f3.js: A Parametric Design Tool for Physical Computing Devices for Both Interaction Designers and End-users

The presented system is publicly available at http://f3js.org
Physical computing devices, everywhere
Physical computing devices, everywhere

- Physical user interfaces (as opposed to GUI)
- Home appliances
- Smart devices
- Internet of Things
Personal fabrication made easy

3D print in “Maker” venues

Photo taken by Atsushi Tadokoro (CC BY 2.0)
https://www.flickr.com/photos/tadokoro/5138646645

Soldering with parents

Photo taken by Mitch Altman (CC BY-SA 2.0)
https://www.flickr.com/photos/maltman23/6954963529

Working prototypes

How about device programming & assembly?
Research questions regarding physical computing devices

For interaction designers
• How can we support prototyping of the devices?

For end users
• How can we support personal customization of the devices?
Preliminary observations:
photos of 200 devices and informal interviews
Design patterns in Physical User Interfaces

- 187 devices have physical user interfaces on planar surfaces
- 139 devices have modules placed along straight lines
- 51 devices have modules placed on circular paths
Mental gap between software & hardware

- Designers need to imagine hardware while writing code
- “new Button()” does not infer any hardware layout
Difficulties in exploring design alternatives

• Expensive **switching** cost between two activities in two tools
• Prior efforts in either one of these (software **or** hardware)
Preliminary observations (summary)

- Typical **design patterns** should have tool support
- **Mental gap** between software & hardware exists
- **Comparing alternatives** is crucial for good design
f3.js: integrated support for programmers

- Live Programming with intuitive APIs of features & layout
- Interactive development of IoT devices in one environment

```javascript
// draw arc
var thetaOffset = 25 / r;
var theta = ((Math.PI * 2) - (thetaOffset * 2 * numBuzzers)) / numBuzzers;
var n = 50;
for (var j = 0; j <= n; j++) {
  var t = thetaOffset + (theta / n) * j;
  pp.lineTo(
    Math.sin(t) * r,
    Math.cos(t) * r - r
  );
}
```
Module repository for hardware metrics
APIs for 3D extrusion and 2D layout

Extrusion (w/o opposite plane)

Partial extrusion (black lines ignored)

3D extrusion

Line layout
(align, padding: 25mm, wrap: true)

Line layout
(distribute, rotate: false)

Circle layout
(distribute, valign: top, clockwise: false)

2D layout

Circle layout
(distribute, size: false, rotate: false)
f3.js for parametric design of physical computing devices

- Typical **design patterns** should have tool support
- **Mental gap** between software & hardware exists
- Comparing **alternatives** is crucial for good design

☑ supported
☑ addressed
☑ supported
f3.js: customizing support for end-users

- **Interactive UIs** for customization
- Automatic generation of **device building instructions**
User studies

• 14 teams to create physical computing devices with f3.js
  • 5 interaction designers and 16 university students
  • Intel Edison and Grove modules, acrylic panels and screws provided

• 3 interaction designers and 3 end-users with revised f3.js
  • 3 interaction designers asked to create parametric designs
  • 3 end-users asked to customize and assemble devices
User studies: results & discussions

• Creativity support environments, not tools
• 3D vs 2D layout managers
• Interface builders are important
• Code-centric tool complements to 3D modeling tools
• Domain-specific language support (like HTML)
f3.js: A Parametric Design Tool for Physical Computing Devices for Both Interaction Designers and End-users

The presented system is publicly available at http://f3js.org