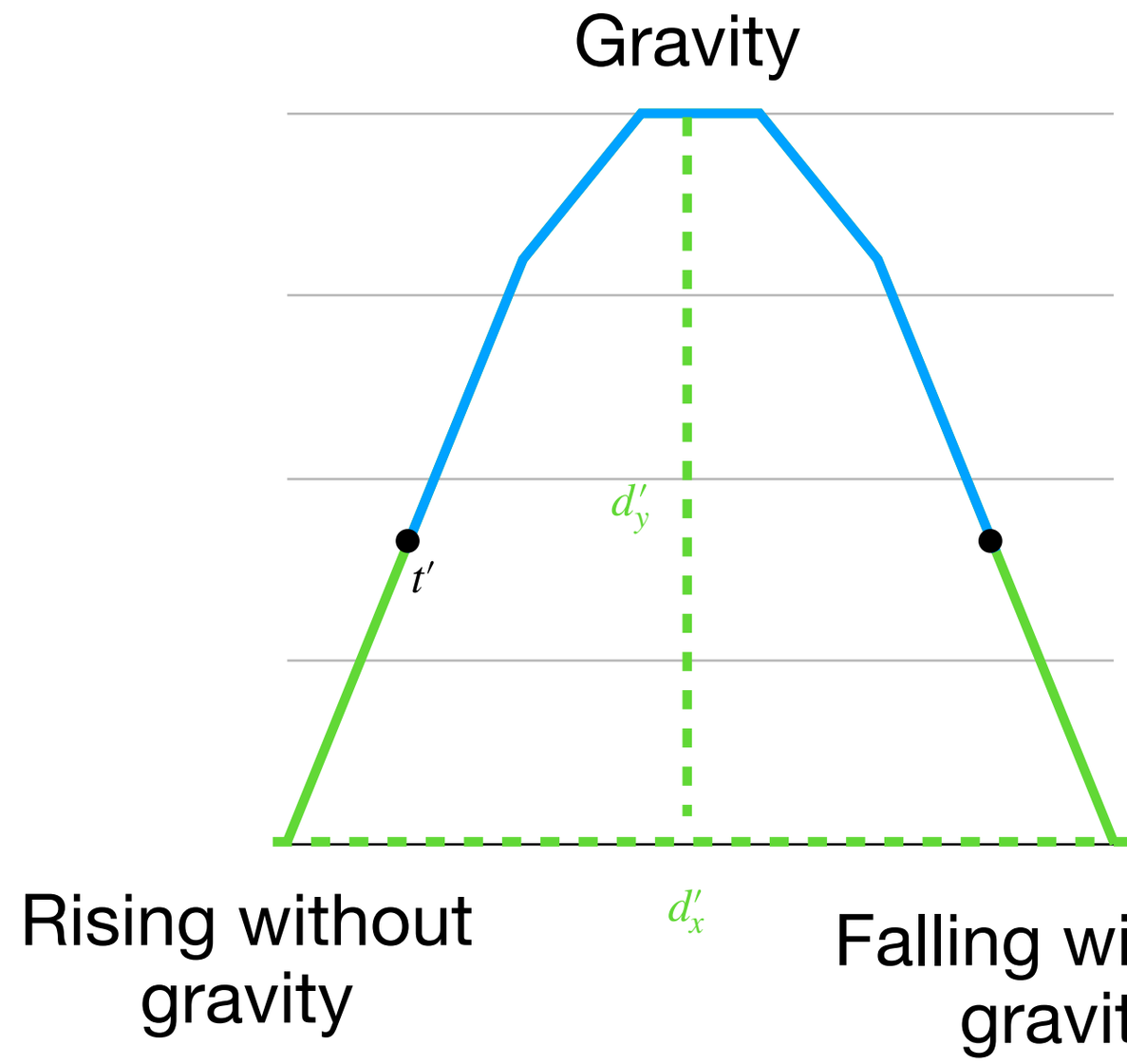
$$t_2 = 3/5, t_1 = 3/10$$

$$u_y = 2(1.5)/(3/10) = 10$$

$$g = -2(1.5)/(3/10)^2 = -100/3$$

Some math

High jump



Some math

High jump

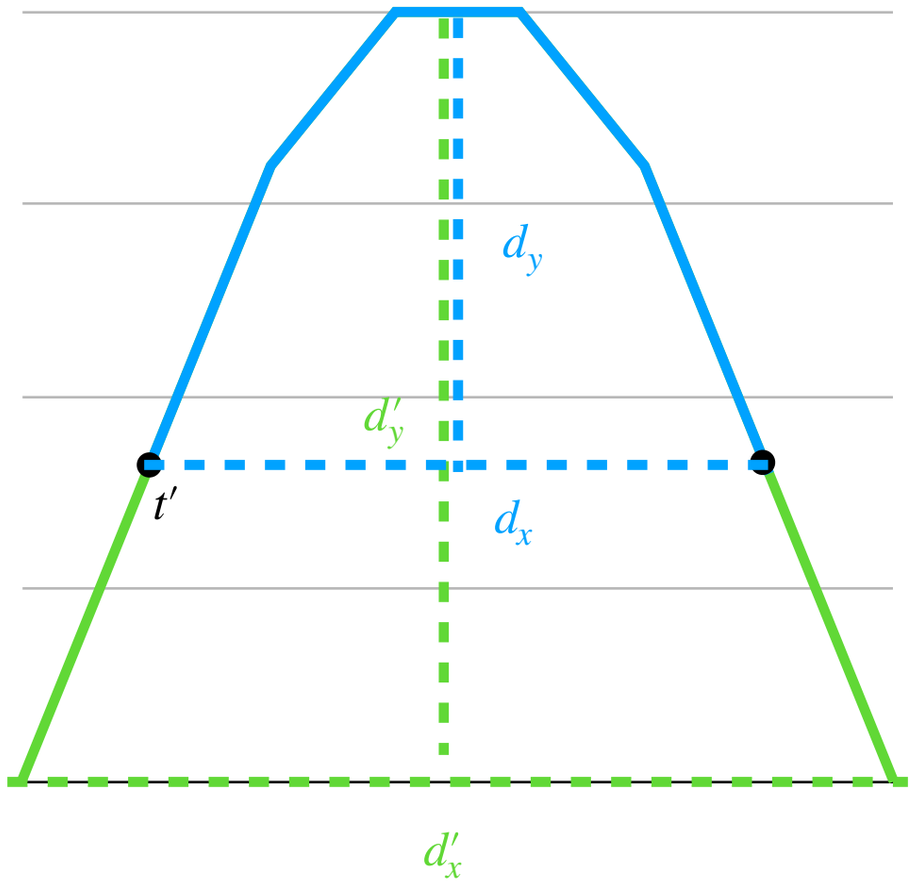
$$u_y t' = (d'_y - d_y)$$

$$t' = (d'_y - d_y)/u_y$$

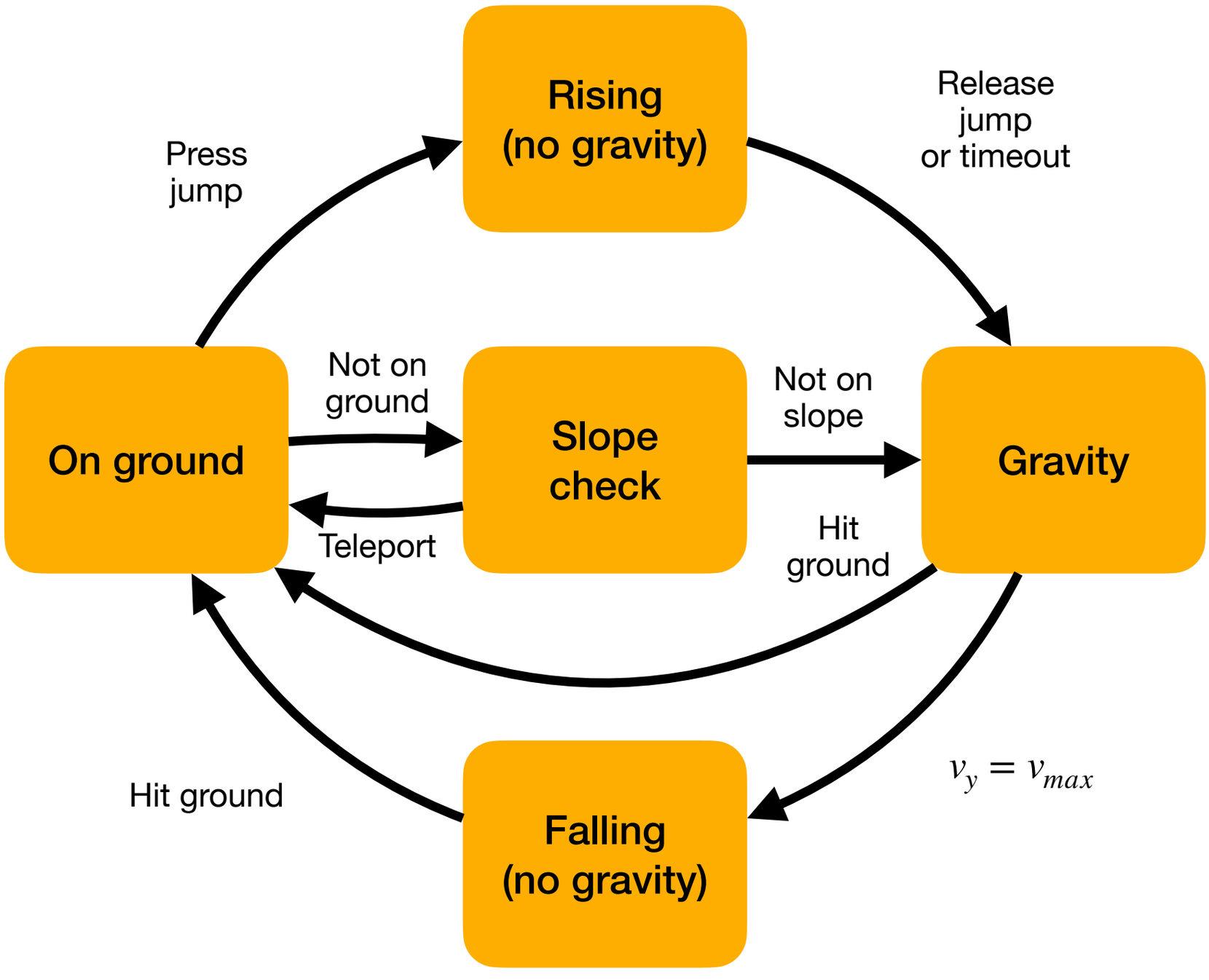
$$t' = 1/10$$

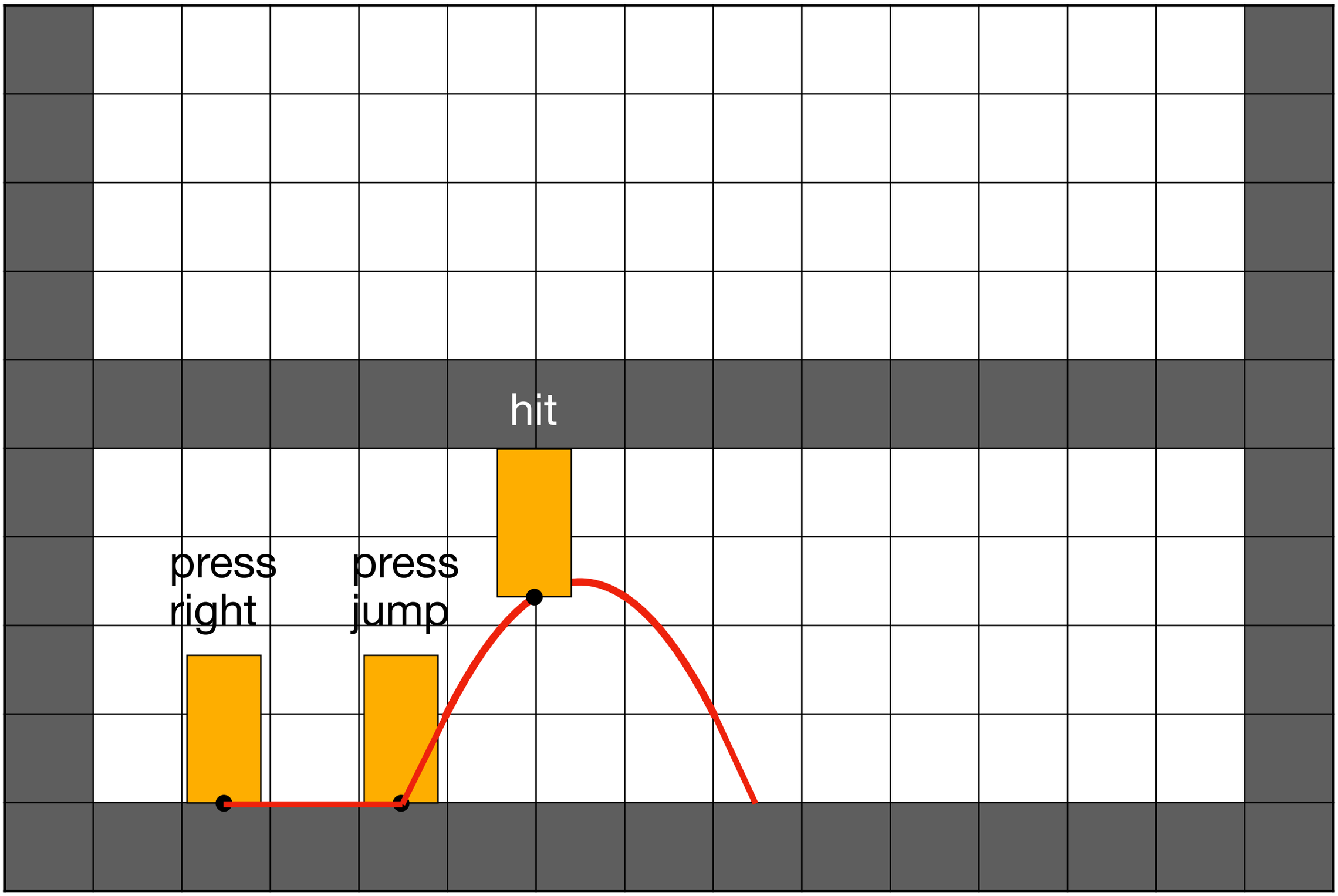
$$d'_x = d_x + 2v_x t'$$

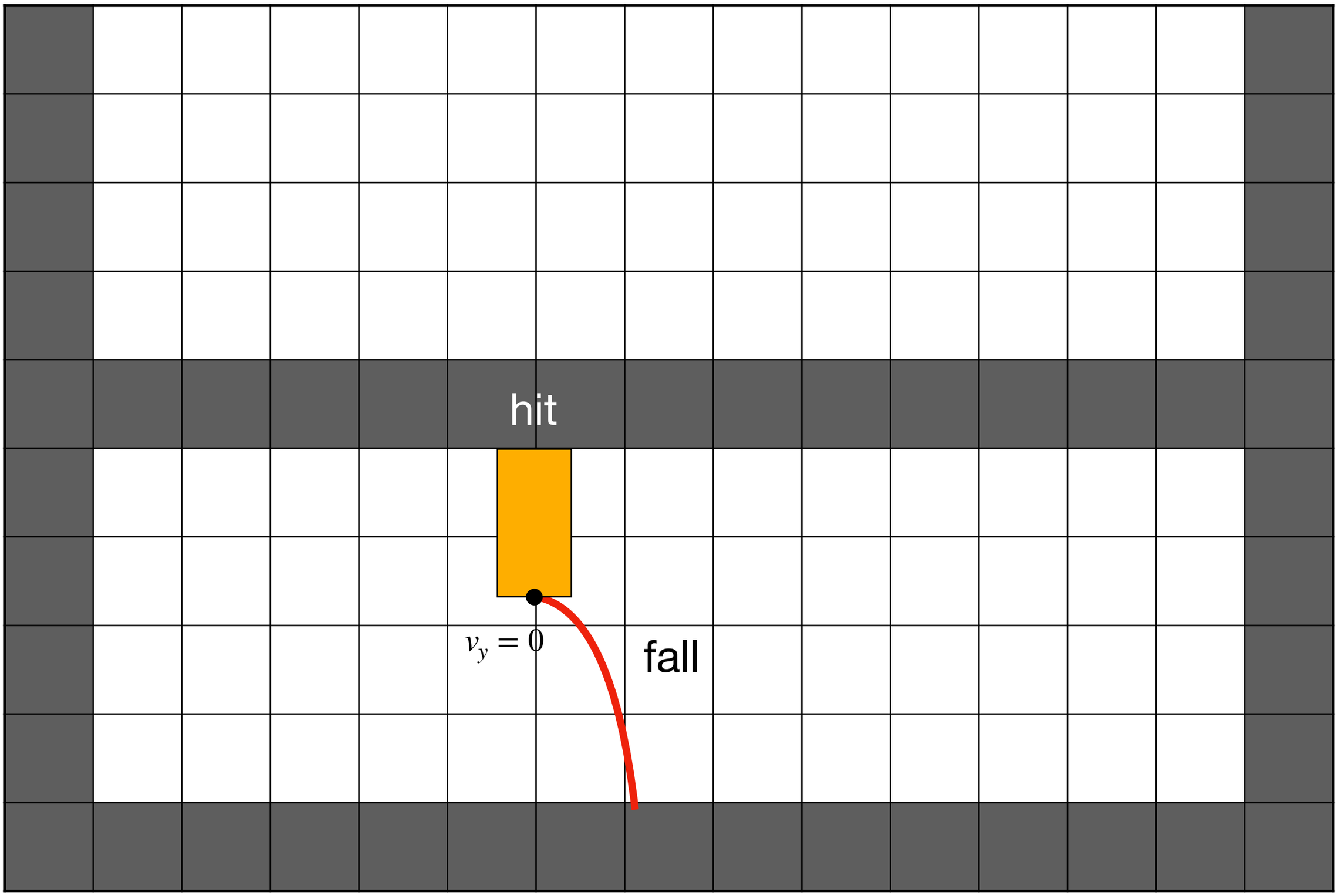
$$d'_x = 3 + 2(5)/10 = 4$$



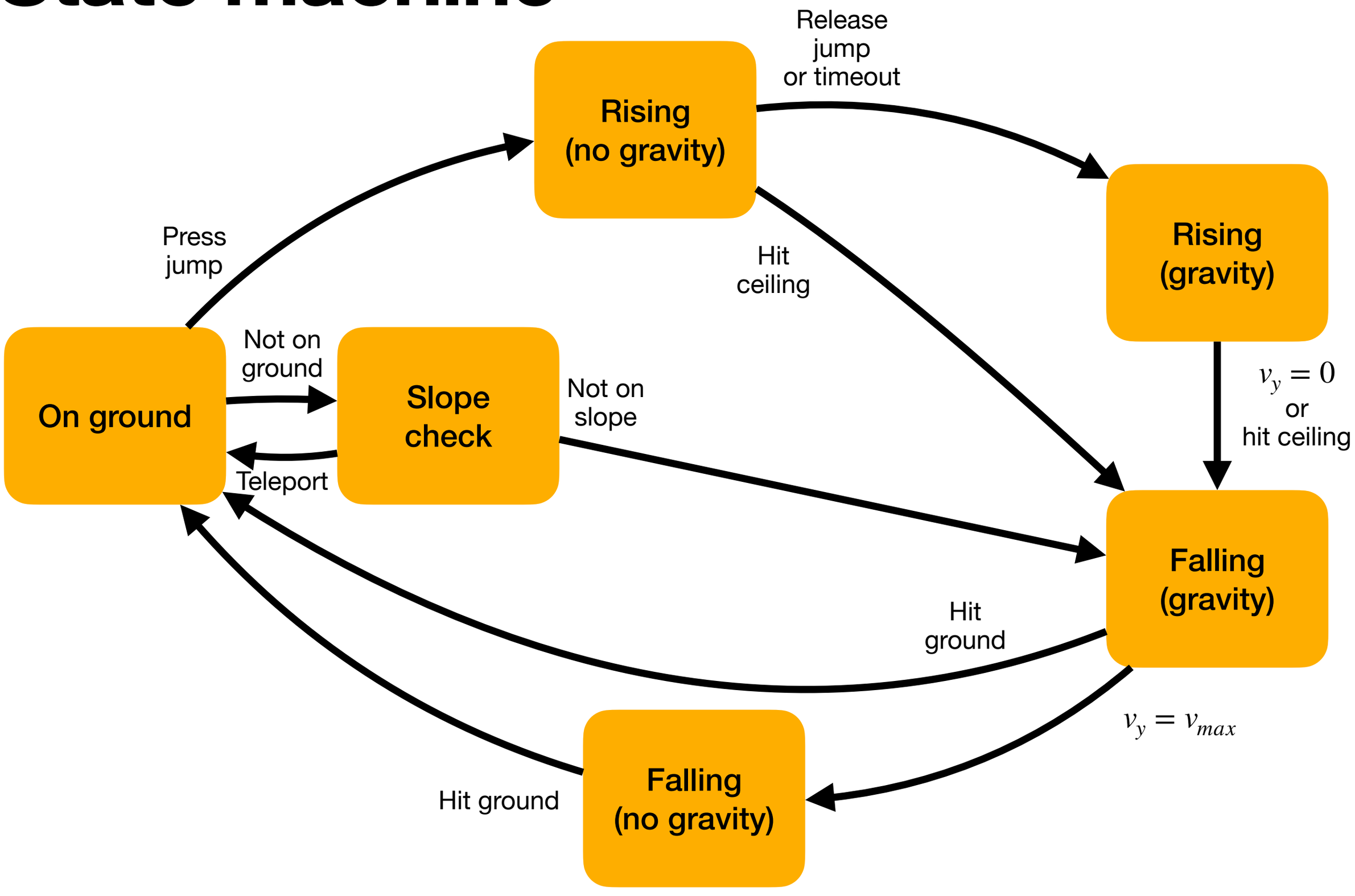
State machine

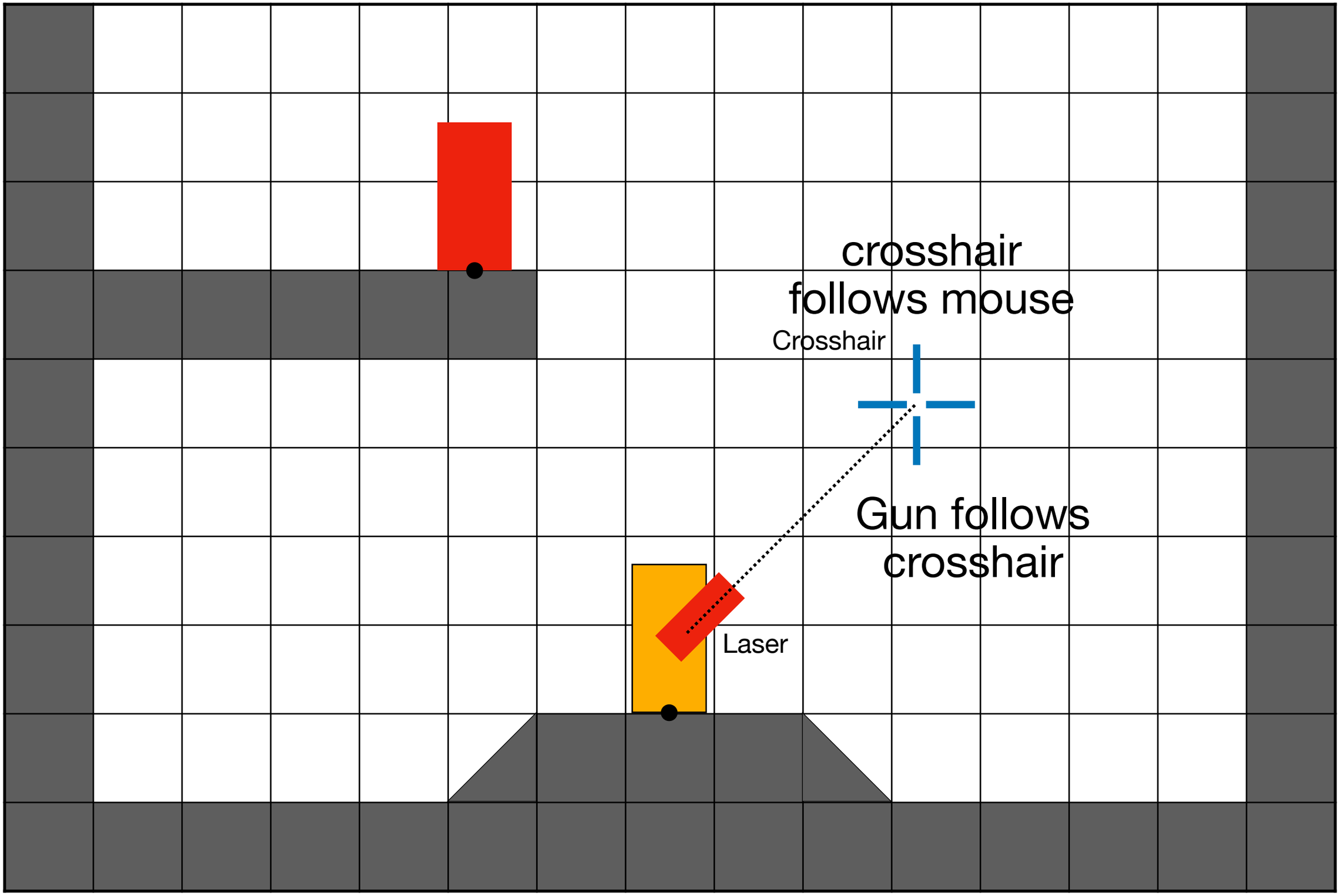


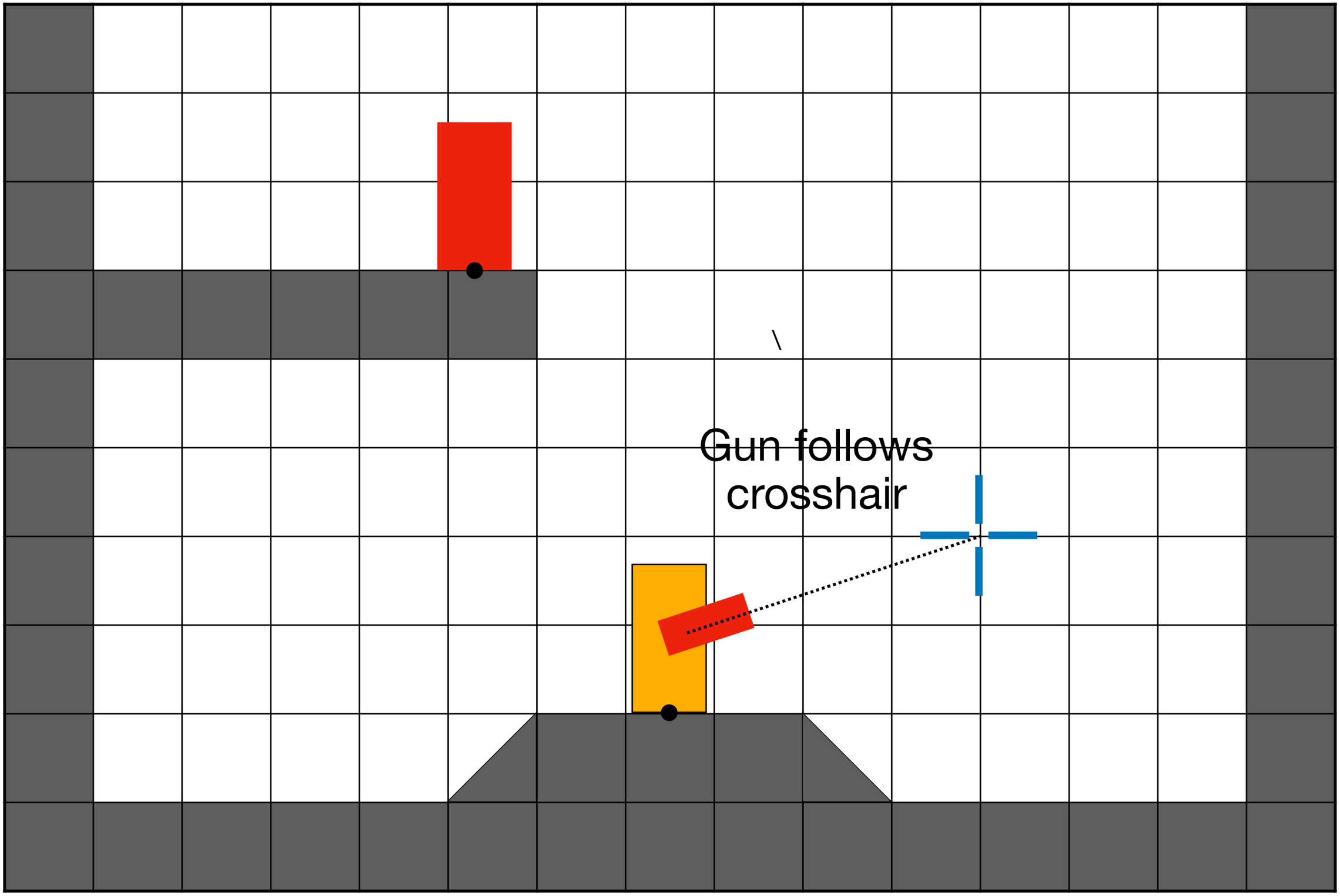

hit
press
right
press
jump

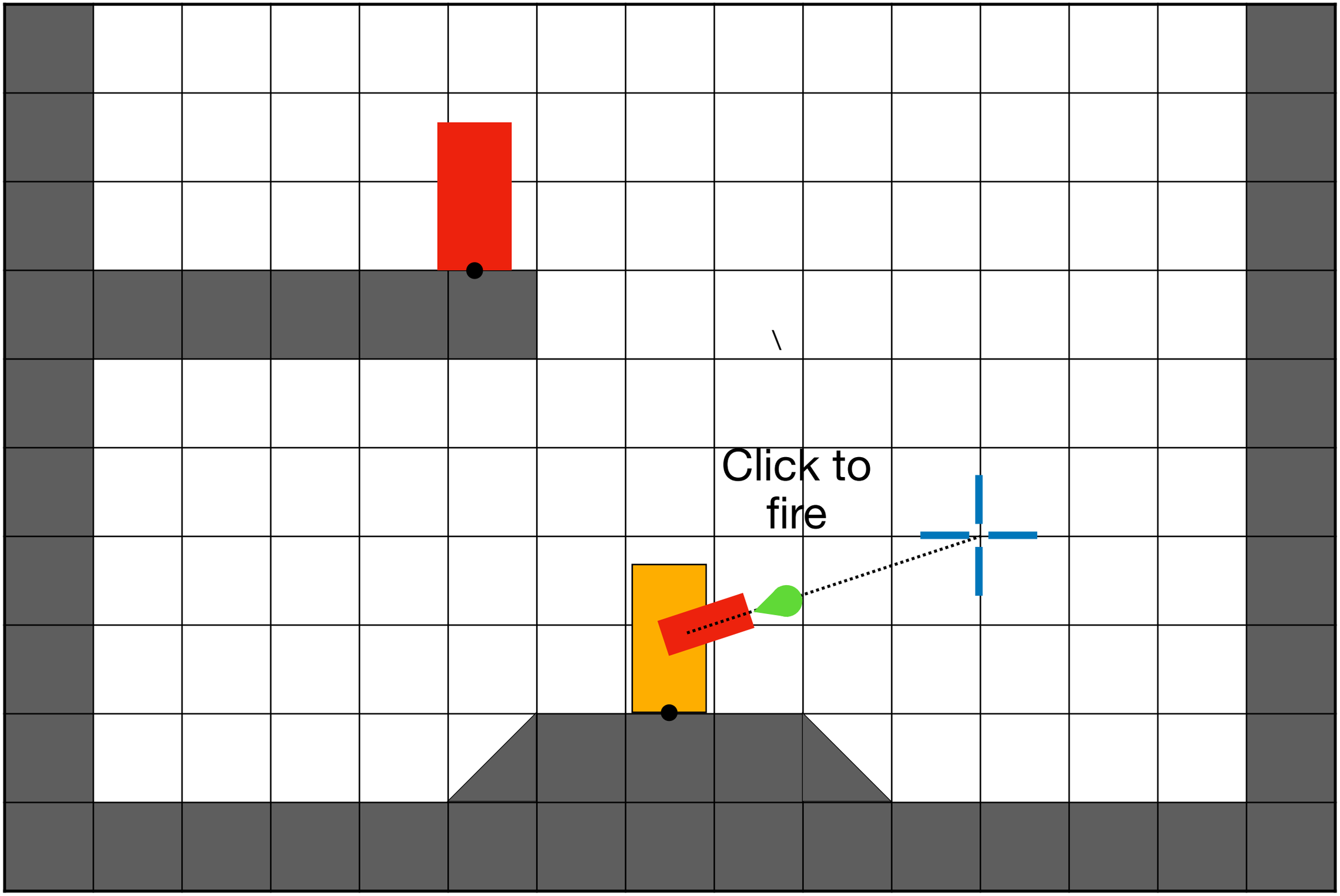

hit
$v_y = 0$
fall

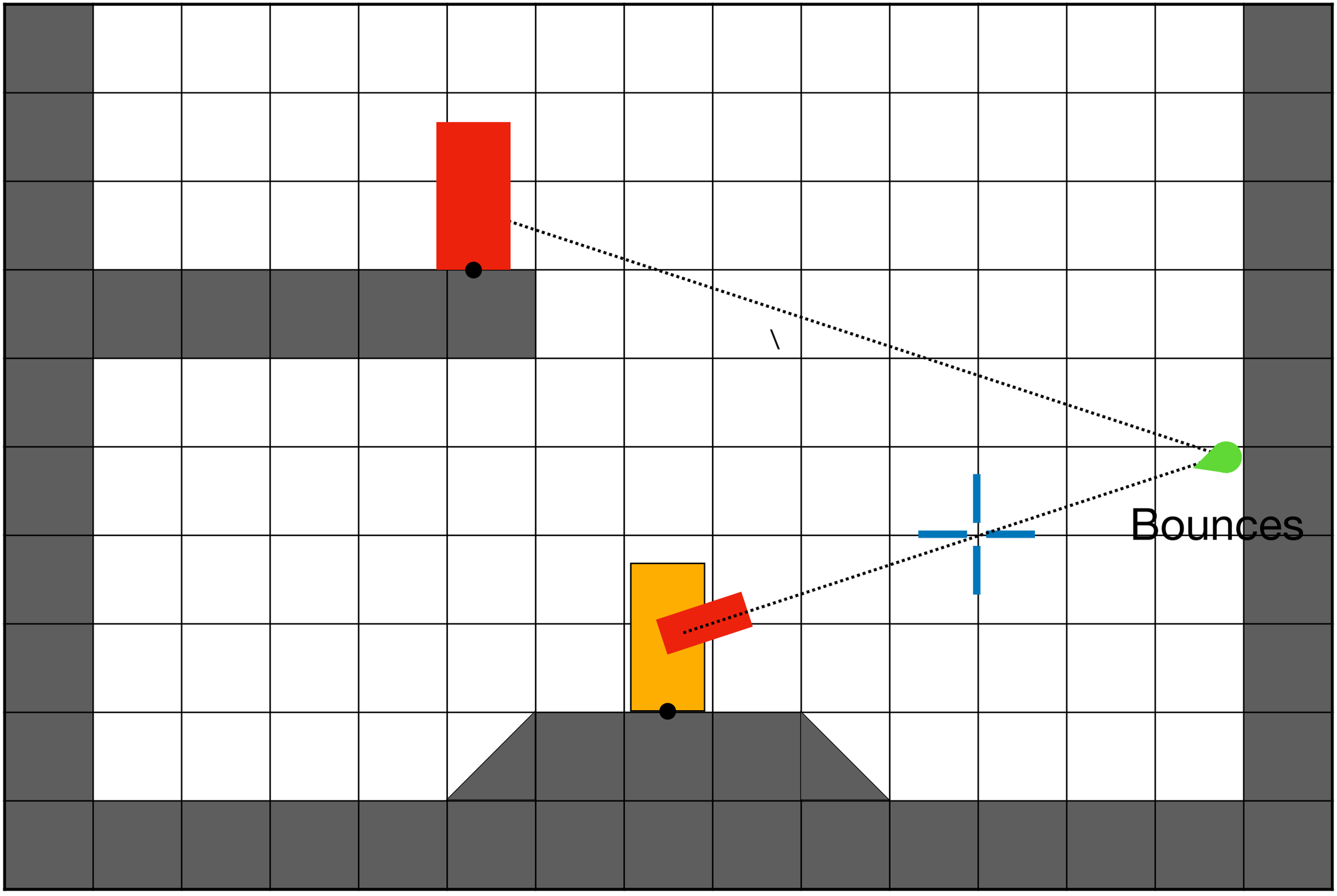
State machine

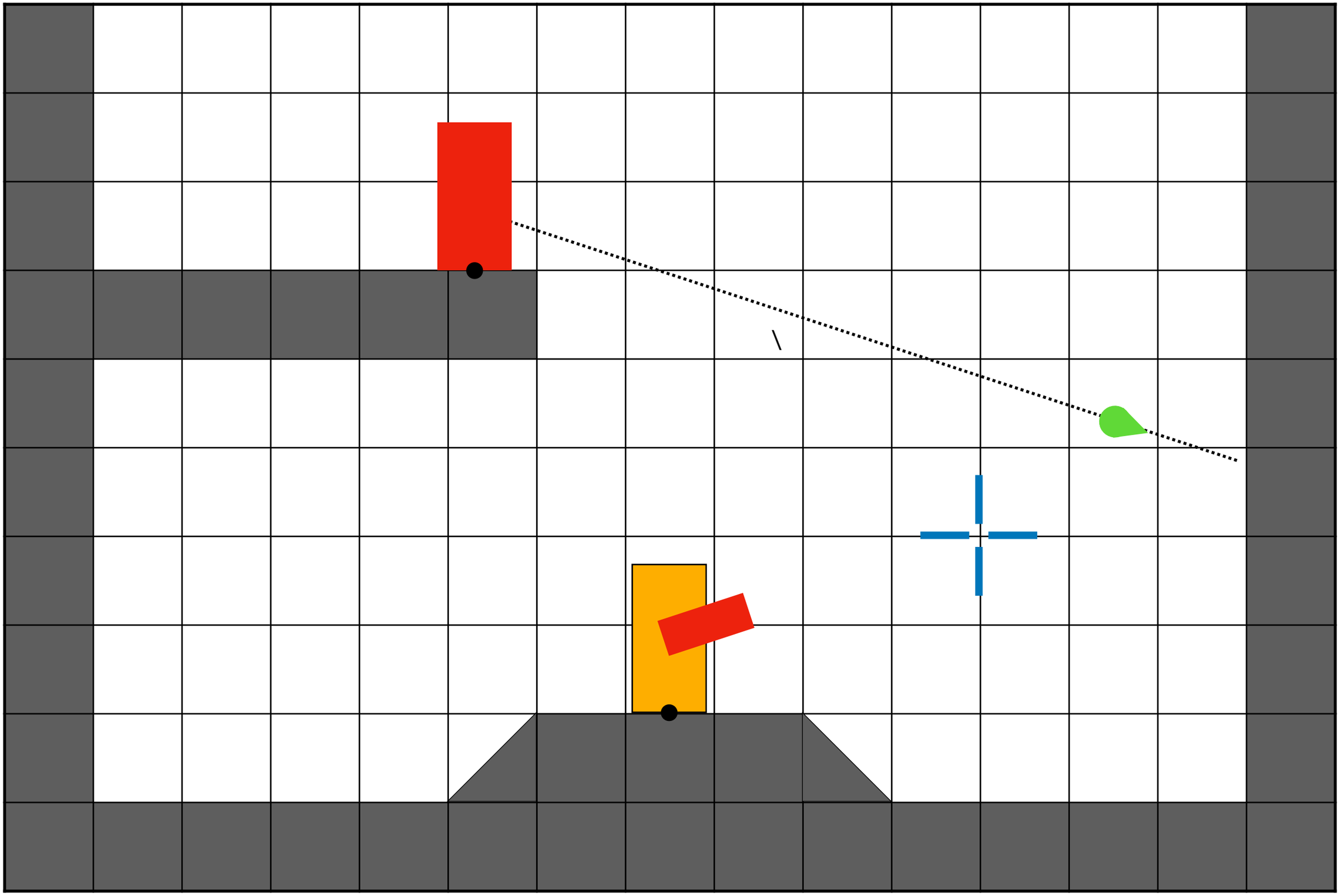


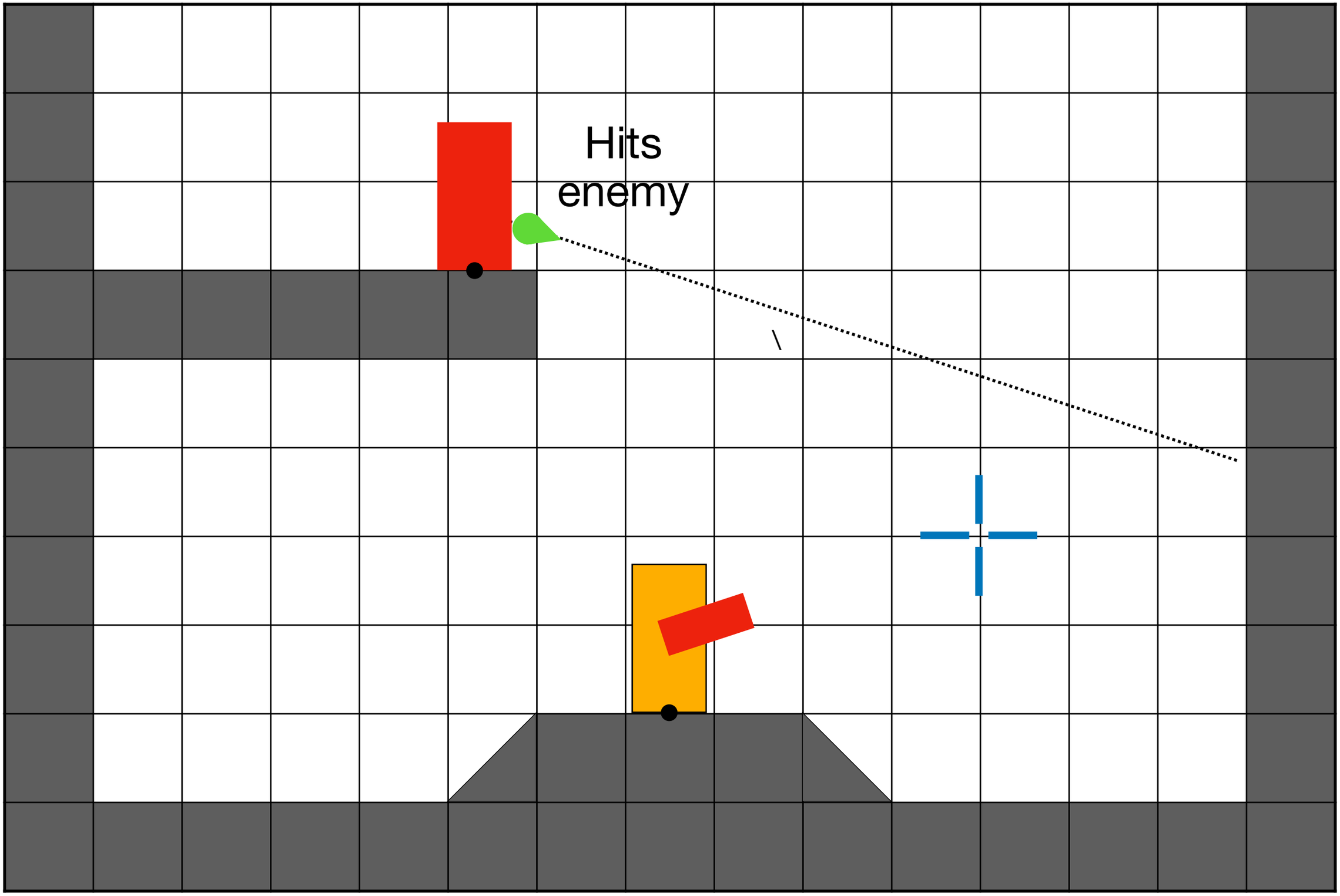

crosshair
follows mouse
Crosshair
Gun follows
crosshair
Laser

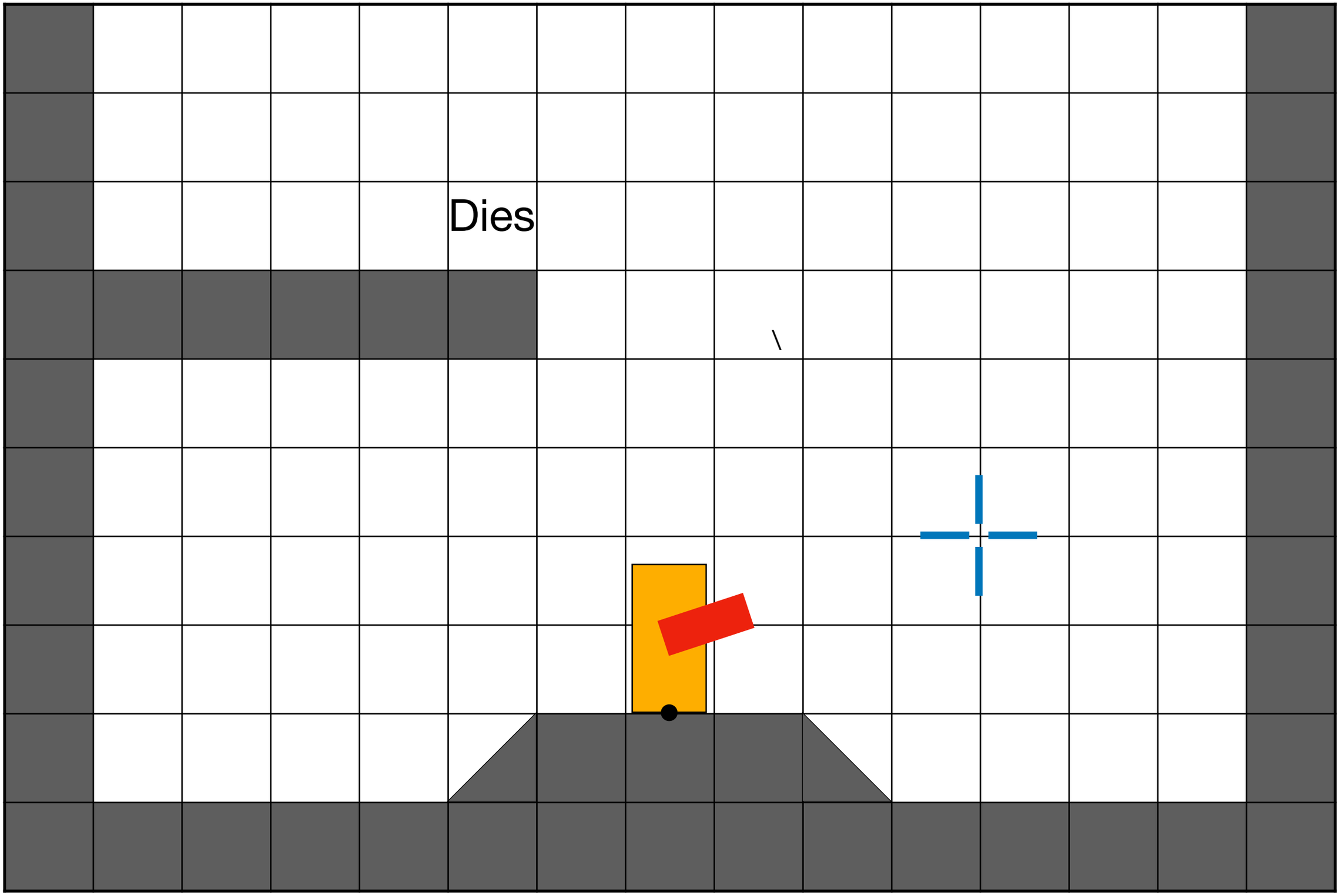

Gun follows
crosshair

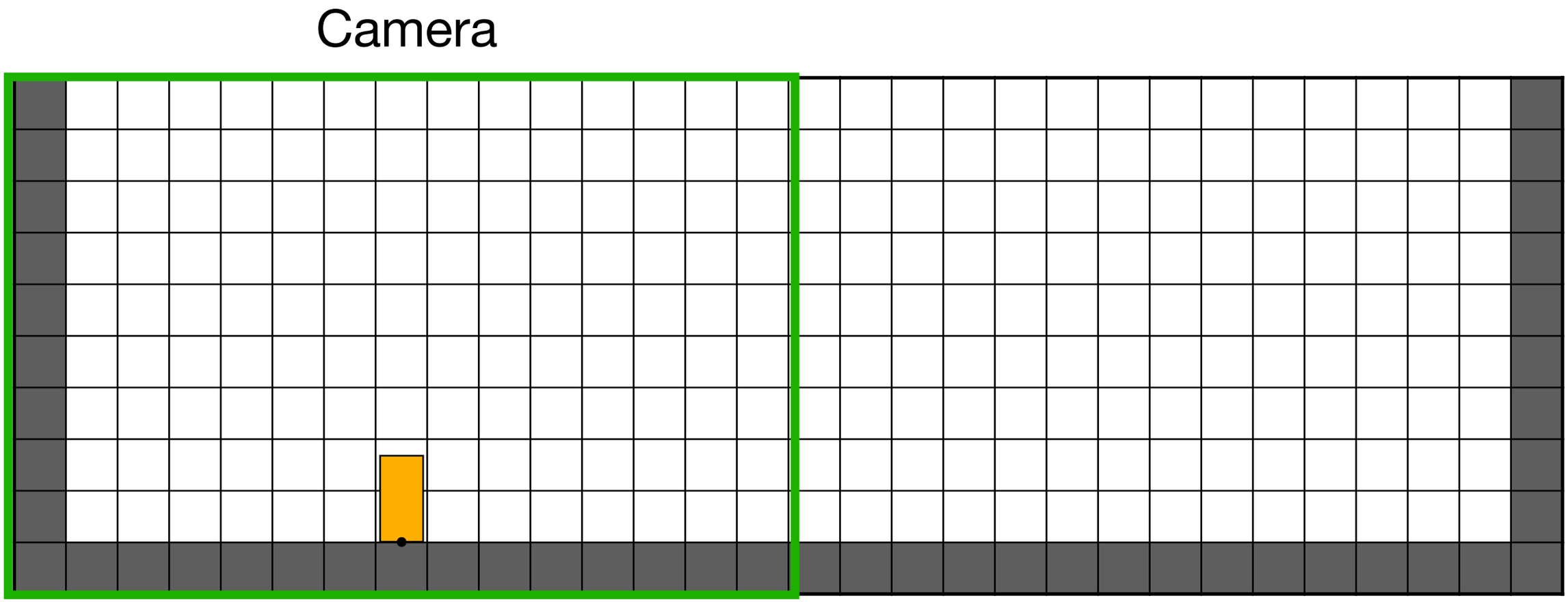

\
Click to
fire

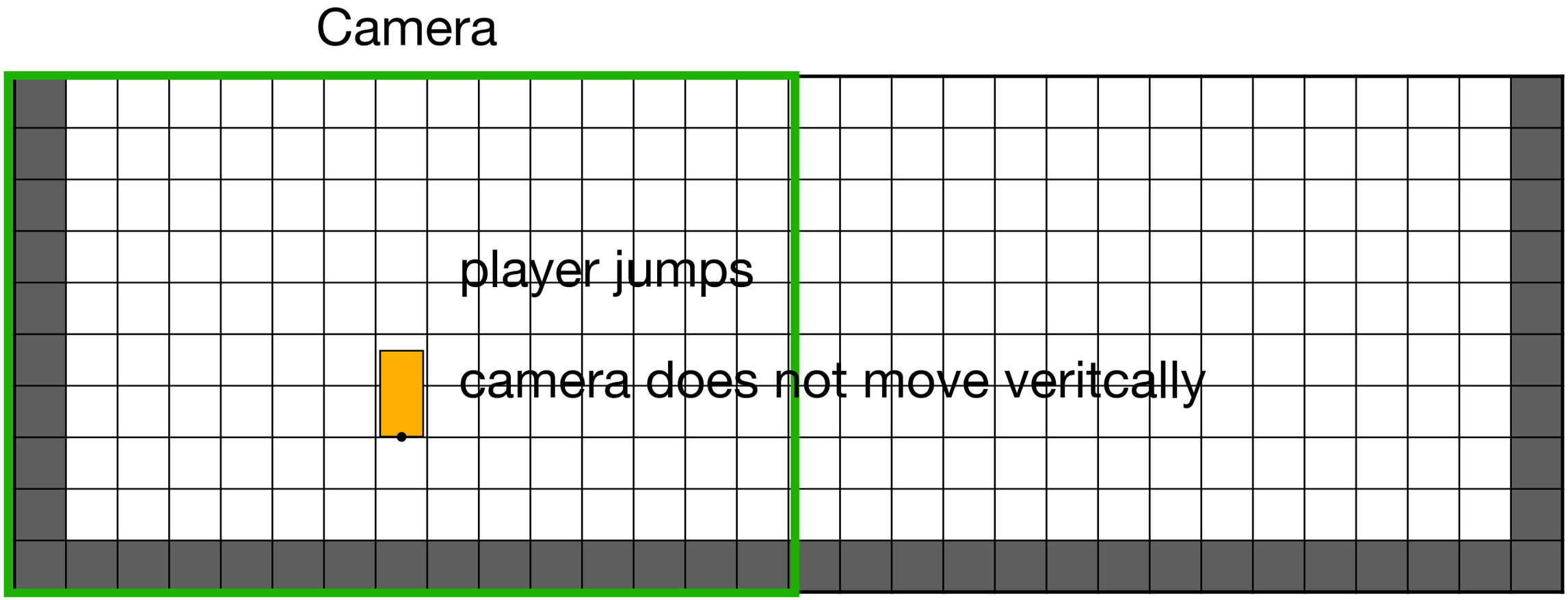

Bounces

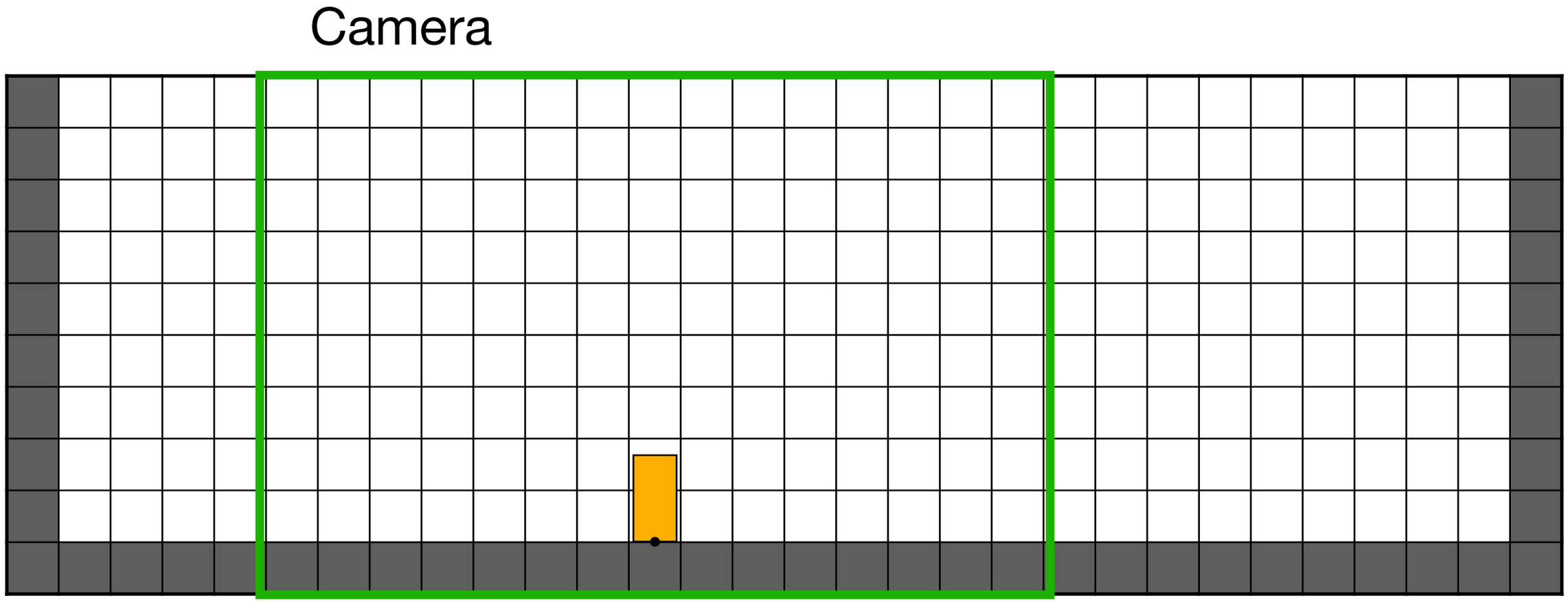



Hits
enemy


Dies
\

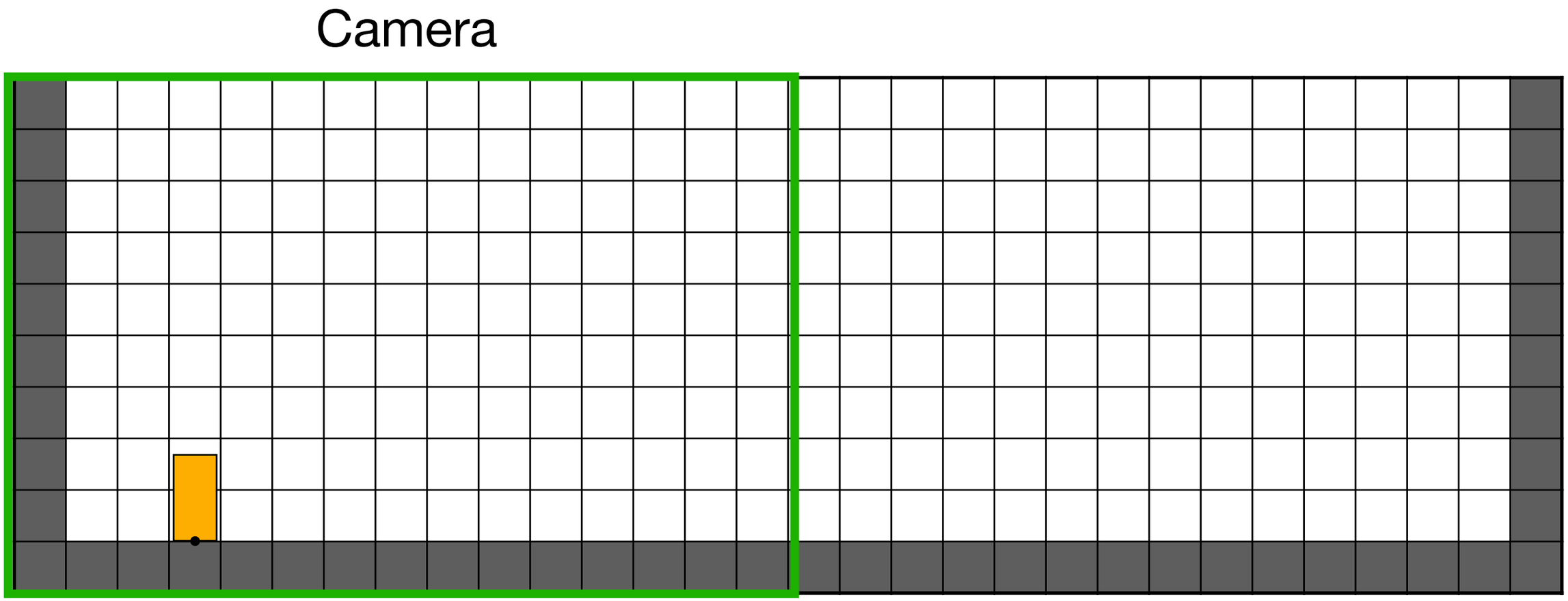

Camera


Camera
player jumps
camera does not move veritcally

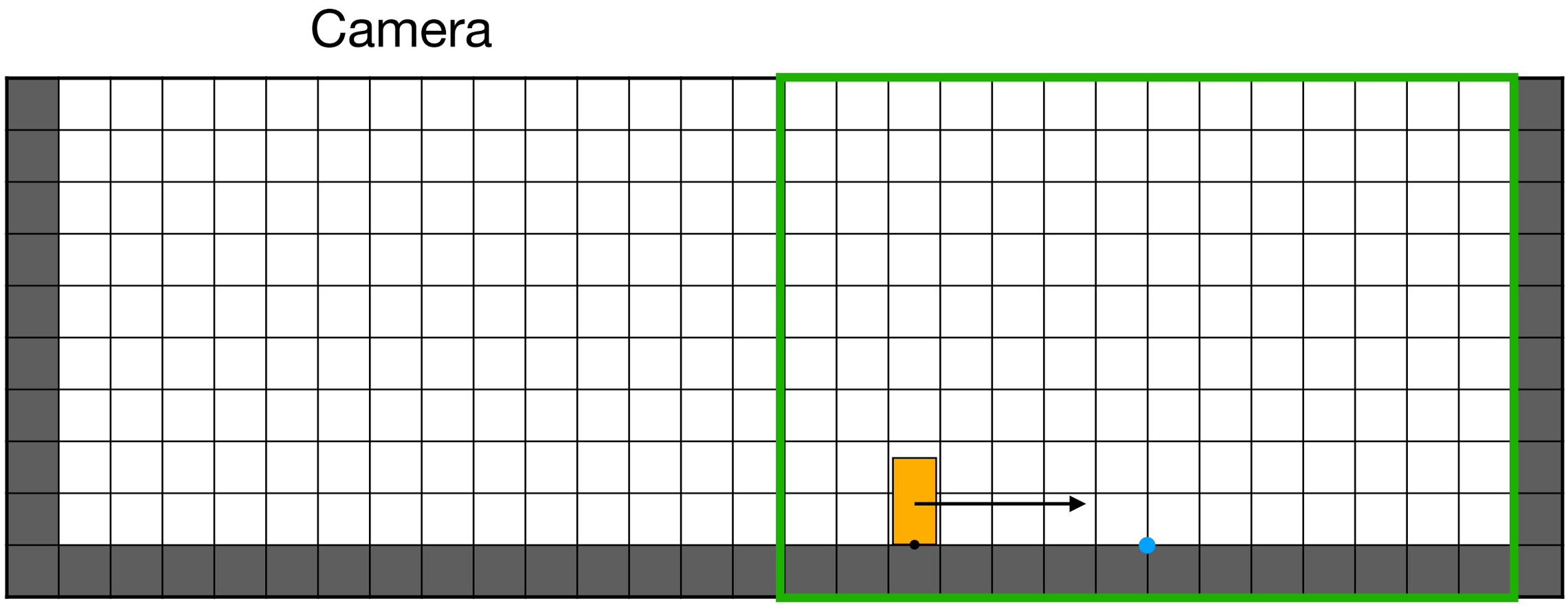


Camera moves to keep player at centre

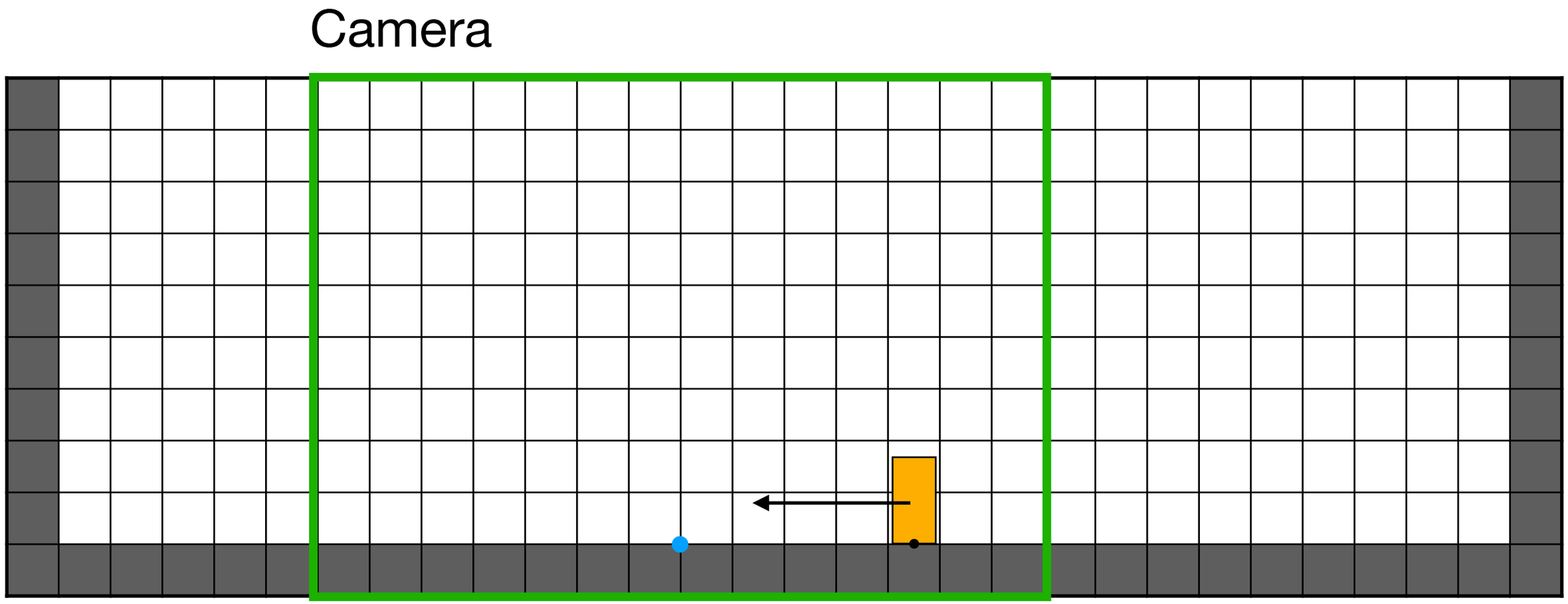
horizontally



Camera stops at edge of level



Camera moves in front of player



Camera moves in front of player

World

Follower

On LateUpdate:
* Move camera to follow player

get velocity

PlayerMove

Properties:
* Velocity

On Update:
* Read controls

On FixedUpdate:
* Move player

collides

CoinCollect

On player enters trigger collider:
* Destroy coin
* Update GameManager

coin collected
message

GameManager
(Singleton)

State:
* Coins collected

Properties:
* Coins collected

On Coin Collected:
* Increment coins collected
* Send CoinCollected event

get number of coins

subscribe to
coin collection events

UI

UIManager

On Coin Collected:
* Update coins UI