



**Francois**

[linkedin.com/in/francois-baldassari](https://www.linkedin.com/in/francois-baldassari)



**Heiko**

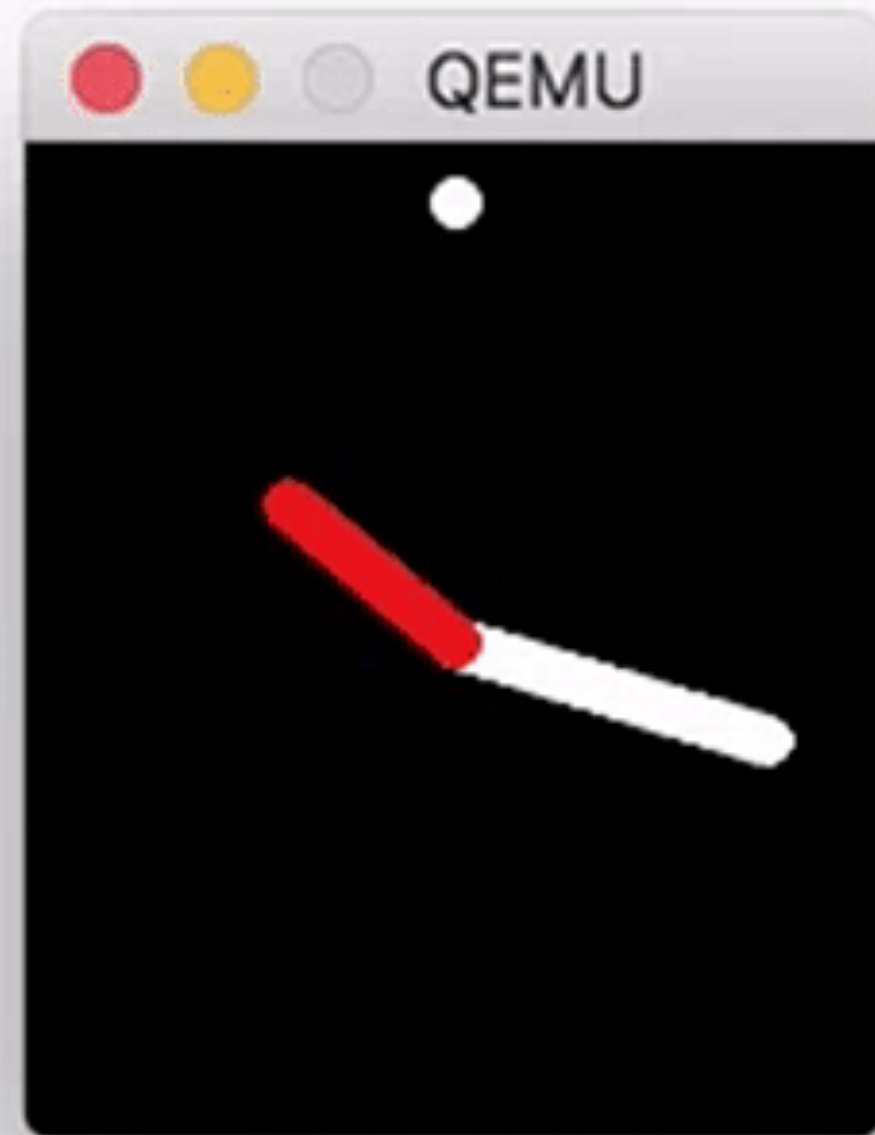
[linkedin.com/in/hbehrens](https://www.linkedin.com/in/hbehrens)





	Pebble Time	Pebble Time Round
CPU	ARM Cortex M4 @ 100 MHz 64KB RAM (incl. code and 2KB stack)	
Display	144x168px @ 64 colors (Rectangular)	180x180 @ 64 colors (Circular)
Peripheral	Bluetooth LE, 4 Buttons, Accelerometer, Magnetometer, Ambient Light Sensor, Vibrations, Microphone, SmartStrap	
OS	based on FreeRTOS with multiple tasks (system UI + unprivileged apps)	
3rd Party	appstore with >12,000 apps, >500 daily active developers CLI SDK & cloud-based environment	





Pebble default watchface "TicToc" on firmware 3.12 is using JerryScript.

SDK for 3rd-party developers on the road map for this year.

```
var renderState = {
  minute: {style: 'white', scale: 0.80, angle: 0},
  hour:   {style: 'red',   scale: 0.51, angle: 0}
};

var drawHand = function(handState, ctx, cx, cy, maxRadius) {
  ctx.lineWidth = 8;
  ctx.strokeStyle = handState.style;
  ctx.beginPath();
  ctx.moveTo(cx, cy);
  ctx.lineTo(cx + Math.sin(handState.angle) * handState.scale * maxRadius,
            cy + Math.cos(handState.angle) * handState.scale * maxRadius);
  ctx.stroke();
};

rocky.on('draw', function(drawEvent) {
  var ctx = drawEvent.context;
  var w = this.innerWidth;
  var h = this.innerHeight;

  // clear canvas on each render
  ctx.fillStyle = 'black';
  ctx.fillRect(0, 0, w, h);

  // center point
  var cx = w / 2;
  var cy = h / 2;
  var maxRadius = Math.min(w, h - 2 * 10) / 2;
  drawHand(renderState.minute, ctx, cx, cy, maxRadius);
  drawHand(renderState.hour, ctx, cx, cy, maxRadius);

  // Draw a 12 o'clock indicator
  drawHand({style: 'white', scale: 0, angle: 0}, ctx, cx, 8, 0);
});

// listener is called on each full minute and once immediately after registration
rocky.on('minutecchange', function(e) {
  var wfh = new WatchfaceHelper(e.date);
  renderState.minute.angle = wfh.minuteAngle;
  renderState.hour.angle = wfh.hourAngle;
  rocky.requestDraw();
});
```

# Findings (based off JerryScript mid Feb)

- **global state introduces challenges**  
(multi tasking, uninitialized state)
- **launch performance** (due to parsing, execution)
  - **parsing** (10KB JS ~5s)
  - **execution** (below 10 FPS)
- **stack** (TicToc.js requires ~8KB)
- **development is difficult**
  - **debugging** (poor stack traces)
  - **education around embedded** (low-memory)



WebAPIs Example x Heiko

file:///Users/behrens/Documents/projects/pebble/rockyjs/examples/webapis/index.html

Paused in debugger 493px x 590px

Elements Console Sources Network Timeline Profiles Resources Security Audits Grunt

html-binding.js tictoc.js\* rocky-WebAPIBinding.js VM166 tictoc.js x

```
21 ctx.lineTo(cx + Math.sin(handState.angle) * handState.scale * maxRadius,
22           cy + Math.cos(handState.angle) * handState.scale * maxRadius);
23 ctx.stroke();
24 };
25
26 // the 'draw' event is being emitted after each call to rocky.requestDraw() but
27 // at most once for each screen update, even if .requestDraw() is called frequently
28 // the 'draw' event might also fire at other meaningful times (e.g. upon launch)
29 rocky.on('draw', function(drawEvent) { drawEvent = Object {context: R...y.CanvasRenderingCont
30     var ctx = drawEvent.context; ctx = R...y.CanvasRenderingContext2D {binding: Object, cPtr: 1
31     var w = this.innerWidth; w = 144
32     var h = this.innerHeight; h = 168
33
34     // clear canvas on each render
35     ctx.fillStyle = 'black'; ctx = R...y.CanvasRenderingContext2D {binding: Object, cPtr: 16, c
36     ctx.fillRect(0, 0, w, h); w = 144, h = 168
37
38     // center point
39     var cx = w / 2; cx = 72, w = 144
40     var cy = h / 2; cy = 84, h = 168
41     var maxRadius = Math.min(w, h - 2 * 10) / 2; maxRadius = 72, w = 144
42     drawHand(renderState.minute, ctx, cx, cy, maxRadius);
43     drawHand(renderState.hour, ctx, cx, cy, maxRadius);
44
45     // Draw a 12 o clock indicator
46     drawHand({style: 'white', scale: 0, angle: 0}, ctx, cx, 8, 0);
47     // overdraw center so that no white part of the minute hand is visible
48     drawHand({style: 'red', scale: 0, angle: 0}, ctx, cx, cy, 0);
49 });
50
51 // listener is called on each full minute and once immediately after registration
52 rocky.on('minutechange', function(e) {
53     // WatchfaceHelper will later be extracted as npm module
54     var wfh = new Rocky.WatchfaceHelper(e.date);
55     renderState.minute.angle = wfh.minuteAngle;
56     renderState.hour.angle = wfh.hourAngle;
57     rocky.requestDraw();
58 });
59
```

Watch

Call Stack

- (anonymous function) VM166 tictoc.js:42
- rocky-WebAPIBinding.js:17
- \_private.subscriptions.emit
- rocky-WebAPIBinding.js:32
- \_private.binding.update\_proc
- callback html-binding.js:77
- (anonymous function) html-binding.js:79

Paused on a JavaScript breakpoint.

Scope

Local

- ctx: Rocky.CanvasRenderingContext2D
  - cx: 72
  - cy: 84
- drawEvent: Object
  - h: 168
  - maxRadius: 72
- this: Rocky.WebAPIBinding
  - w: 144

Global Window

Breakpoints

- VM166 tictoc.js:42
  - drawHand(renderState.minute, ctx, cx, ...

DOM Breakpoints

XHR Breakpoints +

Event Listener Breakpoints

Event Listeners ↻

{ } Line 42, Column 3

<http://pebble.github.io/rockyjs/>

Cross-compiled our firmware's application layer with Emscripten to develop apps in the browser.

# Near-Term Challenges

- stable snapshot format
  - rare changes (or forward-compatible)
  - compatible across platforms
  - ideal: executes & stores heap of root-level script
- lower stack requirements (e.g. storing as part of `mem_heap_t`)
- education / low memory strategies

# Medium-Term Challenges

- performance
- multi-instance
- debugging & tooling