

```

<polymer-element name="webaudio-knob"
  attributes="value min max step splites
  diameter src" on-mousedown="mousedown">
  <template>
    <div class="knob" id="knob">
      <span id="value-tip">{{value}}</span>
    </div>
    <style>
      @host {
        * {
          display: inline-block;
        }
      }
      #knob {
        cursor: pointer;
        width: 64px;
        height: 64px;
        background-position-y: -64px;
        position: relative;
      }
      #value-tip {
        opacity: 0;
        border: solid 1px #666;
        background-color: #eee;
        position: absolute;
        top: 0;
        right: 0;
        padding: 1px 4px;
        font-size: 10px;
        font-family: Helvetica;
        transition: opacity 0.3s;
      }
    </style>
  </template>
  <script>
    Polymer('webaudio-knob', (function() {
      var mousemove = function(e) {
        var offset = (this.startPos - e.y) ||
0;
        var value = this.startVal +
(e.shiftKey ? ~(offset / 3) : this.step *
offset);
        update.bind(this)(value);
      };

      var cancel = function(e) {
        this.startPos = null;
        this.$['value-tip'].style.opacity = 0;

window.removeEventListener('mousemove',
this.boundMousemove, true);
        window.removeEventListener('mouseup',
this.boudCancel, true);
      };

      var update = function(value) {
        this.value = value < this.min ?
this.min : value > this.max ? this.max :
value;
        var range = Math.abs(this.min) +
Math.abs(this.max);
        var value = '-' + ~(this.splites /
range * (range - this.min + this.value) + 1) *
this.diameter + 'px';

this.$['knob'].style.backgroundPositionY =
value;
        this.fire('change');
      };

      return {
        ready: function() {
          this.$['knob'].style.width =
this.diameter+'px';
          this.$['knob'].style.height =
this.diameter+'px';
          this.$['knob'].style.background =
'url('+this.src+')';
          update.bind(this)(this.value);
        },
        value: 0,
        min: 0,
        max: 127,
        diameter: 64,
        step: 3,
        splites: 30,
        src: 'img/org_amp.png',
        startPos: null,
        startVal: 0,
        mousedown: function(e) {
          this.$['value-tip'].style.opacity =
1;

          this.startPos = e.y;
          this.startVal = this.value;
          this.boundMousemove =
mousemove.bind(this);
          this.boundCancel =
cancel.bind(this);
          window.addEventListener('mousemove',
this.boundMousemove, true);
          window.addEventListener('mouseup',
this.boundCancel, true);
          e.preventDefault();
        }
      };
    })());
  </script>
</polymer-element>

```