Keyboard: Hardware Device Issues

- Keyboard design
- Keyboard layout
  - CR, BSP, DEL, CTRL, FN
  - Cursor keys
  - Number pad (123 on top vs. bottom)
  - QWERTY, DSK,...
- Key size/shape
  - Flat vs. contoured top
- Key type
  - E.g., membrane
- Scale
Keyboard: Hardware Device Issues

- Keyboard design
- Keyboard layout
  - CR, BSP, DEL, CTRL, FN
  - Cursor keys
  - Number pad (123 on top vs. bottom)
  - QWERTY, DSK, ...
- Key size/shape
  - Flat vs. contoured top
- Key type
  - E.g., membrane
- Scale

Locator

- Specifies a location
- Can be used to select, position, orient, specify path, quantify, enter text
Locator: Hardware Device Issues

- Grip
  - Stylus vs. mouse
- Time to pick up
- Active vs. passive stylus
- Mouse tracking technology
- Case study: Mouse design
  - Shape
  - Buttons
  - ...

Case Study: Mouse Design
Case Study: Hacking a new device

Soap

- Wireless optical mouse internals repackaged in lozenge-shaped plastic “core” and surrounded by cloth “hull”
- Manipulation approaches
  - “Joystick”
    - Similar to handheld spring-loaded joysticks and “upside-down optical mouse”-like devices
  - “Belt”
    - Drag hull with core stationary
  - “Soap”
    - Flip core with hull stationary


---

Three-State Model

- Mouse
  - Two states: tracking (button up), dragging (button down)
  - State names are just examples

- Touch Tablet/Pad (separate from display)
  - Two states: out of range (not touching), tracking (touching)
  - State 0 is used to designate a state the system cannot sense
  - State 1–State 0 transition = “pen has left the paper”
    - Does this generate an explicit event?

Note: Actual devices shown in photos may have a superset of the states listed here

---
Three-State Model  

- Stylus with Tip Switch
  - Three states: *out of range* (not touching), *tracking* (touching, with switch open), *dragging* (touching, with switch closed)

Multi-button Mouse
- Three states: *tracking* (buttons up), *dragging a* (button a down), *dragging b* (button b down)
  - Could use double clicking on a one-button mouse to distinguish among states 2a/b
  - Could also have state 2ab (both buttons down)

Note: State names (*tracking*, *dragging*) are just examples. E.g., state 2 could instead be drawing, interacting with a menu,...

Note: *Tracking* is sometimes known as *hovering* (esp. when the device doesn’t change position for some set amount of time)

- How to support selection in 0–1 state device?
  - Use state 1–state 0 transition (liftoff)
  - Use double tap/click
  - Use timeout cue (point at object for time \( t \geq t_{\text{select}} \))
  - Use pressure threshold (if device can detect >1 level of pressure)
  - Use number of fingers or area of touch (if device can detect)

- Use other hand on other device (e.g., to push a button)
- Add a button to device


- Direct input devices (e.g., touch screen)
  - Out of range state 0 actually supports passive tracking since unsensed pointing device/finger can act as its own cursor
  - Not state 1, since system cannot sense it

- In contrast, compare with the [indirect] touch tablet/pad (in which finger on tablet/pad cannot be viewed in context of screen in state 0)
  - Same gesture, different context

- **Mouse**
- **Legacy AM/FM radio**

Fitts’s Law  P. Fitts, 1954

- Predictive model of time $MT$ to move a distance $A$ to target of width $W$.
  - $MT$ increases with increasing $A$, decreases with increasing $W$
  - Farther/smaller target $\rightarrow$ longer time to reach
    Closer/bigger target $\rightarrow$ shorter time to reach
  - $MT = C_1 + C_2 \ ID$
    - $ID$ = Index of Difficulty (function of $A$ and $W$)
    - $C_1$ = Device/appendage-dependent constant
    - $C_2$ = Device/appendage-dependent constant
Fitts’s Law P. Fitts, 1954

- Original task used electrical contacts
- Parameters varied from $A = 1''$, $W = 1''$ to $A = 16''$, $W = .25''$
- $ID = \log_2 (2A / W)$
  - Conventionally measured in bits, after Shannon
- $ID = \log_2 (A / W + 1)$
  - Later formulation has slightly better fit, and assures positive $ID$ (Mackenzie)

In original reciprocal tapping task, participant alternated between tapping two bars

MT = $C_1 + C_2 \, ID$

- where $ID$ measured in bits
- $C_2$ measured in secs/bit, ca. .1 sec/bit (range ca. 83 msec/bit – 430 msec/bit)
  - E.g., higher for button-down dragging
- $IP$ (Index of Performance) = $1 / C_2$
  - Measured in bits/sec (ca. 12 – 2.3 bits/sec)
  - Also known as throughput or bandwidth
- $MT = C_1 + ID / IP$
Fitts’s Law P. Fitts, 1954

- \( MT = C_1 + C_2 \log_2 (A / W + 1) \)
- \( MT = \ldots + C_2 \ ID \)
  - \( C_2 = \) slope
    - Higher \( C_2 \) means steeper curve, corresponding to lower IP (1/C_2)
- \( MT = C_1 + C_2 \ ID \)
  - \( C_1 = \) accounts for intercept offset from 0

\[ \text{MT (secs)} \quad \text{ID (bits)} \]

Hard

Easy