

# Hack the Mic: IoT with Kinoma

Lizzie Prader  
lizzie@kinoma.com





# The Internet of Things (IoT)



## Most Recent Customer Reviews

★★★★★ **Essential in life**

I used to walk around asking myself, "How many eggs are in the fridge?" I no longer have to worry, it is always at my fingertips.

Published 4 days ago by JR

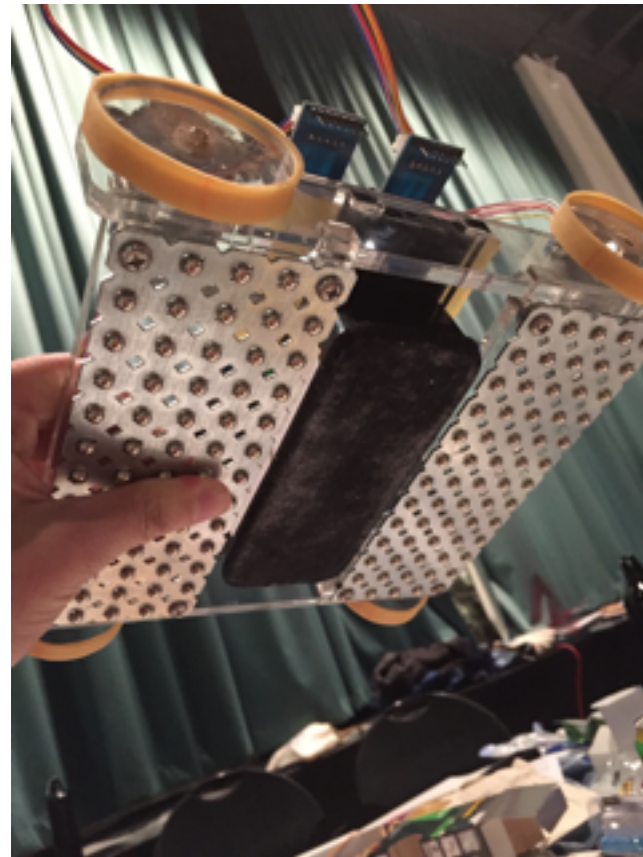
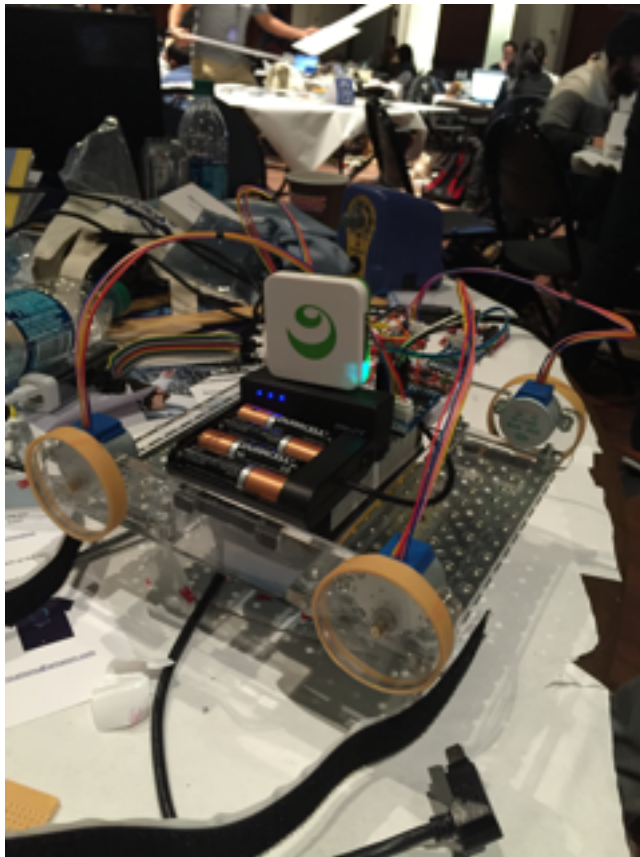




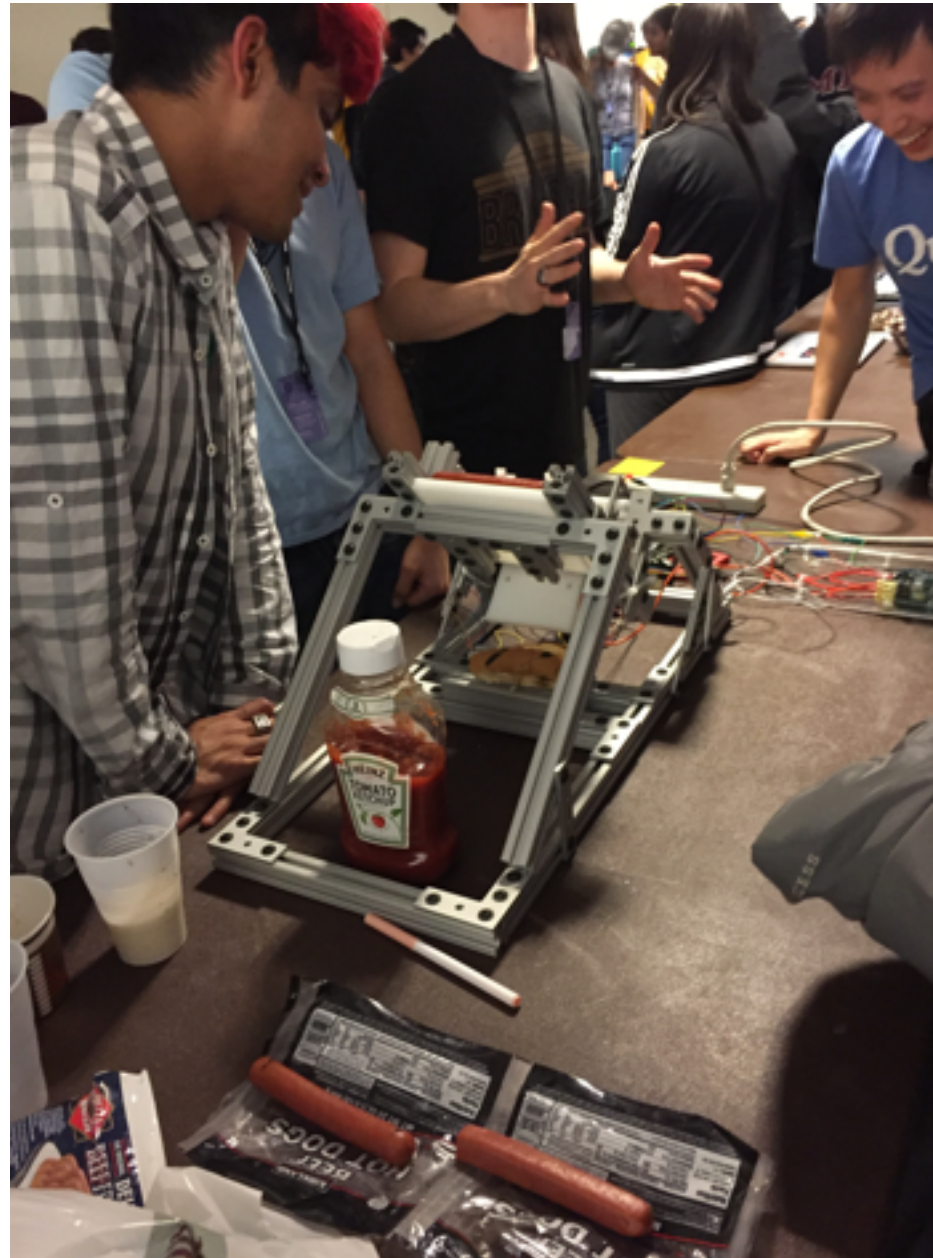
“Never lose your page again”



# MakeMIT



# Making without a cause





# The Maker Movement

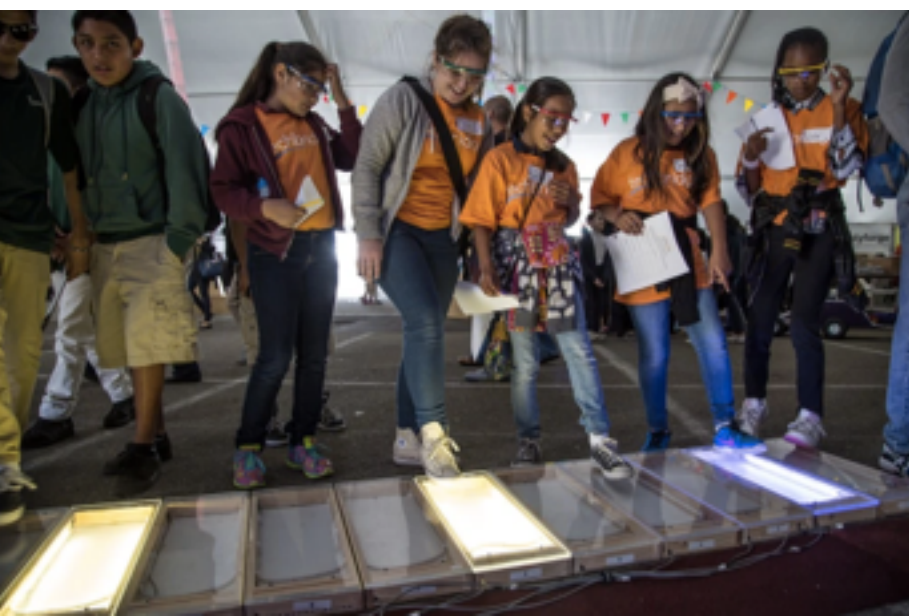
- DIY meets tech
- Computers, robotics, 3-D printing, metalworking, woodworking, etc.

# Maker media

**Make:** Make Magazine



Maker Faires

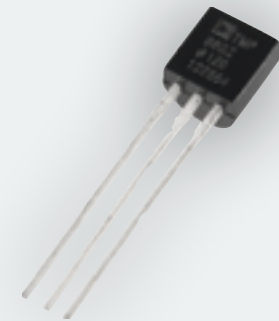
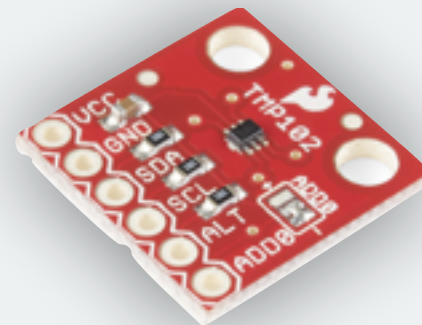
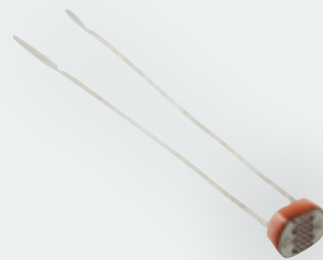
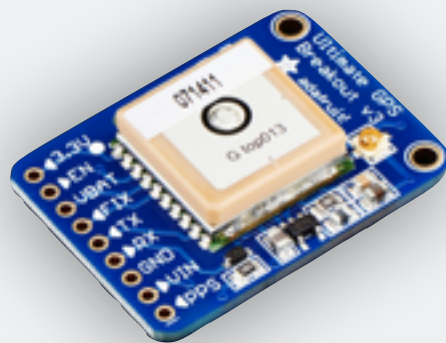
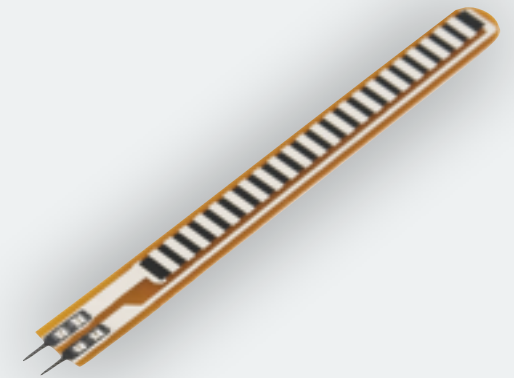
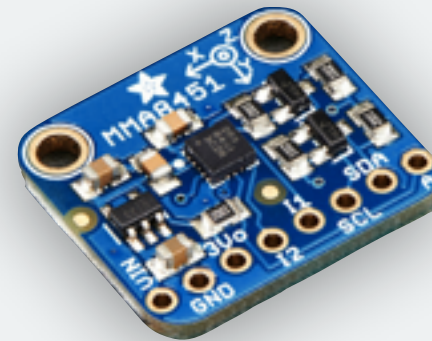
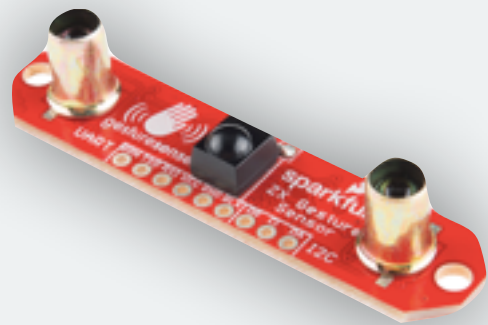




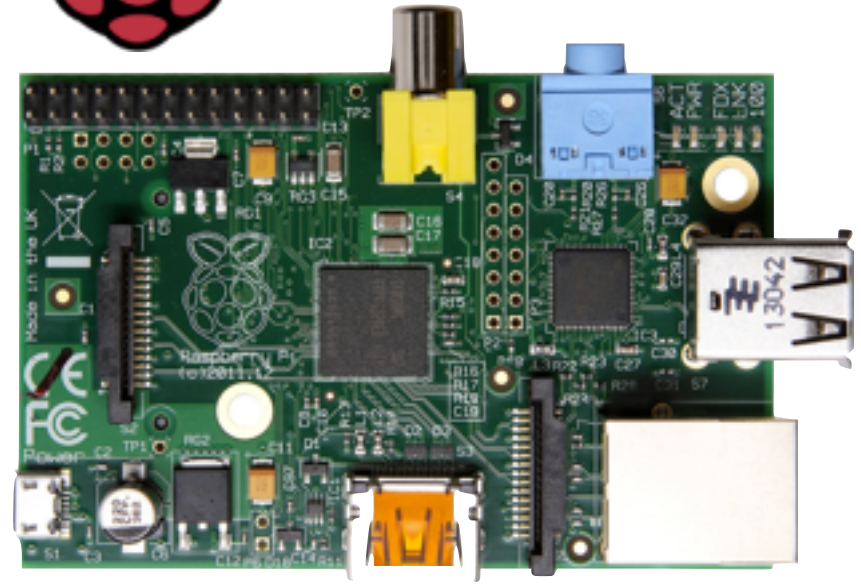
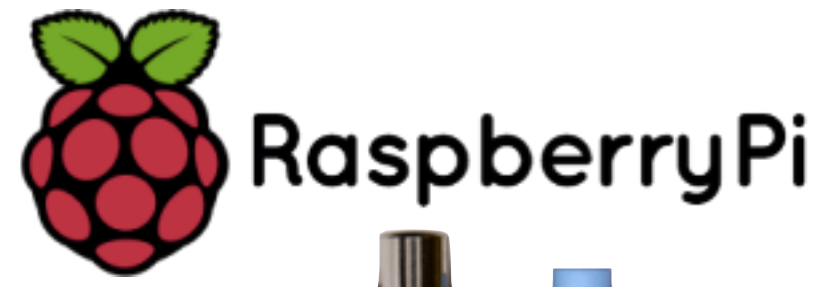
# Noisebridge Hackerspace

- Beginner friendly events with many materials provided
- [www.meetup.com/noisebridge](http://www.meetup.com/noisebridge)

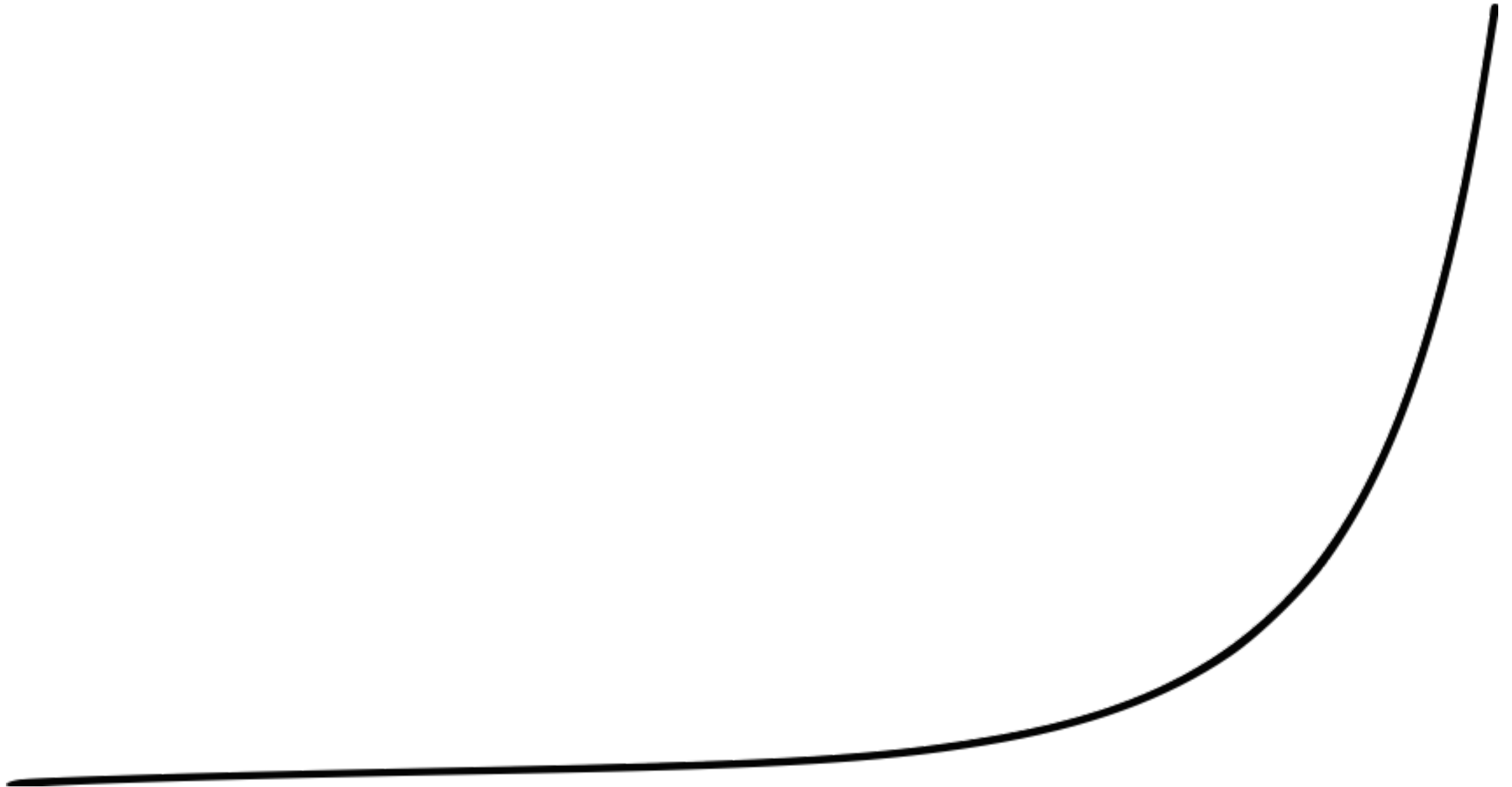








Getting started is hard







# Kinoma Element and Kinoma Create



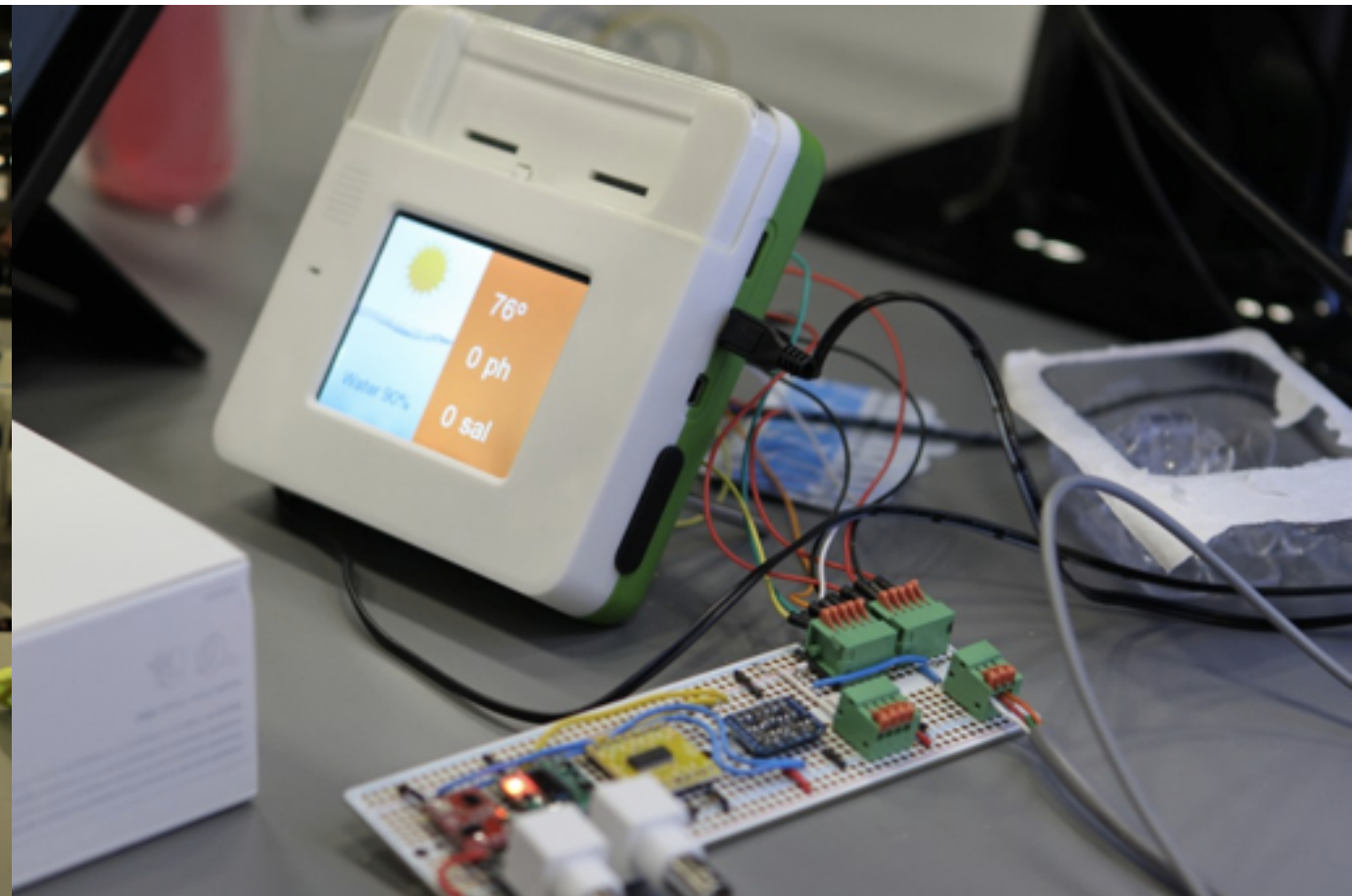
# Not your average prototyping boards





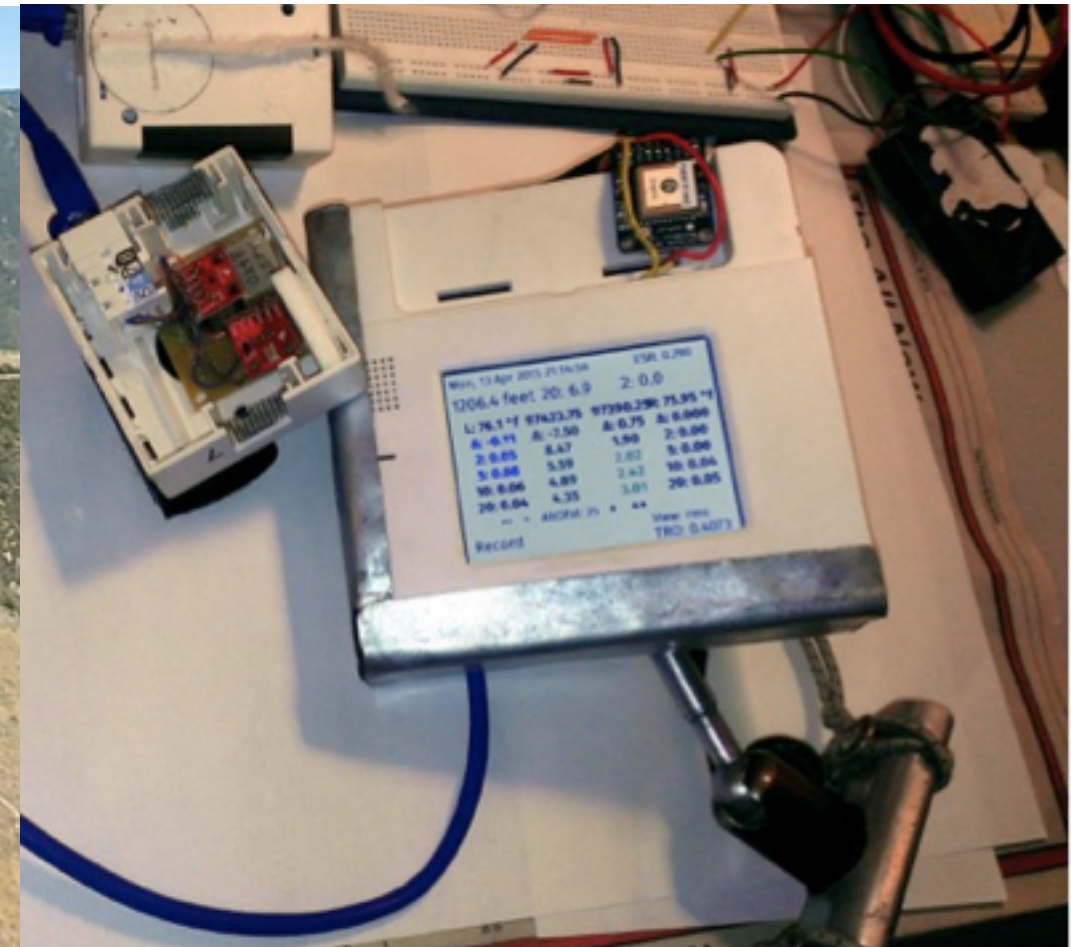
# Aquarium monitor

Kinoma



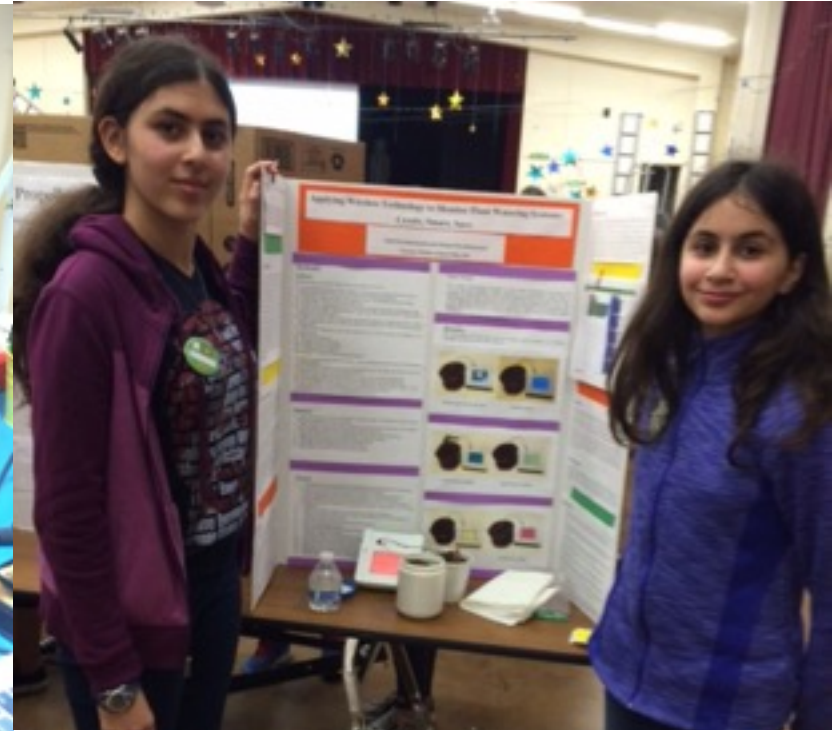
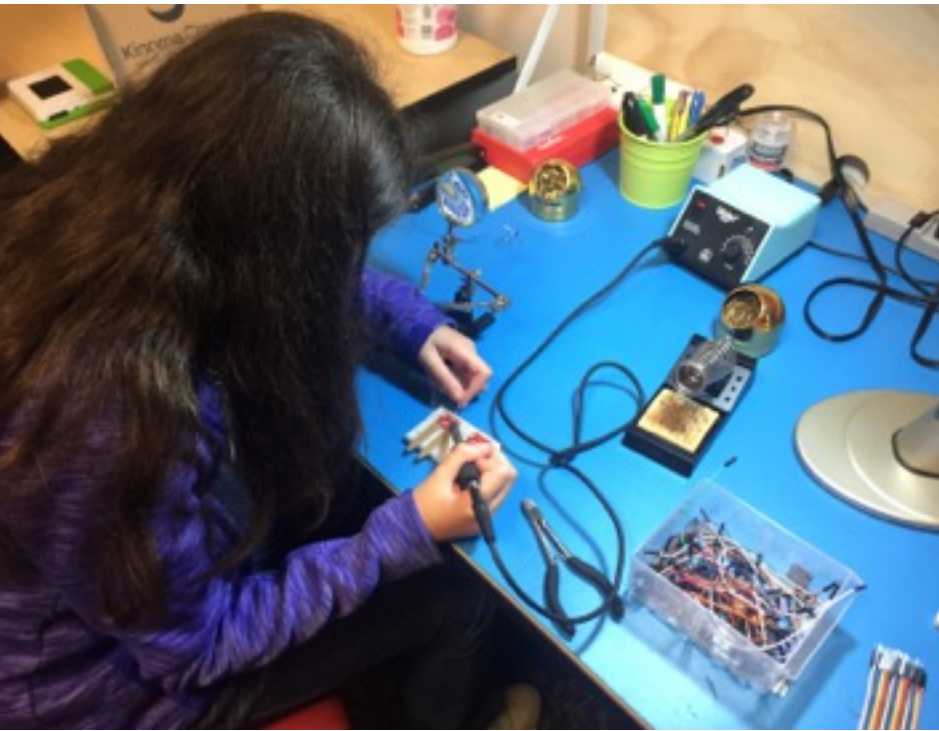


# Hang gliding helper





# Plant monitor







(video)

# Tri-color LED

# A note to new makers

- Never view projects at face value
- Prototyping kits let you decide what to make with them



# The Kinoma platform



- Prototyping devices
- Our own programming language
- Tools for developers to program them



# JavaScript has proven to be accessible to designers, students, and engineers

```
var redSkin = new Skin({ fill: 'red',});
var blueSkin = new Skin({ fill: 'blue',});

var labelStyle = new Style({ color: 'white', font: 'bold 36px', horizontal: 'center', vertical: 'middle', });

var MainContainer = Container.template($ => ({
  left: 0, right: 0, top: 0, bottom: 0,
  active: true, skin: blueSkin,
  contents: [
    Label($, { left: 0, right: 0, top: 0, bottom: 0, style: labelStyle, string: $.text })
  ],
  behavior: Behavior({
    onTouchBegan: function(container, id, x, y, ticks) {
      container.skin = redSkin;
    },
    onTouchEnded: function(container, id, x, y, ticks) {
      container.skin = blueSkin;
    }
  })
}));

application.add( new MainContainer({ text: "Hello, KPR" }));
```

# 4 hardware trends to watch in 2016

Early signals of what's to come in the hardware world.



**O'REILLY®**

## **High level programming languages on embedded systems**

Relatedly, writing software to control drones, vending machines, and dishwashers has become as easy as spinning up a website. Fast, efficient processors ... are turning JavaScript into a popular embedded programming language—unthinkable less than a decade ago.



# Kinoma Studio

The screenshot displays the Kinoma Studio IDE interface. The title bar shows the file path: `Kinoma Edit - hello/src/main.js - Kinoma Studio - /Users/lprader/Documents/Kinoma Studio`. The interface includes a Project Explorer on the left, a code editor in the center, and a Console/Debug area at the bottom.

**Project Explorer:** Shows a project structure with a folder named `hello` containing `application.xml` and a sub-folder `src` containing `main.js`. Other projects listed include `i2c-color-sensor`, `i2c-monster-mayhem`, `i2c-nfc`, `i2c-temperature`, `i2c-temperature`, `i2c-touchshield`, `k5_example`, `k5_pubnub_light`, `Kangaroo Disco`, `kangaroo-disco`, `kangaroo-text`, `license-simple`, `light-share-create`, `light-share-element`, `LightsOn-client`, `LightsOn-server`, `logs`, `media-player`, and `MonsterDisplay [kinoma-stuff master]`.

**Code Editor:** The `main.js` file is open, showing the following JavaScript code:

```
//@program
var redSkin = new Skin({ fill: 'red',});
var blueSkin = new Skin({ fill: 'blue',});

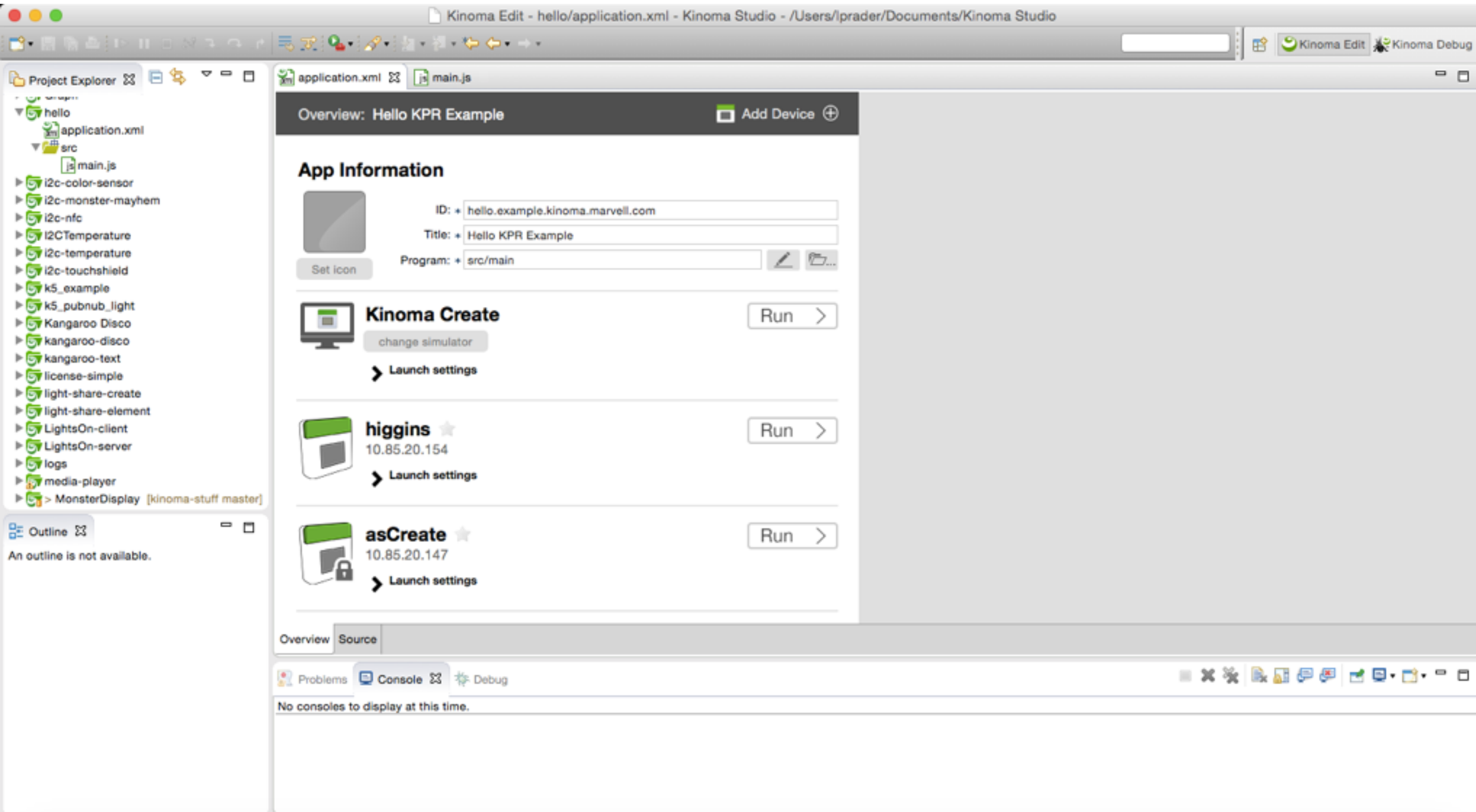
var labelStyle = new Style({ color: 'white', font: 'bold 36px', horizontal: 'center', vertical: 'middle', });

var MainContainer = Container.template($ => ({
  left: 0, right: 0, top: 0, bottom: 0,
  active: true, skin: blueSkin,
  contents: [
    Label($, { left: 0, right: 0, top: 0, bottom: 0, style: labelStyle, string: $.text })
  ],
  behavior: Behavior({
    onTouchBegan: function(container, id, x, y, ticks) {
      container.skin = redSkin;
    },
    onTouchEnded: function(container, id, x, y, ticks) {
      container.skin = blueSkin;
    }
  })
}));

application.add( new MainContainer({ text: "Hello, KPR" }));
```

**Console/Debug:** The `Console` tab is active, displaying the message: "No consoles to display at this time."

# Kinoma Studio



# KinomaJS Blocks

The image shows the KinomaJS Blocks editor interface. The top bar includes the KinomaJS logo, a search icon, a "View JavaScript Source" button, a "blocklyApp" field, a "target IP" field, and a "RUN >" button. The left sidebar contains a category menu with "Blockly" and "KinomaJS" sections. The "Blockly" section includes Logic, Loops, Math, Text, Lists, Colour, Variables, and Functions. The "KinomaJS" section includes UI, Pins, Time, Sound, and Advanced. The main workspace is a grid where blocks are assembled. The "screen" block is titled "Screen1" and has a "default colour" block. The "onCreate" block contains several UI elements: a "label" with text "Last Received Message:", a "label" with text "(nothing received)", a "rectangle" with a blue background, a "label" with text "Send:", a "button" with text "Hi!", a "button" with text ":", a "button" with text ":D", and a "button" with text "WOW". Each button has an "onTouch" and "offTouch" block. The "offTouch" blocks for the buttons call "sendMessage with: message" with the button's text. The "sendMessage" block is defined in a "server" block as "call receiveMessage with: x at PARTNER\_IP\_ADDRESS". The "receiveMessage" block is defined in a "server" block as "write light with 1", "set receivedMessageLabel text to message", "timeout timer on 1000 ms", and "onTimeout write light with 0". The "pins" block contains "onCreate" blocks for "make ground on pin 51 with name ground" and "make digital out on pin 52 with name light", and an "onReady" block.





(video)

# KinomaJS Blocks



- As easy as it gets
- No downloading
- No wires



+



+





# Everyone can be a maker





# Thank you!



kinoma.com



@kinoma



kinoma



kinomahq



kinoma



# Kinoma

# Questions?

 kinoma.com

 @kinoma

 kinoma

 kinomahq

 kinoma

