A11Y IRL

* accessibility in real life.

Hi.

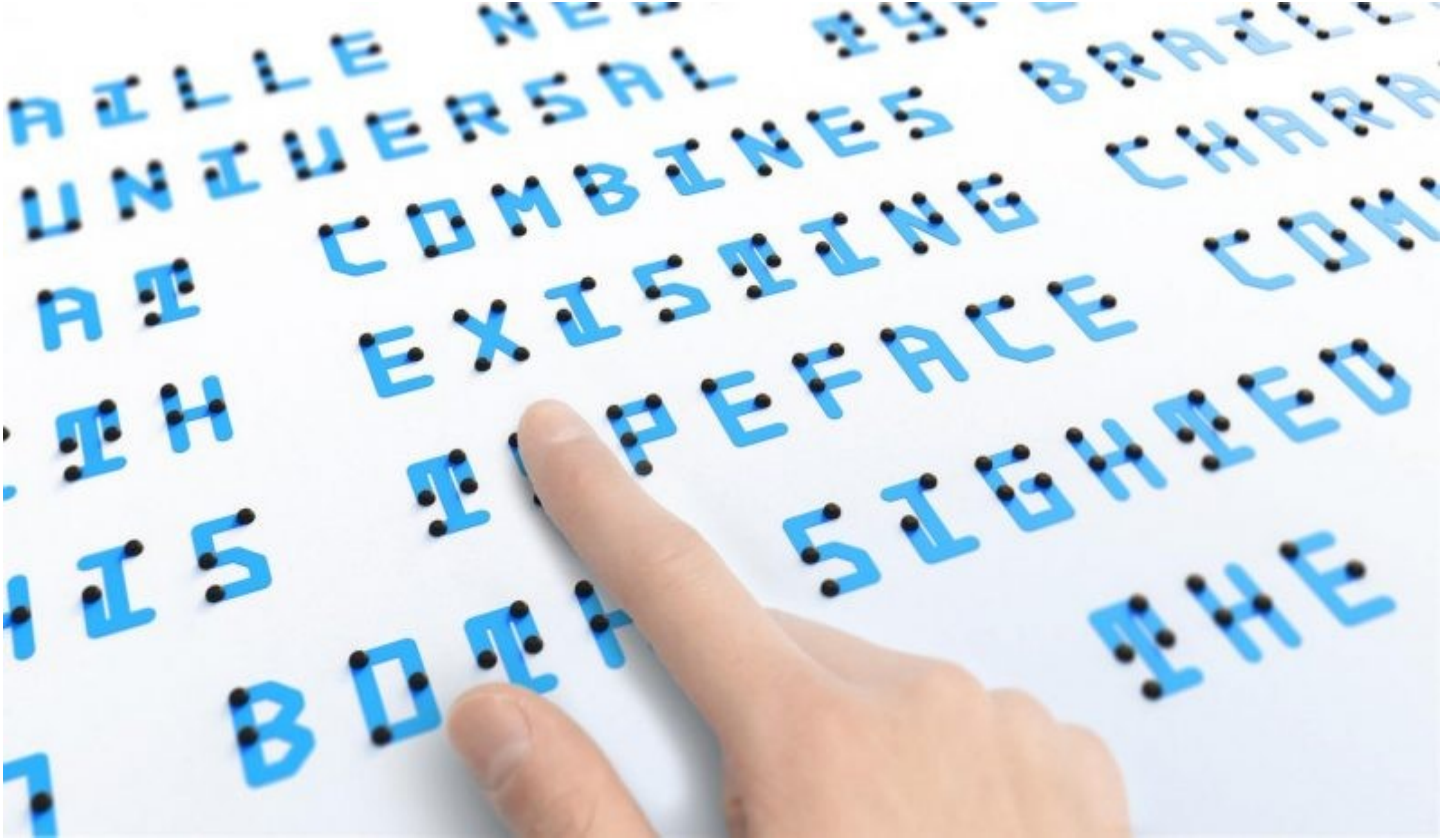
I am @vorillaz.
Software Engineer at Workable.

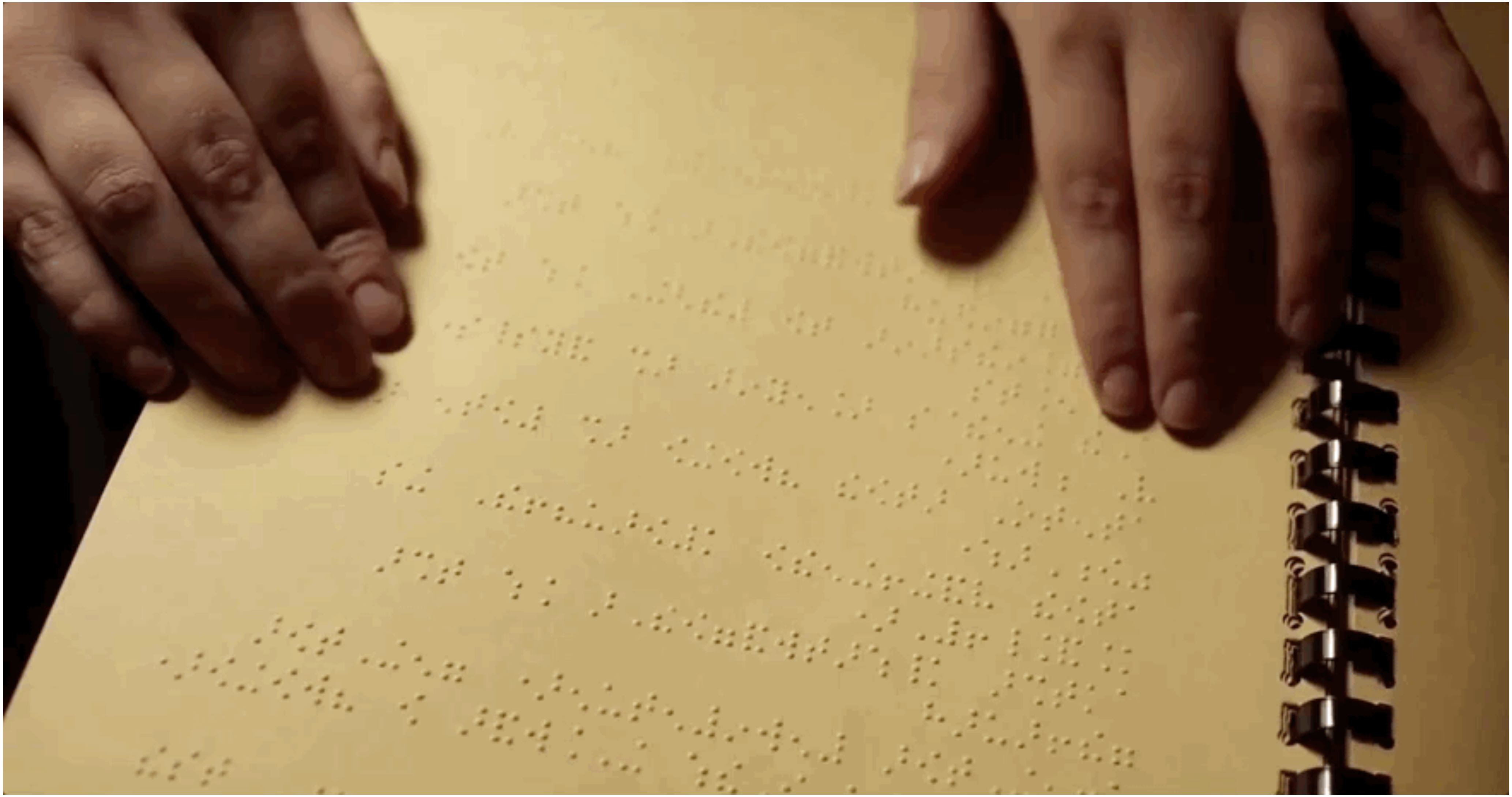
The community evolves.

> Accessibility refers to the design of products, devices, services, or environments for people who experience disabilities.

~200 years old ideas

Louis Braille


AILLE NE
UNIVERSAL
BRAILLE
COMBINES
CHARA
AT
EXISTING
COM
TH
TYPEFACE
IS
SIGHTED
BOTH
THE



Books

Typewriters

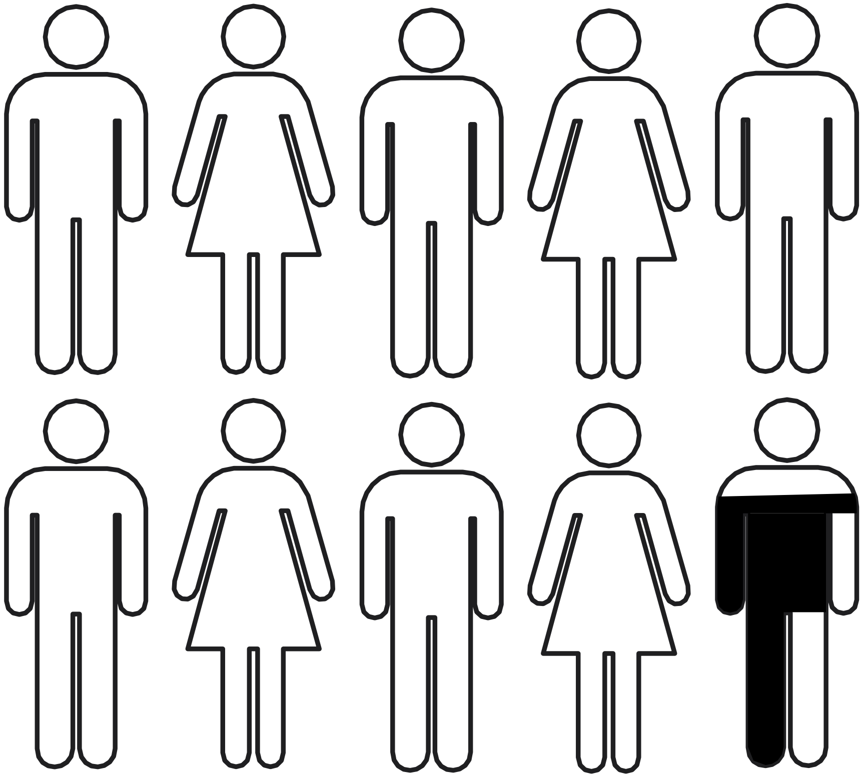
Refreshable displays

Right?

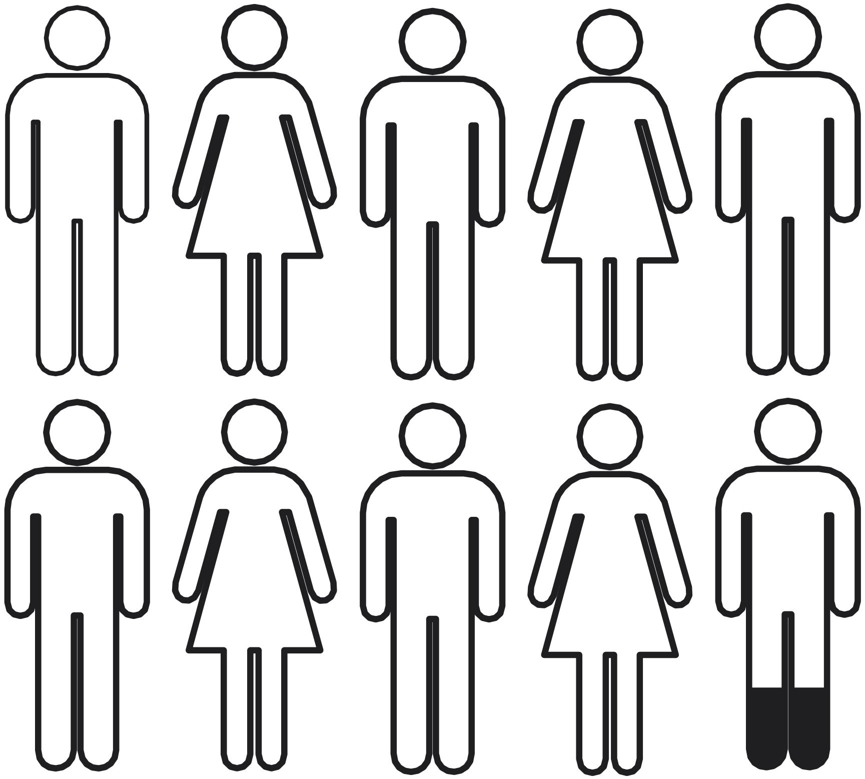
Wrong!

253 million people live with vision impairment.

🇺🇸 8.5% Braille readers



🌍 ~1% Braille readers



Literacy rate in 🇪🇹

49.1%

But why?

Super expensive books

Only 1% of books are available in Braille.

Super expensive Braille displays

$500 to $8,000

Expensive typewriters

~ $800

Can we help?

Ok, let's create a device.

Goals:

Easy to build & hack.

Accessible.

Cheap.

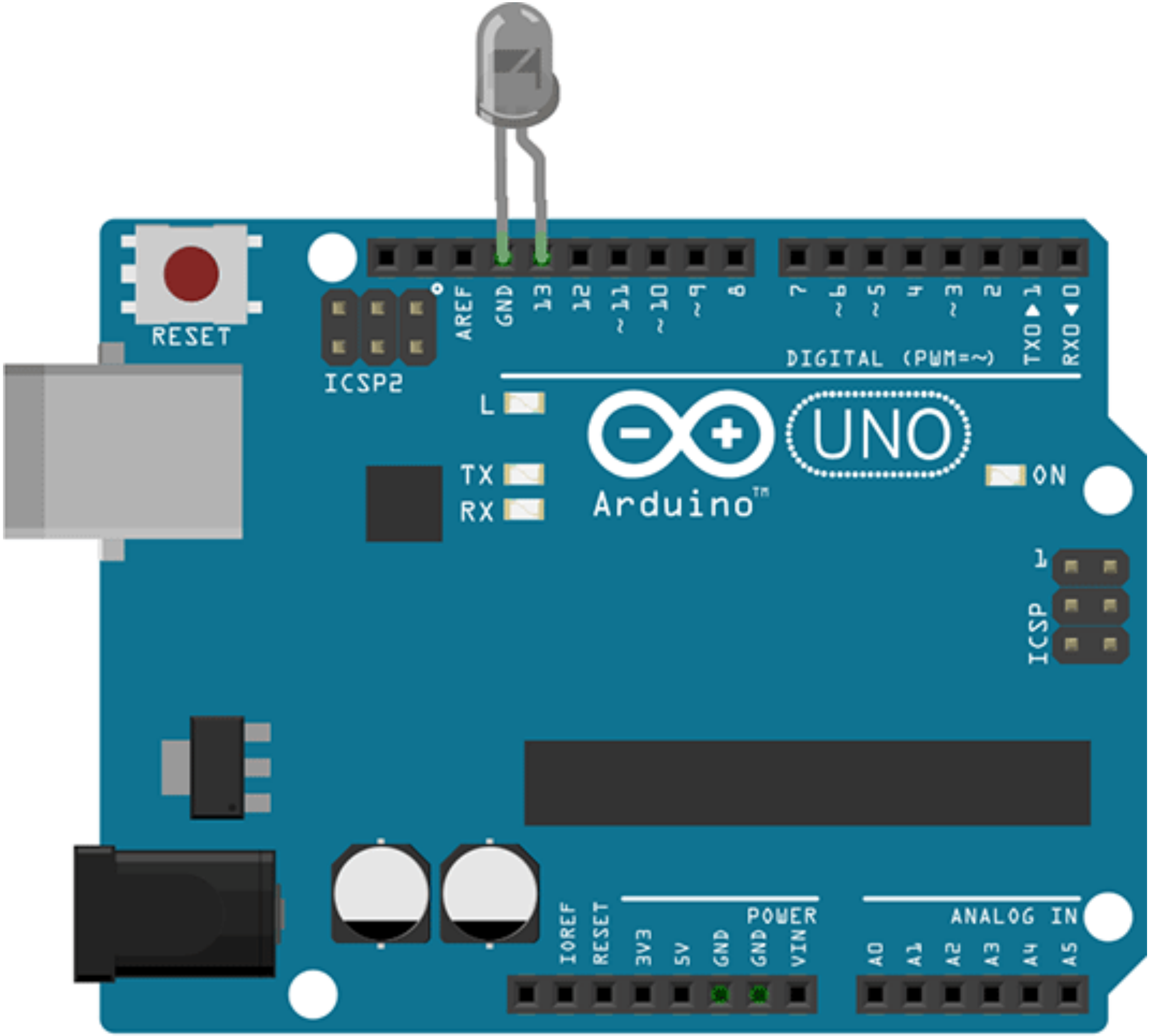
But how?

Microcontrollers

Arduino

Raspberry Pi

Photon

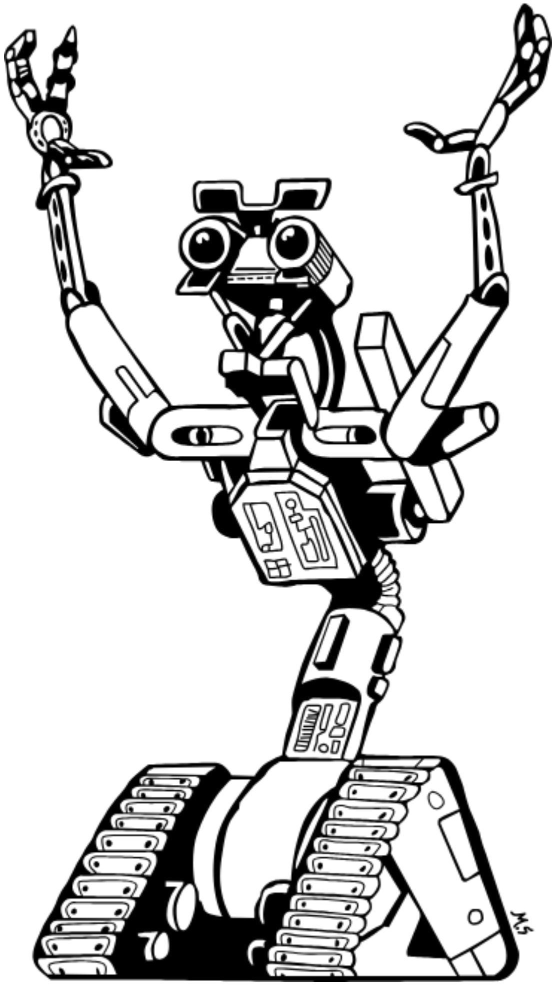

RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX▶1
RX◀0
DIGITAL (PWM=~)
L
TX
RX
UNO
Arduino™
ON
1
ICSP
IOREF
RESET
3V3
5V
GND
GND
VIN
POWER
ANALOG IN
A0
A1
A2
A3
A4
A5

How to code this?

```
int ledPin = 13;
void setup() {
 pinMode(ledPin, OUTPUT);
}

void loop() {
 digitalWrite(ledPin, HIGH);
 delay(500);
 digitalWrite(ledPin, LOW);
 delay(500);
}
```

Johnny Five.js



```
const five = require('johnny-five');
const board = new five.Board();

board.on(‘ready',() ⇒ {
 const led =  new  five.Led(13);

 led.blink(500, () ⇒ {

 console.log('Blink callback!’);

 });
});
```

JavaScript? Meh.

JavaScript feels so natural.

Event driven programming

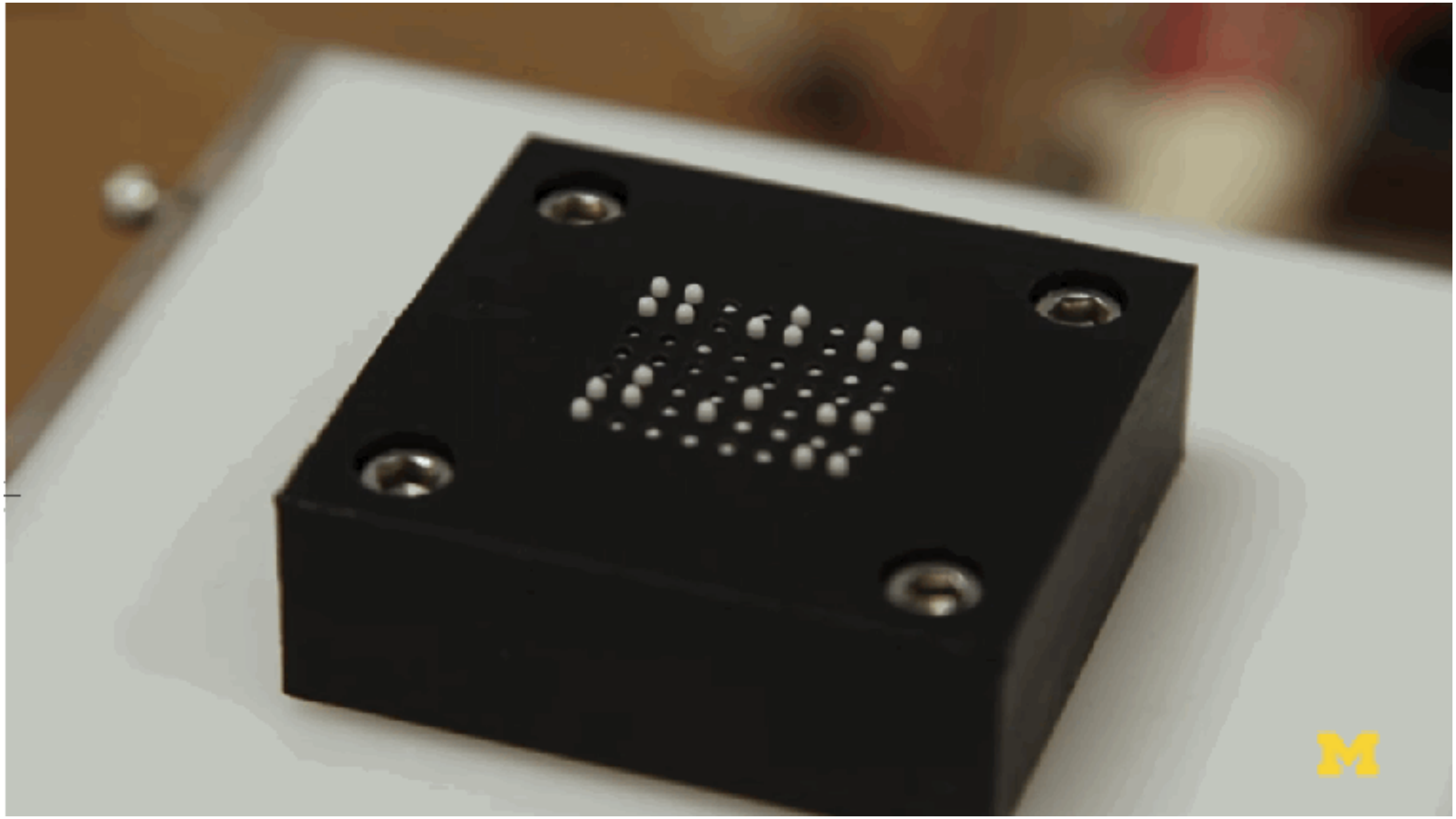
Use the Node.js ecosystem.

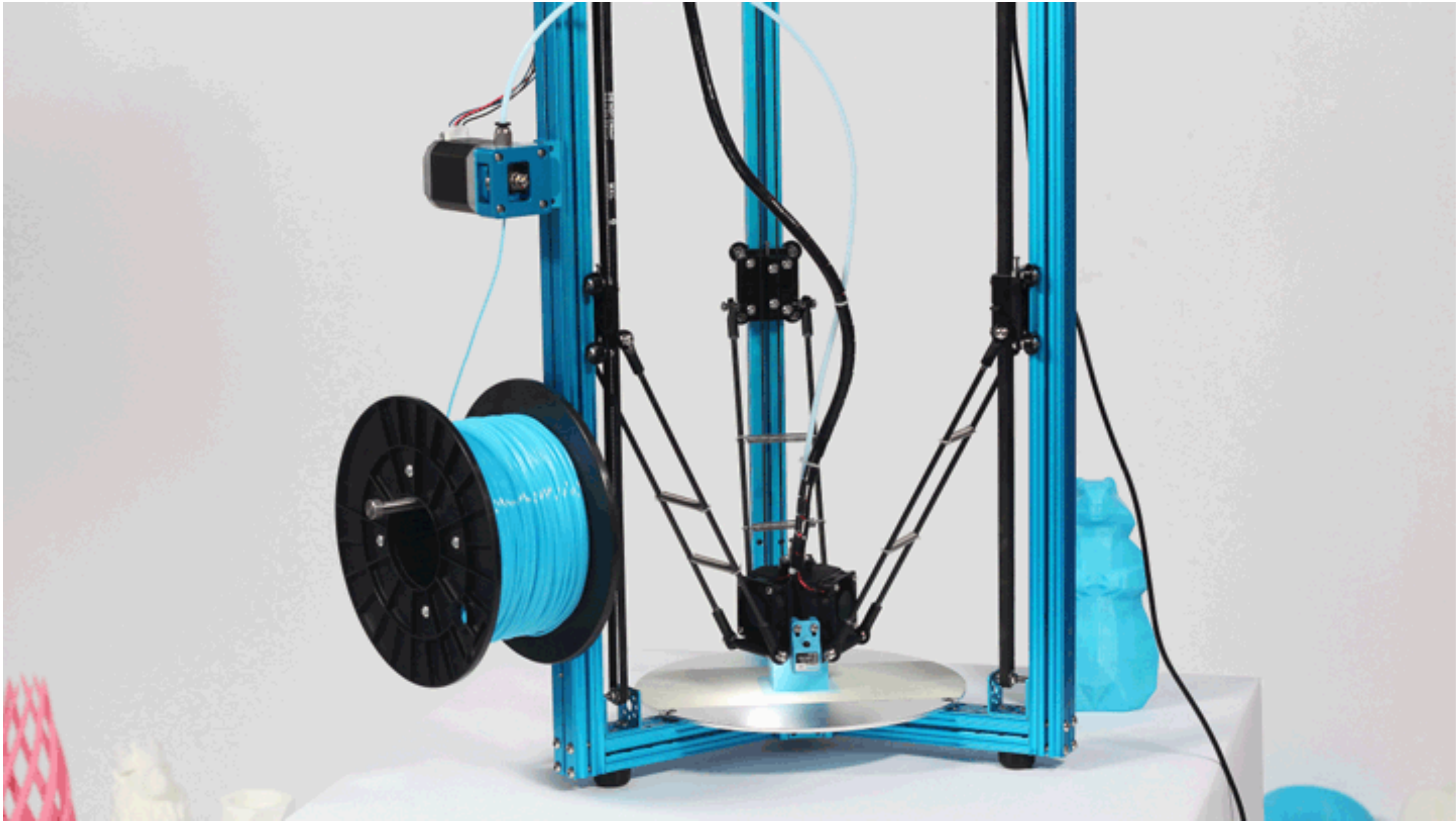
Portability.

Run the same code with different devices.

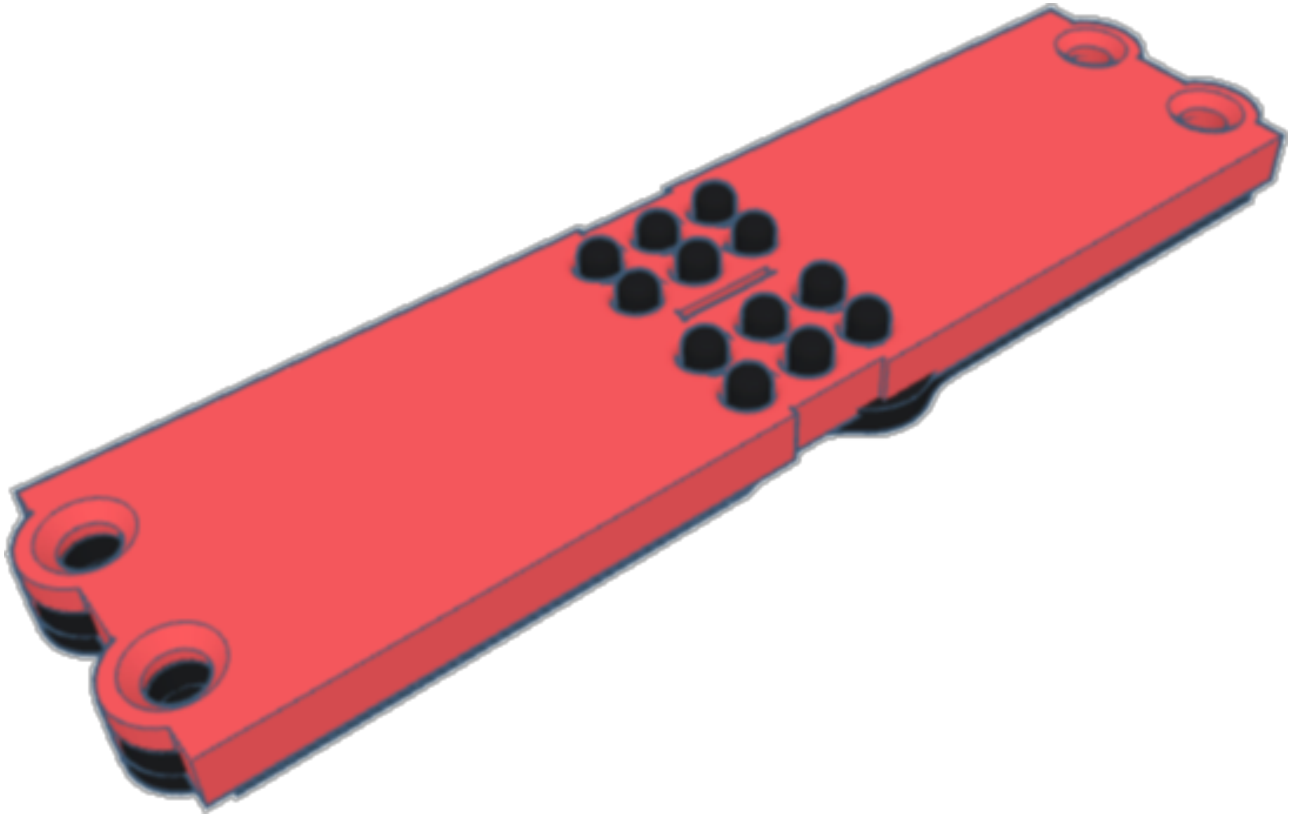
It's the community's mindset.

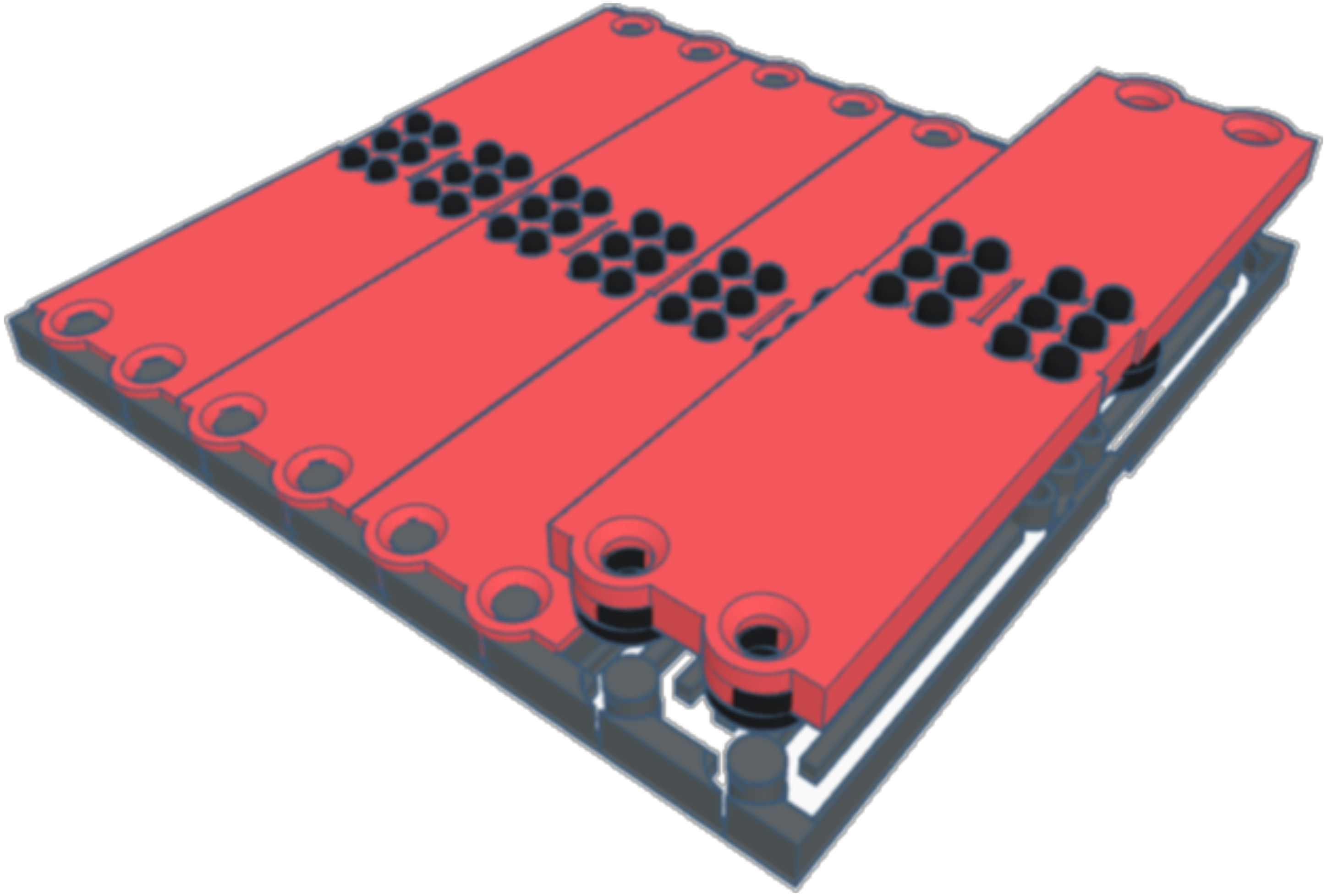
Prototype



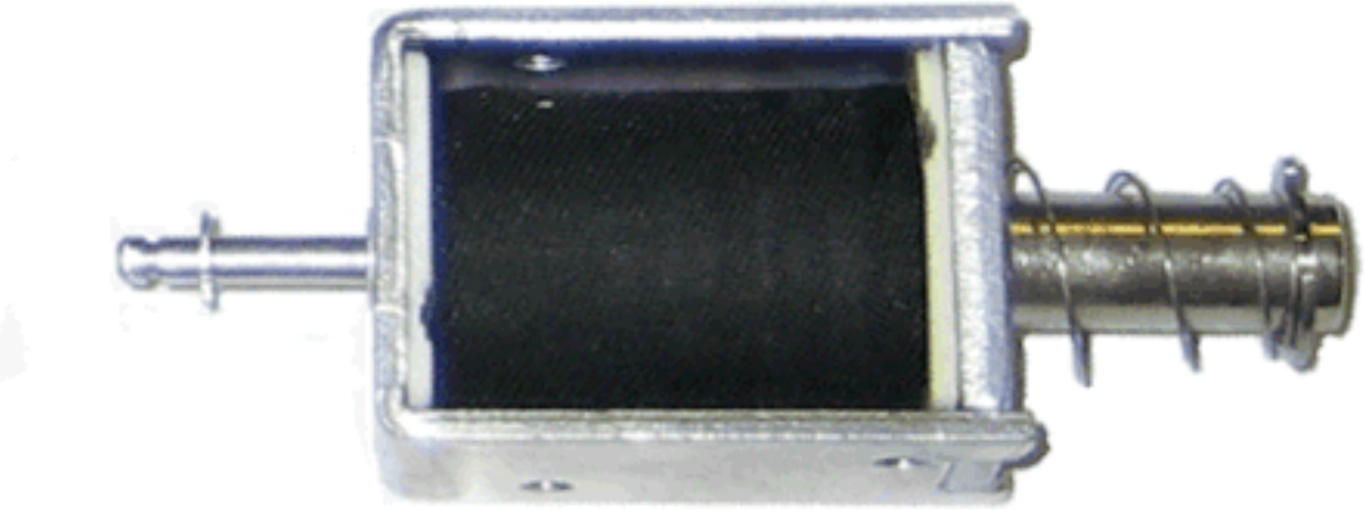


Modular cells





How to pop a pin?



All together now.

Text to Braille to Motion

A	⠁	[1,0,0,0,0,0]
B	⠃	[1,0,0,0,0,0]
Y	⠽	[1,1,0,1,1,1]
Z	⠵	[1,0,0,1,1,1]

```
const brailleMap = {
  a: [1,0,0,0,0,0],
  b: [1,0,1,0,0,0],
  // ...
};

const toBraille = c => {
  return brailleMap[c];
};
```

```
class Cell {
  constructor(five, pins) {
    this.solenoids = pins.map(p => {   return
      new five.Pin(p);
    });
  }

  draw(char) {
    this.solenoids.forEach((s, i) => {
      if (char[i] === 0) {
        s.low();
      } else {   s.high();
      }
    });
  }
}
```

```
const j5 = require('johnny-five');
const Cell = require('./cell');
const board = new j5.Board();

board.on('ready', function() {
 const char = new Cell(j5, {
  pins: [13,12,11,10,9,8]
 });
 char.draw('A');
});
```

Demo

Results

Easy to build.

Expandable.

Costs ~40€ for 12 cells.

Ok. What's next?

V2

Better design & feedback

Smaller with micro steppers

Cheaper

Try different Arduino modules

I had a dream.

I need you.

https://github.com/braillejs

⠞⠓⠁⠝⠅⠎

Thanks.

Ask me two questions

Links:

- World Health Organization, Global Data on Visual Impairments
- Braille Literacy Statistics