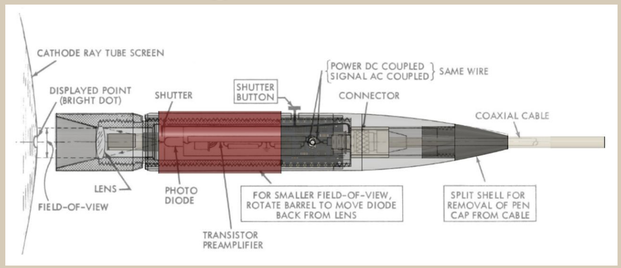
Sketchpad++

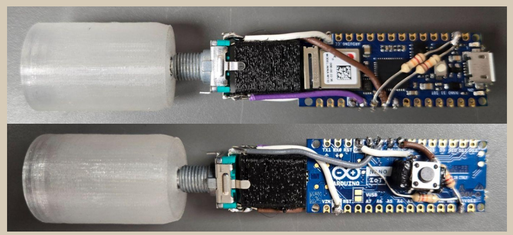
Parker Connelly, Daniel Flyer, Daniel Simone, Alex Ni

Press L to initiate a **line** from cursor position, and flick to finish it. Press C to initiate a *clockwise* **circle** (X *counterclockwise*), flick to set the radius, draw the circle, and flick again to finish it.
While hovering over a point, press M to **move** it.
While hovering over any element, press D to **delete** it.
While hovering over a line, press E / P and select another line to initiate an **equal length/parallel line constraint**.
While hovering over a line, press V / H to make it **vertical/horizontal**.
Press Q to **display** *objects under the constraints* and T to watch the constraints happen (**tick**) in real-time. Press A / S to **zoom in/out**.

Light Pen



- Diagram overlaid with *original Sketchpad schematic* for the Light Pen - closely mimics original.
- 6-½” in length (165.4mm) and just under ¾” (21.2mm) in diameter.
- Composed of four **3D printed parts**.
- Integrates **Arduino Nano 33 IoT** with **LSM6DS3 IMU chip**, **rotary encoder**, push button, and resistors.



- Two adjustable elements:
  - **Shutter button:** press once to stop mouse movement, align with cursor, press to re-engage
  - **Focus knob:** rotate to adjust the mouse sensitivity based on the distance between pen and screen.

References

I. E. Sutherland, "Sketchpad---A man-machine graphical communication system," *Proc. Spring Joint Computer Conference (AFIPS)*, 1963, pp. 328-346.

Hardware Integration

- Light Pen IMU reads *3 axis gyro* and *accelerometer.*
- **IMU measurements** converted into *euler angles* using a *Madgwick Filter*.
- Angle is projected into **X,Y position** on the screen based on the distance that is **set by the rotating head**.
- Position sent to Python script, via *serial*, that controls **mouse position**.
- Positions relative to reference point set by clicking the shutter button.
- A **fast flick** of the pen is detected by the gyro sensor and causes a **mouse click** to occur.

JavaScript

- Try it out in your browser at parkerc22.github.io/Sketchpadpp/
- Recreated function of **drawing components** and **constraints**, but with *different object hierarchies* and *enforcement mechanisms* from the original.
- Supports **moving points** and **snapping** onto lines, circles, and other points.
- Supports **radial**, **length**, **parallel**, and **horizontal/vertical** constraints.
- **Zoom in/out** centered at cursor.
- Component **deletion** affects other dependent components.