

# ARDW: An Augmented Reality Workbench for Printed Circuit Board Debugging

Ishan Chatterjee  
ichat@cs.washington.edu  
Paul G. Allen School of Computer  
Science & Engineering, University of  
Washington  
Seattle, WA

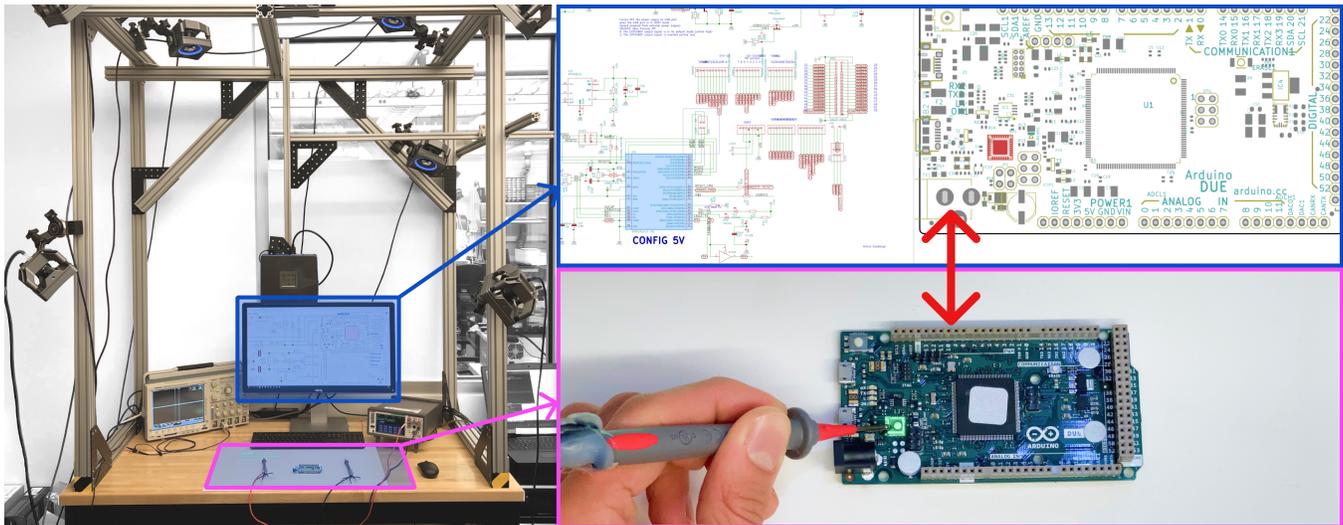
Tadeusz Pforte  
pfortet@cs.washington.edu  
Paul G. Allen School of Computer  
Science & Engineering, University of  
Washington  
Seattle, WA

Aspen Tng  
aspentng@uw.edu  
Human-Computer Interaction +  
Design, University of Washington  
Seattle, WA

Farshid Salemi Parizi  
farshid@uw.edu  
Electrical and Computer Engineering  
University of Washington  
Seattle, WA

Chaoran Chen  
chaoranc@andrew.cmu.edu  
Human-Computer Interaction  
Institute, Carnegie Mellon University  
Pittsburgh, PA

Shwetak Patel  
shwetak@cs.washington.edu  
Paul G. Allen School of Computer  
Science & Engineering, University of  
Washington  
Seattle, WA



**Figure 1:** ARDW is an end-to-end system (left) that enables cross-linking between the physical printed circuit board (bottom right) and the design files on a PC (top right) via projected AR (green highlight, bottom right). When a user selects a components, pins or pads on the PCB, the corresponding element gets highlighted on the design files and vice versa (right).

## ABSTRACT

Debugging printed circuit boards (PCBs) can be a time-consuming process, requiring frequent context switching between PCB design files (schematic and layout) and the physical PCB. To assist electrical engineers in debugging PCBs, we present ARDW, an augmented reality workbench consisting of a monitor interface featuring PCB

design files, a projector-augmented workspace for PCBs, tracked test probes for selection and measurement, and a connected test instrument. The system supports common debugging workflows for augmented visualization on the physical PCB as well as augmented interaction with the tracked probes. We quantitatively and qualitatively evaluate the system with 10 electrical engineers from industry and academia, finding that ARDW speeds up board navigation and provides engineers with greater confidence in debugging. We discuss practical design considerations and paths for improvement to future systems. A video demo of the system may be accessed here: <https://youtu.be/RbENbf5W1fc>.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).  
UIST '22, October 29–November 2, 2022, Bend, OR, USA

© 2022 Association for Computing Machinery.  
ACM ISBN 978-1-4503-9320-1/22/10...\$15.00  
<https://doi.org/10.1145/3526113.3545684>

## CCS CONCEPTS

- **Human-centered computing** → **Mixed / augmented reality**;
- **Hardware** → **Printed circuit boards**; *Board- and system-level test*.

## KEYWORDS

Augmented Reality, Printed Circuit Board, Hardware, Debugging

### ACM Reference Format:

Ishan Chatterjee, Tadeusz Pforte, Aspen Tng, Farshid Salemi Parizi, Chaoran Chen, and Shwetak Patel. 2022. ARDW: An Augmented Reality Workbench for Printed Circuit Board Debugging. In *The 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22)*, October 29–November 2, 2022, Bend, OR, USA. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3526113.3545684>

## 1 INTRODUCTION

The increasing ubiquity of technology has been spurred by rapid advances in electronics manufacturing over the past few decades. With developments such as surface mount technology (SMT) and integrated circuit (IC) packaging, printed circuit boards (PCBs) have become increasingly dense and complex featuring tens, hundreds, or even thousands of components and double-digit layer counts. Meanwhile, the tools to support engineers in debugging faulty PCBs during the design phase remain mostly unchanged — engineers probe various nodes on the PCB with test equipment while referencing the schematic and layout diagrams to reason about their circuit.

During the process of debugging a new PCB design, engineers must constantly move between their schematic, layout, and physical representations in order to validate their design or understand the nature of a design failure<sup>1</sup> (Fig. 1). Through current ECAD tools, this can be a time-consuming and error-prone process which typically involves manually following correspondences and nomenclatures across different software applications, often using each application’s textual “find” command. To locate the corresponding area-of-interest on the board, the engineer must textually match the reference designator against the board’s silkscreen (if printed) or visually pattern match layout representation to the physical PCB, which can become more challenging with dense components or different orientations. As a debugging procedure typically involves tens to hundreds of these correspondences, this procedure can quickly become tedious with the ever-increasing complexity of board designs only exacerbating the challenge. While the human-computer interaction (HCI) community has mainly focused on supporting makers, prototypers, hobbyists, and students with tools to help support the debugging of low-volume breadboard designs [12, 14, 19, 24], there has been less work toward supporting the workflow of circuit

<sup>1</sup>Some helpful technical background and terminology: The electrical engineering design process typically starts with designing a circuit to meet a set of functional requirements. Using an electronic computer-aided design (ECAD) tool, the logic of the circuit is formalized via schematic capture into a *schematic diagram*, which visualizes the circuit’s components as symbols and the circuit’s interconnections (*nets*) as topological lines between the components’ pins. This logic is then transferred to a *layout diagram*, where components and connections are placed in a physical coordinate space. Finally, the design is then physically fabricated and assembled into a functional PCB, where the components are soldered onto the surface of a fiberglass board with conductive pads, vias, and traces running buried within its layers. For images, see online tutorials on Sparkfun (<https://learn.sparkfun.com/tutorials/pcb-basics/all>) or Adafruit (<https://learn.adafruit.com/making-pcbs-with-oshpark-and-eagle>).

design and development moving towards production [13]. We seek to add to the small but growing area of HCI literature focusing on PCB design [8, 16, 21]. Finally, an increasingly globalized supply chain, design outsourcing, and pandemic-normalized remote work have each introduced new challenges in debugging PCB designs collaboratively.

Previous design research has shown that electrical engineers are interested in augmented reality (AR) as a paradigm to help reduce the cognitive load, accelerate common tasks within debugging workflows, and enable remote collaboration [6]. In this work, we build and present ARDW (Augmented Reality Debugging Workbench), an open-source, end-to-end system that enables cross-linking between virtual design files and the physical PCB via projected AR and tracked test probes. At its core, ARDW enables *augmented visualization* by facilitating for selections in the design files to be highlighted on the physical PCB, and *augmented interaction* by allowing for selections and measurements on the physical PCBs to be carried to the design files. In this way, ARDW is the first system to provide an augmented *bidirectional* cross-link across schematic, layout, and physical PCB. We conduct a study with ten electrical engineers using the tool across a set of board navigation, bring up, and simulated debugging tasks. All participants verified that the reduced context-switching between the PCB and design files allowing for users to more efficiently localize items on the PCB and capture measurements, and making their debugging experience more seamless. Finally, building on the user feedback from the user study, we offer design considerations towards producing future AR systems for PCB debugging.

## 2 RELATED WORK

Electrical engineers use a diverse set of inspection tools across various stages of the design process. In the early engineering validation test phase, manual tools such as oscilloscopes, logic analyzers, and multimeters are used to provide measurements of signal properties on the physical board. For high volume production, select functions maybe be tested via automated in-circuit testing (ICT) equipment, a test machine that accesses a set of PCB test points via a custom spring-loaded pogo pin fixture, making ICT expensive and time-consuming to set up. While allowing for high-throughput testing, the test suite only has access to the limited set of test points for diagnostic checks and is therefore used as an end-of-line functional check, rather than allowing for root causing and free-form debugging. In this paper, we focus on tools to enable better debugging for the early stages of development and validation. In this section, we first cover related work on enabling better debugging workflow on breadboards and later cover work related to extending the capabilities of PCBs.

### 2.1 Extending Breadboards

Breadboards offer a gridded construction base in which electronic components can be inserted or removed. Because of their easy solderless reconfigurability, breadboards are commonly used amongst hobbyists, students, designers, and engineers for prototyping simple, one-off circuits. Prior work has explored augmenting breadboards to extend their capabilities, primarily for enabling measurement and testing features. Toastboard [7], CurrentViz [24], and

Heimdall [12] make real-time measurements on a circuit prototyped on an instrumented breadboard, specifically voltage and current information respectively, and show this information on a computer screen. Toastboard additionally uses status LEDs on each row of the breadboard to provide localized feedback, a feature cited by all of their study participants as being useful during debugging. In addition to making measurements, previous work has also explored adding *in situ* reconfiguration and test capabilities to breadboards. Visible Breadboard allows a user to create virtual wires by dragging a finger along the desired path on a breadboard [17]. A layer of relays under the breadboard reconfigures itself to form the desired electronic connection. Proxino similarly enables reconfiguration of an underlying layer such that users can combine virtual and physical components [23].

Due to their modular nature, breadboards are helpful for rapid prototyping and primarily used for educational purposes. However, they are often not practical in supporting the prototyping of circuits for commercial products, as the physical design of breadboards restricts the possible package choices and layouts of the electronic components and introduces contact resistance and parasitic capacitance that can affect sensitive or high speed analog signals. Furthermore, replicating breadboards for more than a handful of prototypes can be tedious. In our work, we seek to support engineers in academia and industry who generally develop PCBs for medium-volume development and mass production.

## 2.2 Extending Printed Circuit Boards

**2.2.1 Augmented Visualization on PCBs.** A number of systems incorporated virtual augmentations on PCBs in varying capacities. InspectAR is a startup that has developed a tool using mobile AR to overlay elements of a board layout and associated metadata onto the camera’s view of a PCB displayed on a mobile tablet or PC [2]. While serving detailed visualizations and component datasheets, the tool does not support direct interaction with the PCB itself, measurement interactions, or a topological schematic view for debugging. The Mascot, a robotic workbench from Robotas, helps to support operators performing hand assembly of through hole components [3]. The tool allows for preloading of assembly steps, which controls automated picking carousels and a projected laser spot showing assembly position on a clamped PCB. Similarly, Hahn et al. [10] built an AR system that delivers textual and graphical cues through a head-mounted display for assisting workers performing PCB assembly, and indicates that the tool helped minimize error in part picking and assembly. HoLOsci [11] is a tool that incorporates voice commands and grabs an oscilloscope screen and overlays it onto the handheld probe to prevent users from slipping off their probe point when interacting directly with the instrument panel.

Hahn et al.’s tool, InspectAR, Mascot, and HoLOsci all provide board-locked augmented instruction for PCB workflow, driving information from the virtual design files or instrument panel to the user’s view of the PCB. Our work seeks to also incorporate augmented interaction and measurement to pass data in the opposite direction, that is, interactive capture directly on the physical PCB can be passed to the virtual design files to assist engineers in debugging.

**2.2.2 Augmented Interaction and Measurement on PCBs.** Pinpoint [21] is a tool designed to assist in PCB debugging by adding up to 16 jumper pads into a PCB design before fabrication, generates a bed-of-nails jig to interface with the jumper pads, and modifies connections through a GUI-controlled relay board. While not using augmented reality per se, the tool connects the virtual and the physical by using GUI-controlled relays to make and break these connections. However, modifications to layout in acceptable in production designs in support of test is typically restricted to adding test points only on critical, low-speed nets for bed-of-nails ICT or manual access for workbench debugging. Further work by Stransnick et al. has explored the utility of tightly coupling simulated and physical versions of the same circuit to support interactive debugging by leveraging instrumentation and programmable test hardware [20]. Our work seeks to support existing debugging workflows that do not modify the PCB design or add new test hardware, and instead ease access to measurement points by guiding users with augmentations.

Most relevant to our work is BoardLab which presents a magnetically tracked stylus that enables interactions from board to schematic, such as selecting and identifying components on the schematic by touching the components on the board as well as taking voltage measurements and having the measurement annotated on the schematic [8]. Although the system looks promising, BoardLab does not comment on system accuracy, implement board tracking, or conduct a usability study of their device. Our work fuses the augmented interaction demonstrated in BoardLab with augmented visualization on the PCB, and discusses the practical usability considerations of the system.

**2.2.3 Combining Augmented Visualization and Interaction on PCBs.** In previous design work [6], we presented a set of augmented reality *interaction techniques*, called Augmented Silkscreen, to assist electrical engineers in PCB debugging. Through a set of remote design interviews, we found that *combining* augmented visualization (as seen in works like InspectAR) and augmented interaction (as seen in works like BoardLab) on PCBs unlocks promising avenues to alleviate frequent context switching between schematic, layout, and physical PCB. By having information flow bi-directionally between virtual design files and physical PCB, compelling interactions can be realized; for example, providing on-board augmentations to guide a user toward a measurement and then logging that measurement and its location. In this work, we build an end-to-end system that implements the interaction techniques proposed in Augmented Silkscreen. We run a user study to explore the efficacy of the system in practical scenarios.

## 3 WORKFLOW AND SYSTEM FEATURES

We introduce the features of ARDW through an exemplary PCB development scenario. The walk-through illustrates a few common issues electrical engineers may encounter during a typical development scenario and discusses how ARDW can help to assist the electrical engineer in debugging them. Figure 2 and the video demo (<https://youtu.be/RbENbf5Wifc>) provide images and clips to help visualize these interactions. All features discussed in this section are implemented in our system. See section 4 for full implementation details.

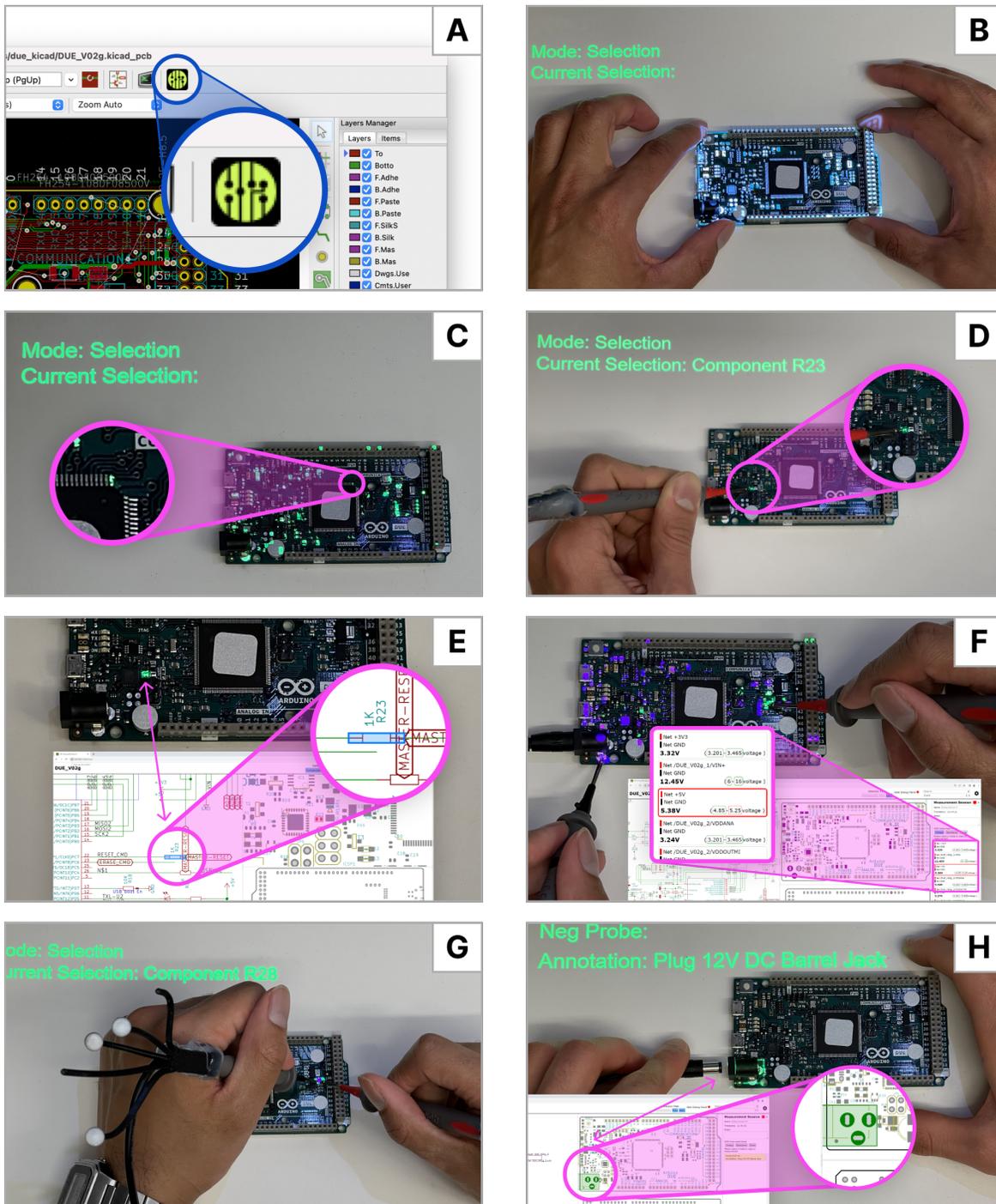


Figure 2: (A) users can bring in their file from KiCAD using ARDW’s import utility, (B) user manually aligns PCB to projection, (C) user leverages ARDW to highlight all pin 1s, including the pin on this IC, (D) user employs tracked probe to select components or pins by dwelling on the components, (E) by selecting in any view, such as the board, the item in the schematic and layout are both highlight, (F) when a user specifies a guided measurement in the measurement panel, the probe points are highlighted, the user can take the measurement and it is automatically captured in the measurement panel, (G) selecting an item in the schematic highlights the component on the board allowing users to find the correct item quickly, (H) users can provide guided annotations which are particularly helpful when collaborating with others remotely.

Kofi is designing an 8-layer PCB for a microcontroller development kit in KiCAD. The design is in the early Engineering Validation Test (EVT) phase of the hardware development process, where quantities are low, between 50 and 500 units.<sup>2</sup> Kofi and his team have received the shipment of 70 assembled boards from the fabrication house. Kofi has a ARDW system installed in his lab’s workspace. ARDW generally mirrors the set up of other workbenches in the lab with a rubberized, anti-static mat on the table surface, a PC, monitor, keyboard, and mouse, and a benchtop digital multimeter; however, it also has a few additional components: a set of eight overhead-mounted tracking cameras and overhead mounted projector (see fig. 4).

### 3.1 Loading Design

To start, Kofi first exports his design from KiCAD into ARDW’s viewer using ARDW’s import utility, which integrates into KiCAD as a third-party plug-in (see Fig. 2A). The plug-in parses .svgs plotted from KiCAD’s schematic editor as well as the netlist files produced during the PCB’s design phase. This loads the schematic sheets and the board’s front and back views into the on-screen interface of the workbench’s monitor. By default, the schematic and layout sit side-by-side, however if Kofi’s screen real estate is at a premium, it is possible to resize views or just see a single view.

By default, an outline of the board’s edge cuts as well as the pads for all components are projected on the anti-static mat on the table surface via the overhead mounted projector. Kofi aligns the board with the projected outline, or, if preferred, attaches stick-on tracking markers to flat areas of the PCB to permit the board to be tracked by the overhead cameras (see Fig. 2B).

### 3.2 Visual Inspection

Kofi first starts with a manufacturing check on a subset of the delivered boards. Kofi first seeks to check that the “Do Not Populate” (DNP) components are correct.<sup>3</sup> He selects the “DNP” filter which highlights all components on the board that should not be installed. Typically, Kofi would need to look between a layout diagram where DNP components are highlighted and board multiple times to ensure that each component is not installed appropriately. On smaller or more mature boards this number can be a handful of components, but on larger or early development designs this may be tens of components. With ARDW, Kofi is able to immediately check the appropriate components are not installed.

Next, Kofi looks to confirm the orientation of all functionally-asymmetric components such as electrolytic capacitors, diodes, and ICs, are correct. They select the “Pin 1” filter in ARDW which highlights the leading pin of each component (see Fig. 2C). Kofi ensures the package’s pin 1 marking matches the highlighted pin 1.

Finally, Kofi performs a visual inspection of the board looking for board damage, anomalous component placement, cold or cracked solder joints, or other manufacturing issues. He finds a tombstoned

resistor.<sup>4</sup> Kofi selects the component by placing the tip of his tracked probe on top of it and holding for half a second (see Fig. 2D). The component is highlighted via projected augmentation as well as in the schematic and layout views indicating selection (see Fig. 2E). Kofi can immediately see in the schematic view that the resistor is in series with the enable pin of a critical IC. He can observe in the layout view that the highlighted footprint differs from other adjacent resistors of the same package which likely caused the issue. He re-solders the component by hand, and notes that this incorrect footprint will need to be rectified in the next revision. He cycles through other boards while keeping the resistor highlight on to spot-check that area on other boards, finding the same issue occurring on two other boards.

Without ARDW, Kofi would typically see a fault on the board, such as a tombstoned resistor, and then visually pattern match that area of the board with the layout diagram to determine the damaged component’s reference designator (for example, R238). He can then use this reference designator or cross-probing to find the corresponding component in the schematic, and determine the component’s function. Often small components and dense boards do not include silkscreen with the component’s reference designator. With ARDW, moving between the representations is accelerated.

### 3.3 Bring up

Next Kofi performs a bring-up procedure for his boards. During bring-up, engineers typically step through all the major power rails on their boards, first probing for resistance (to ensure no shorts to ground). They then apply power and measure the voltage of each rail to ensure each rail are within an expected voltage bound, often recording their measurements in a prepared spreadsheet. With ARDW’s measurement sidebar (Fig. 2F, right), Kofi pre-loads a set of rails he is interested in measuring, the type of measurement, and optional test bounds. He then clicks “record.” ARDW starts with his first specified measurement—a voltage measurement between net VSYS\_3V3 and GND. The system highlights all the pin locations for the VSYS\_3V3 net in green (the color of the dot tracked under the positive probe) and the pin locations for GND in purple (the color of the dot tracked under the negative probe) (see Fig. 2F). Kofi can immediately identify the most convenient location to take the measurement. ARDW is connected to his digital multimeter via the VISA API allowing the system to automatically set the instrument into voltage mode. Kofi places his probes on the highlighted pads and holds for half a second. The system recognizes the location of the probes to match the currently-specified measurement, captures the voltage, and records it to the debug panel, highlighting if the measurement panel is outside the specified test bounds. The system moves to the next guided measurement panel and displays the next set of measurement pin or net locations. We call this mode “guided measurement.” In this manner, Kofi is able to efficiently move through the set of required measurement test cases.

### 3.4 Free-form debugging

After bring-up, Kofi finds that a net, +5V, is sitting slightly higher than its upper test bound on three of the boards he has tested. He

<sup>2</sup>For more information on hardware development and release process see: <https://instrumental.com/resources/factory/hardware-engineers-speak-in-code-evt-dvt-pvt-decoded/>

<sup>3</sup>also known as “nostuff”, “NS”, “unstuff”, engineers will instruct the assembly house to omit populating components to disconnect unused sub circuits or provide optionality to modify functionality without a board re-spin

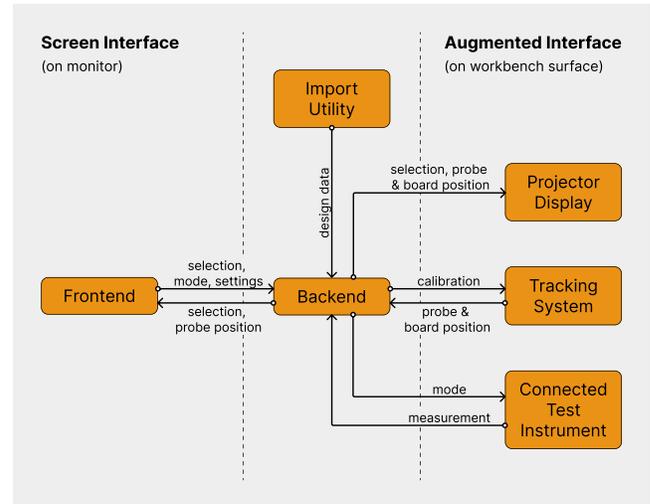
<sup>4</sup>when a two-pin package only solders on one side causing the other terminal to float above its pad

selects one of the faulty boards, and traces the power path to localize the issue. He starts at the barrel jack input, clicking on the jack's positive pin in the schematic. This highlights the corresponding pin on the board. He chooses to record his measurements, selecting the record button in the measurement panel. He uses the system's probe selection filter, deselecting components and nets, to filter only by pins. This way, when he takes the voltage measurement, the system records the pin name locations and associated measurement. He finds the voltage to match his expectations. He continues down the power path repeating the set of above steps, using the schematic to guide his series of measurements. Finally, he arrives at the output of the 12V-to-5V buck converter IC, noting the higher-than-expected voltage on the output despite the expected voltage on the input. After localizing the issue to the buck converter sub-circuit, he accesses the IC's datasheet and determines the buck's feedback network uses a resistor divider to set the output voltage. He removes power from the board, and puts the DMM into resistance mode. Selecting the resistor in the schematic highlights the component on the board (see Fig. 2G). He probes across the resistor and notes the top resistor in the resistor divider is 5% lower than its nominal value listed in the schematic. He takes the same measurement across a random sample of boards finding that the faulty boards all have a similar anomaly. He realizes that he mistakenly approved a manufacturer-requested resistor substitution for a  $\pm 5\%$  tolerance component, whereas originally his design called for a  $\pm 1\%$  tolerance part. As opposed to guided measurement described in the previous section, free-form debugging allows for the engineer to perform root causing on-the-fly and for their measurements to be automatically logged.

### 3.5 Collaboration

To validate his hypothesis, Kofilooks to perform a rework on ten boards and validate that the rail sits within specified test bounds. However, Kofi's rework team sits in a different office. He sends ten boards to the rework team's office which has an ARDW system in the rework lab. Traditionally, Kofi would take a screenshot of his layout and email an annotated image of the rework request to the rework technician, Deepali. With ARDW, Kofi instead is able to provide a set of annotated instructions to be projected directly on the PCB (see Fig. 2H). Additionally, due to ARDW's client-server architecture, Kofi and Deepali can share a session, facilitating bidirectional remote collaboration in real-time. While on a call, Kofi and Deepali launch a shared session, Kofi highlights the mis-toleranced resistor by selecting the component on his layout, Deepali sees the corresponding component highlighted on her board. Observing the board directly, Deepali sees a tall plastic header next in the vicinity of the resistor, which impedes access with a hot air gun to replace the close-by resistor. Noting the header is non-critical, Kofi permits the rework engineer to remove the header as necessary. This type of reasoning about the physical board can be performed quickly with a shared ARDW session, whereas such a realization and discussion might take multiple exchanges if sharing only 2D screenshots of the board as it typically done.

Kofi prepares a set of annotated instructions for Deepali to validate the fix, following the rework. Some instructions, like plugging in power to the DC barrel jack or toggling a specific switch on the



**Figure 3: Flow of information in the system.** ARDW consists of three main parts. The import utility imports all the design files, the screen interface renders in-browser and displayed the design files and settings to the user, and the augmented interface combines the tracking system, the projector display and the connected test instruments to let the user interact with the board. All these components are connected to the server backend.

board are associated with a component, while some instructions like specifying a guided measurement between the buck IC's output pin and ground are associated with pin locations. As Deepali performs the rework, ARDW projects the steps sequentially on the board reducing ambiguity and the chance for error. When they perform the specified measurement, that measurement is automatically captured and can be exported to a log to share with Kofi. This function helps the design electrical engineer to delegate repetitive validation procedures to others.

## 4 IMPLEMENTATION

Our system has three main components. The *import utility* is used to load designs from KiCAD, an open-source ECAD tool. The *screen interface* on the workbench's monitor delivers the schematic and layout views as well as the bulk of the visualization control of the system. Finally, the *augmented interface* in turn consists of three sub-parts: a *tracking system* for tracking the positions of test instrument probes and the board, a *projector display* for providing augmentations on the physical PCB, and the *connected test instrument* for capturing measurements. These three main components are linked via a Flask<sup>5</sup> server backend. The flow of information within the system can be seen in Figure 3. All of our code is open-source and can be accessed here: <https://github.com/ubicomplab/ar-debug-workbench>.

### 4.1 Import Utility

The import utility is a plugin for KiCad 5.9, an open-source and free ECAD tool. The plugin is written in python 2.7 and runs in KiCad's

<sup>5</sup><https://flask.palletsprojects.com/en/2.1.x/>

layout editor. For schematic data, the plugin consumes SVGs of each schematic page from EESchema, KiCAD’s schematic capture application as well as component library (.lib and .cache-lib) files for schematic symbol hitboxes. Next, our utility cleans the SVGs and collects component metadata data from the schematic file (.sch). We collect net metadata from the netlist (.net) file. For layout data, we extend IBOM’s KiCAD plug-in [1] which uses an API within KiCAD’s layout editor to collect and organize layout data into a format that can be rendered via HTML5 canvas. All above data is finally passed to the server backend as json files, schdata.json and pcbdata.json, which further processes them, including matching all components, pins, and nets across schematic and layout data structures via reference designator or net name.

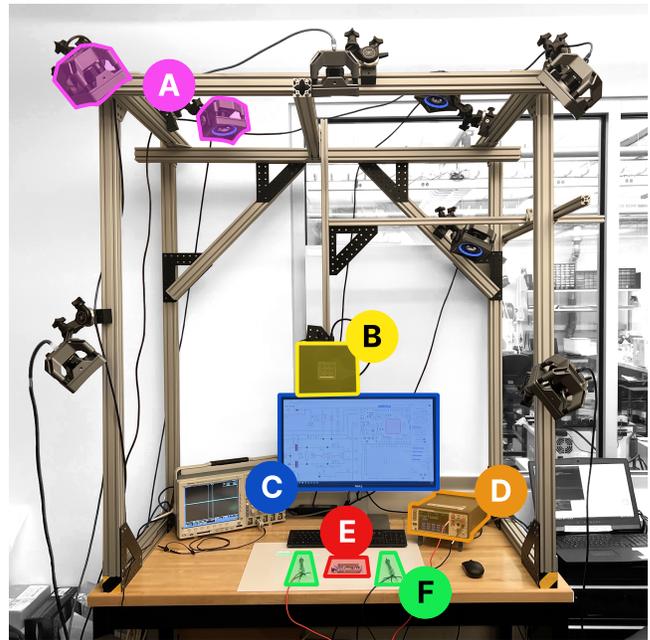
## 4.2 Screen Interface

The screen interface is a web application driven by the Python 3.7 Flask-server backend. It is rendered in-browser and displayed on a 27-inch monitor in front of the user.

The frontend is written in Javascript and is connected to the backend via Socket.IO<sup>6</sup>, which is used to communicate events such as selections and settings changes, but also enables rendering the probe and board positions in real time. This also means that several windows of the interface can be open at once. For example, if a user has multiple monitors, they can have the schematic open in full screen on one monitor and the layout in full screen on the other. We facilitate remote collaboration by using ngrok<sup>7</sup>, a tool that can temporarily exposes the localhost to the internet, connecting multiple computers with full selection cross-linking and augmentation synchronization. When collaborating, both screen and projection interfaces are mirrored. When any user probes or selects, the action is reflected across all users in the same session. As in an individual session, guided measurements and guided annotations can be authored via the measurement sidebar.

The interface consists of (1) a schematic and layout visualizer, (2) a search bar, and (3) a measurement sidebar (see Figure 5). We extend IBOM Visualizer [1], which renders and supports interactions with the PCB layout, including metadata such as pin 1 and DNP designations, by also displaying the schematic sheets. As preferred by electrical engineers [6], the schematic and layout sit side-by-side and take up the majority of the screen, but can be customized to the user’s liking. The layout is rendered in an HTML5 canvas using design file data in pcbdata.json from the import utility. For performance reasons, visuals that are frequently updated, such as selection highlights and probe locations, are drawn in a separate canvas layer above the layout. The schematic is rendered primarily through the SVGs from the import process. Only the selection highlights are rendered in an HTML5 canvas above the SVG, using design file data in schdata.json.

Board elements can be selected in both the schematic and the layout by either clicking on them directly, or selecting them in the search bar. In many cases, clicking on an element in the schematic or layout will hit several hitboxes at once, which is resolved by a popup disambiguation menu. Our system supports cross-linking between board representations, meaning any selection made in



**Figure 4: The workbench.** We used a motion capture system to track the pose of the board and test probes (A). A 1920-by-1080 resolution LED projector is used to project downwards on to the PCB (B). A screen in front of the user shows all the design files and settings (C). ARDW uses a connected benchtop digital multimeter to facilitate measurement (D). We use retroreflective markers to facilitate tracking of the board and test probes (E, F).

one representation will also appear on the others, including with augmentations on the physical board.

The measurement sidebar can be opened to use connected test instruments with ARDW and contains a handful of distinct functionalities. The record button at the top allows the user to change the augmented interface between selection mode and measurement mode. The DMM instrument panel imitates the display of a digital DMM, allowing users to change the mode of the DMM to voltage, resistance, or diode and see the current value being measured. Users can also create and run measurement sessions as part of a bring-up or free form debugging workflow.

## 4.3 Augmented Interface

The augmented interface is supported by three main subsystems: (1) the tracking system calculates the pose of both probes and the board in real time, (2) the projector display provides augmentations on the physical PCB and the surrounding workbench surface, (3) and the connected test instrument captures measurements. In the following subsections, we elaborate more on how these subsystems work.

**4.3.1 Tracking System.** We use an 8-camera optical motion capture<sup>8</sup> system (calibrated accuracy of 0.3 millimeters) mounted on an

<sup>6</sup><https://socket.io/>

<sup>7</sup><https://ngrok.com/>

<sup>8</sup>OptiTrack Prime 17W

aluminum frame to track the ground truth position and orientation of the board and the test probes as shown in Figure 4. To facilitate this, we place IR retroreflective markers on both the boards and the test probes. The markers for the test probes are placed on 3D-printed crowns which are affixed to the top of the test probes, ensuring they remain in sight of the cameras as the hand grasps the probe. The motion capture system reports the pose of the board's and probe's crowns in the motion capture coordinate frame at 30 Hz. The data is piped over UDP<sup>9</sup> to the server backend. To address the issue of model fitting noise, which was especially significant when the probe's crowns were close together, we applied an exponential low-pass filter to both probe position ( $\alpha=0.3$ ) and board position ( $\alpha=0.1$ ) [4].

We use a one time calibration procedure to establish the position of the probe's tip relative to the crown's pose. To determine this, we press the probe tip against a surface to keep it fixed in space, while performing a circular motion with the probe. We use least squares to fit a sphere to the crown's poses and assign the sphere center to be the tip position.

**4.3.2 Projector Display.** As debugging generally happens in a fixed bench-top location, we opted for projection-based AR. This allows for direct observation and manipulation of the board without an intervening display or screen, as in optical see-through augmented reality or mobile screen-overlay augmented reality, permitting for viewpoint independent augmentation simplifying render. We use a 1920-by-1080 resolution LED projector<sup>10</sup> to project downwards on to the PCB and white anti-static mat. The projector is mounted on an overhead aluminum frame approximately 0.6 meters over the surface, yielding a 48 by 27 cm display area as depicted in Figure 4. Positioning the board between the throw axis of the projector and the user generally avoids issues with users occluding the projection, except in cases where the user's hands are probing directly from above or the user places their head over the board.

We use a projector brightness of 1000 lumens, which was clearly visible in normal indoor office lighting. To establish the relationship between the tracking system coordinate frame and the projector's display, we project a 3-by-4 checkerboard and sequentially place the probe at each vertex, and use least squares to estimate the projector's projection transform. Since both our tracking system and board design files are in millimeters, we can scale the projection appropriately.

For the projected augmentations, the rendered content being projected is another front-end web page, similar to that of the screen interface, except that the only elements present are a black background and a multi-layer HTML5 canvas with the highlights. As with the screen interface, the projector display is connected via Socket.IO to the server backend to allow for real-time updates and facilitate remote collaboration. Augmentations of board elements, such as highlighting a specific pad, are generated in the same manner as the layout view of the screen interface, which has a 1:1 correspondence with the physical board when scaled correctly.

To facilitate alignment between PCB and the projected augmentations, we project all pads and edge cuts and allow the user to manually translate the board. If marker stickers are affixed to the

board, we also provide a function to snap the projection to the board automatically, or to track the board in real time as it moves around. An outlier filter addresses instability when the probe and tracking markers are within the same vicinity or when a user's hand occludes the tracking markers from the cameras.

**4.3.3 Connected Test Instrument.** To facilitate measurements, ARDWuses a connected benchtop digital multimeter (DMM)<sup>11</sup>. The server backend exposes DMM function selection via the screen interface's measurement panel, allowing the user to set the DMM mode (voltage, resistance, or diode), as well as mirroring the front panel's value. Mode selection and value querying is achieved with industry-standard SCPI commands via the VISA API over a USB 2.0 cable. By adhering to an industry standard command protocol, ARDW is easily extensible to accommodate additional DMM functions, different DMM models, or other types of test equipment, such as oscilloscopes, network analyzers, power supplies, and more.

**4.3.4 Interaction with the Augmented Interface.** The augmented interface has two modes of interaction: selection and measurement. In selection mode, the positive probe of the multimeter serves as the selection probe. To make a selection, the user dwells on the component or pad with the tip of the probe, which was the method of selection preferred by electrical engineers in Augmented Silkscreen [6]. To detect when a probe's tip is dwelling, the backend keeps a short history of the tip position in the last 0.5 seconds and checks if all points lie within a 5 millimeter diameter sphere. Once a dwell has been detected, the current position of the probe tip is projected onto the 2D board layout and processed like a click in the screen interface. To help users see and account for tracking imprecision, each probe has a small colored dot projected at the calculated location of the probe tip. Additionally, all hitboxes are padded by 1 millimeter in all directions, allowing users to select the desired element even if the probe is not recognized as being directly on it.

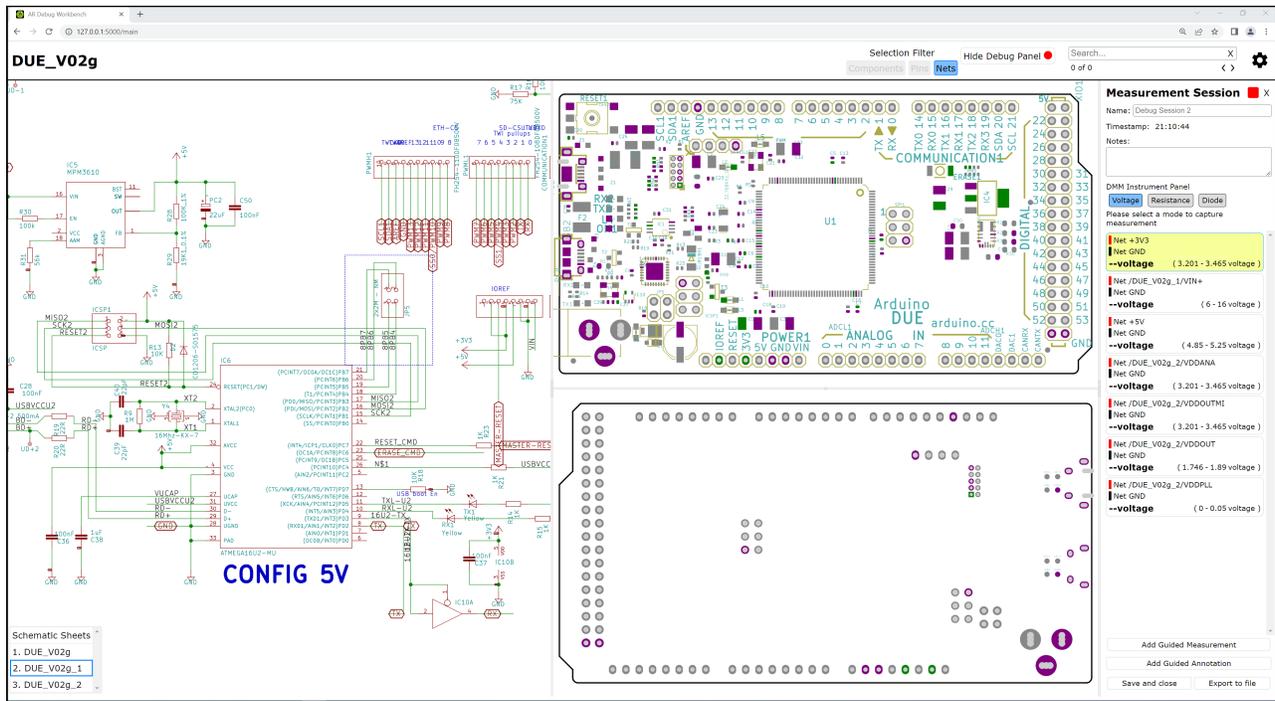
As with the screen interface, there is often a need for disambiguation. There are two sources of selection ambiguity. First, hitboxes of different types generally overlap; for example, a selection on the pad of a component lies within the hitbox of the component, the pin, and the net of the pin. Second, the 1 millimeter hitbox padding means nearby hitboxes are likely to also be hit. To address the first source, the system includes a probe selection filter. From the screen interface, the user can choose between selecting components, pins, nets, or any combination of the three. However, in many cases, manual disambiguation is still necessary. When a selection is ambiguous, a disambiguation menu appears next to the board that lists the reference designators of the possible selections. The user can then make a selection within the menu by tilting their probe forward or back to manipulate the menu selection cursor, whose position corresponds to yaw of the probe, and dwelling the cursor within the desired menu item.

To avoid the issue of constant re-selection with minor adjustments in probe position, the server backend keeps track of a safe zone around the edges of the board and just above the highest component. Outside of this safe zone, no hitscan is necessary. Inside of the safe zone, the probe can make a selection, but it cannot make

<sup>9</sup>User Datagram Protocol

<sup>10</sup>AXAA M7 Projector: [https://www.aaxatech.com/products/M7\\_pico\\_projector.html](https://www.aaxatech.com/products/M7_pico_projector.html)

<sup>11</sup>Keithley DMM6500



**Figure 5:** A screenshot of the screen interface. Schematic is rendered in split screen with front and back layout views on the right. On the far right the measurement panel is open. In this capture, the system is in guided measurement mode with a set of tests loaded and the first measurement queued for capture. In the layout view, the system has highlighted positive probe points in green and negative probe points in purple (which is mirrored on the physical PCB). The selection filter for probe hitboxes (top right) has been limited to nets in this scenario.

another one until it has been invalidated by leaving the safe zone. The result is an intuitive interaction: put the probe down on the board to make a selection, then lift the probe off the board and place it down again to make another.

Finally, unlike in the screen interface, deselection within the board is not supported. Instead, users can deselect by placing the probe tip down on the mat surface outside of the board. If the probe is merely set down, the probe tip rests several millimeters above the mat. To avoid a deselection in this case, the deselection zone extends only 1 millimeter above the mat.

In measurement mode, both the positive and negative multimeter probes are tracked and can select and disambiguate as in selection mode. When both probes have made a selection, a measurement is recorded from the connected test instrument and appears in the measurement sidebar of the screen interface. Unlike in selection mode, a probe is deselected as soon as it is invalidated by leaving the safe zone, rather than waiting for a deselection event, as selection in this mode is only for measuring specific elements.

Measurement mode also has additional disambiguation features. The selection filter works as before, except that ‘component’ is not available; an individual probe must select a pin or net. If a guided measurement has been specified in the screen interface, for example as part of a bring-up workflow, the system will try to automatically resolve ambiguous selections. If a probe makes an ambiguous selection that includes the expected pin or net from

the guided measurement, the system assumes that the probe has been correctly placed and selects the expected element. However, if the expected element is not in the selection, the user will need to disambiguate as normal to proceed with a different measurement from the guided one.

## 5 EVALUATION

To evaluate ARDW, we conducted a three-part user study with 10 electrical engineers. We derive the tasks within each part from common electrical engineering debugging workflows. In each section, we introduce new features of the system to the participant. In the design of the evaluation, we seek to strike a balance between structured usability testing with guided tasks and qualitative free-form exploration, to avoid the pitfalls premature usability testing can bring[9]. For qualitative feedback, we analyzed their responses via thematic analysis [5], first transcribing the interviews, then coding recurring themes, and finally noting outliers from the norm. In part 1, we examine ARDW’s effect on navigation between the board and design files. In part 2, we collect feedback on ARDW’s use for PCB bring-up. In part 3, we gather participant’s thoughts as they use ARDW in a set of free-form debugging tasks.

### 5.1 Participants and Procedure

We recruited 10 participants who hold electrical engineering roles in academic labs and/or industry. During our selection period, we

**Table 1: Recruited participant backgrounds and expertise.**

	Field	Experience	Primary Tool	Designs
P1	Industry	Design, Release, Functional check	Siemens PADS	Two-layer, large amplifier designs; High layer-count, high-speed PCBs such as PCIE and graphics cards
P2	Industry, Academia	Design, Release, Assembly, Functional check	Altium Designer, Eagle	Embedded firmware/hardware
P3	Industry	Design, Release, Functional check	Altium Designer, KiCAD	FPGA board; Embedded systems
P4	Industry, Hobby	Design, Release, Assembly, Functional check	KiCAD	Rework technician in industry; simple 2-layer sensing designs
P5	Academia	Design, Release, Functional check	KiCAD	Small, low power board for communication
P6	Industry	Design, Release, Engineering validation, Mass production	Altium Designer, KiCAD, Nexus	Small, high frequency signal board; Complex mixed signal
P7	Academia	Design, Release, Functional check	KiCAD	Small, single layer FPC for robotics
P8	Industry	Design, Release, Engineering validation, Mass production	Cadence, AutoCAD	FPC (2-3 layers, mixed signal); High-layer count, HDI motherboards with SoC, UFS, DDR memory; Power design
P9	Academia	Design, Release, Engineering validation	Altium Designer	Control board for robotic arm
P10	Academia	Design, Release, Assembly, Functional check	AutoCAD, DXF format	Sensing board for miniature robots

confirmed that all of our participants regularly design and debug PCB designs. Participants' reported design focuses spanned from small, low power boards to FPGAs and complex mixed signal boards. A comprehensive list of their experience can be found in Table 1. Each study took between 60 and 70 minutes for a total of 11 hours of feedback, and participants were compensated for their time. Prior to the study, we ran a pilot participant to determine default settings to use such as hitbox size at appropriate study length. In the study, board tracking was not enabled.

**5.1.1 Procedure of Task 1: Navigation.** We first introduced participants to the system by pointing out major interface components and demonstrating selection mode using the probe. To familiarize participants with selection mode, we presented a navigation task. Electrical engineers frequently navigate between their schematic, layout, and board during a debugging task[6]. Mirroring the evaluation in Augmented Silkscreen[6], Task 1 was split into two timed sub-tasks: (1a) Finding a component on the board given a target in the design files, and (1b) Finding a component in the design files given a target on the board. For Task 1a, a target component was highlighted in the design files of the Arduino Uno R3.<sup>12</sup> Participants selected the corresponding component on the Arduino board with their probe to indicate they found the component. A sound signalled success or failure. Participants could re-select if their first selection was incorrect. With augmented cross-linking enabled, the target component was highlighted on the board as well. In the control, only the schematic and layout were cross-linked as in standard ECAD tools. For Task 1b, a target component was highlighted on the board and the participant selected the corresponding component in the schematic or layout. For each sub-task, participants were presented with 20 randomized component selections: 10 with

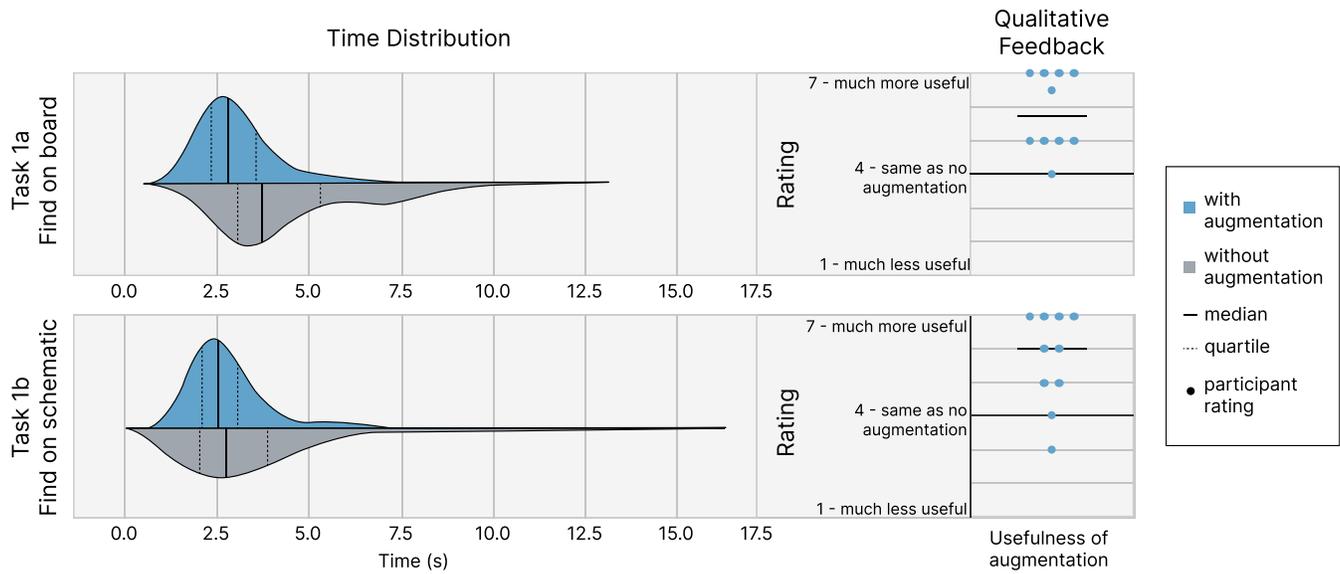
augmented cross-linking, and 10 without augmented cross-linking. The order of presented conditions was counterbalanced across participants. The component selection filter was enabled across all tasks and conditions. In addition to timing the tasks, we collected qualitative feedback, first confirming whether the task featured in their own workflow and then using prompting questions such as "Would you find ARDW to be helpful, not helpful, or have no effect on your workflow?" and "What aspects did you like and not like about using the system for this task?" Finally, we recorded Likert scores for the question "For this task, how useful would ARDW be in your workflow?"

**5.1.2 Procedure of Part 2: Bring Up.** Next, to introduce participants to measurement mode, the search bar, and multi-page schematics, participants performed a board bring-up task on the Arduino Due<sup>13</sup>. We loaded a set of seven voltage rail measurements into the measurement panel. Participants first stepped through the measurements without the augmented cross-linking as a basis of comparison. Then we enabled probe tracking and board augmentation in measurement mode, and participants repeated the same measurement procedure. We recorded their general qualitative feedback, prompting them with the same questions as mentioned in Task 1, and then recorded Likert scores for the questions "For this task, how easy was the procedure with and without ARDW?" and "What was your confidence in executing the task with and without ARDW?"

**5.1.3 Procedure of Part 3: Debugging.** Finally, we introduced the selection filter and demonstrated how to move between selection and measurement mode. Task 3 consisted of four free-form debugging tasks. We introduced errors into four different PCBs to represent a range of possible errors. For the Arduino Due, we soldered an incorrect feedback resistor in a buck network, causing the +5V rail

<sup>12</sup><https://store-usa.arduino.cc/products/arduino-uno-rev3>

<sup>13</sup><https://store.arduino.cc/products/arduino-due>



**Figure 6: The timing distribution and subjective scoring for Task 1a (Find on board) and Task 1b (Find on schematic). In Task 1a, participants performed faster with augmentation than without, while in Task 1b, participants performed similarly in both conditions.**

to sit high (a fault they discovered while performing bring-up in Task 2). For the Arduino Uno, a diode near the power input was placed in reverse polarity. For the Sparkfun RedBoard<sup>14</sup>, a mis-sized current-limiting resistor caused the power LED to be dimmer than expected. Finally, for the Sparkfun Sound Detector<sup>15</sup>, an incorrectly biased op-amp network resulted in a clipped audio stream (as visualized on an adjacent oscilloscope). Each participant debugged two of the boards with ARDW and two without augmented cross-linking with order balanced via Latin square. Because of the unconstrained nature of this task, we pursued a qualitative coding approach. We encouraged participants to think aloud [15], and then asked a set of questions covering their overall impression of the system, limitations, wish-list items, and practical considerations.

## 5.2 Findings

**5.2.1 Findings for Task 1: Navigation.** This initial task was the first time all participants interacted with the system. Participants' first impressions were that of excitement, especially after seeing the components first light up on the board in front of them.

The timing distribution and subjective scoring for Task 1a (*Find on Board*) and Task 1b (*Find in Design Files*) are presented in Figure 6. Shapiro-Wilk tests indicated all data was not normally distributed ( $p < 0.001$ ), so we used the Wilcoxon signed rank test to compare timings with and without augmentation. For Task 1a, participants performed faster with augmentation (Mdn=2.742) than without augmentation (Mdn=3.730). A Wilcoxon signed rank test indicated that this difference was statistically significant ( $p < 0.001$ , effect size  $r = 1.537 > 0.5$ , large effect). For Task 1b, participants had a similar

performance with augmentation (Mdn=2.436) as without augmentation (Mdn=2.790). A Wilcoxon signed rank test indicated that there was no statistically significant difference ( $p = 0.228$ , effect size  $r = 0.381 > 0.3$ , medium effect).

Qualitative rating indicated generally positive sentiments about the usefulness of system in board navigation tasks with ARDW-enabled augmentations compared to the baseline case in both tasks. The high ratings were generally driven by ARDW's ability to reduce context-switching. Nine out of 10 participants mentioned that interacting with the projected augmentation reduced their mental effort to select or locate components, with six of the 10 participants mentioning the projection made it faster to select components. Additionally, a few participants mentioned that ARDW reduces the chances to make mistakes. Five of our 10 participants mentioned that ARDW's cross-linking functionality reduced the need to divide attention across multiple representations or tools. "It highlights things [that I'm looking for] and I know where to go, so it really reduces the amount of time that I need to go back and forth, and dealing with a 3rd item [the mouse]" (P2, Q1). "Like, it would be nice to never have to pull up a board [layout] file, and then only click on a schematic and then, bam!, it shows me where it is on the board. You could eliminate half of the screen. That would be great" (P8, Q2).

Participants generally felt that the larger and denser the board, the more useful the cross-navigation representation would be. "This would be really helpful, I can see that for larger boards this is more helpful, and for smaller boards with smaller components it is extremely helpful. I have had cases with 0603 resistors, and in those cases, it would really help me to find those components" (P2, Q3). For larger, distinctive components the benefit of augmentation is

<sup>14</sup><https://www.sparkfun.com/products/13975>

<sup>15</sup><https://www.sparkfun.com/products/12642>

less realized. “I’ll use it all the time for smaller components, but for larger components, by eyes is faster” (P1, Q4).

All participants picked up the probe selection mechanism quickly in the practice rounds for Task 1. However, in observing the selection behaviors of our participants, we noticed a split between participants touching down on the component versus hovering a few millimeters above components to select. While for some, contact with the component confirmed they were selecting an item, others found it could sometimes slightly shift the board resulting in augmentation misalignment. For example, one participant mentioned “[from] my understanding you kind of want to not necessarily tip off the thing [and shift the board], but you want [the projected dot visible under the probe tip] to be on component... you don’t really have to touch it, right?” (P4, Q5). The board shifting from the augmented projection was the main source of frustration expressed by nearly all participants. This caused their selection to be less precise, slowing their ability to get into the component’s hitbox with the probe and decreasing their confidence in the system. “The only pain in the butt is just applying pressure to the board, the board rocks, so you’re naturally going to get misaligned” (P8, Q6). “Due to the not perfect alignment... I’m definitely towards less confident [because] I have to have the same double check as without augmentation” (P2, Q7). This was most evident for participant P3. A tracking camera had been bumped prior to their study, resulting in poor alignment between the probe’s reported and actual position. The probe tracking’s imprecision made tasks extremely tedious, causing frequent mis-selections (resulting in the lowest data point in both qualitative rating scales).

On the other hand, a number of participants appreciated the directness of interaction afforded by selecting with the probe. “I can go there [tap on the component] directly and get the link to the data sheet” (P2, Q8).

**5.2.2 Findings for Task 2: Bring Up.** Many of the same findings that we learned from Task 1 (Navigation) held true for this bring-up task as well.

In the bring-up scenario, the benefit of reduced context switching was even more noticeable. In the condition without augmentation, participants would typically have to reference the list of required measurements at least once and the design files at least twice for every single measurement to find appropriate probe points for their positive and negative probes. When a probe point proved too small or difficult to access, this back-and-forth multiplied as participants returned to the design files to identify another suitable point. With augmentation, participants’ area of interaction consistently remained on the board. While we did not explicitly record times for the task, as we asked participants to think aloud, they agreed that they were much faster with the augmented measurement. Participants generally found augmentation to make bring-up easier and were more confident in performing bring-up (see Fig. 7). “[It’s] much better with the AR assistant because one, it highlights where the net is with the color code, you don’t need to find net then pin, and two, voltages are automatically loaded into the spread sheet” (P1, Q9).

However, all participants were again challenged by imprecision in probing due to movement of the PCB, exacerbated by three factors: participants used both the positive and negative probes to

take measurements, applying force ensure good contact; the board was connected to power via its DC barrel jack and the wire could tug on the board; and the pins on the Arduino Due were a relatively smaller target than the components on the simpler Arduino Uno.

The smaller augmentation targets also created additional challenges on the precision of the projected highlight as well. Four out of 10 participants also mentioned that the highlight was not precise enough and that the highlights on multiple adjacent fine pitch pins blurred together, therefore reducing their confidence of what to probe for the corresponding measurement. “But yeah, definitely concerned with resolution, because I think most of the stuff that we do is either just denser or smaller” (P8, Q10). “This last one was so small it highlighted both of them” (P4, Q11). One participant mentioned that they would still prefer using a system that is slightly mis-calibrated over having to constantly switch between using probes and mouse. “The augmentation made it easier to probe, [but] when you are probing it, it does shift away. And [the projection] doesn’t move with it. That’s kind of annoying, but that’s still better than having to remove your hands from the probe to find it. So I think it is better in terms of efficiency” (P5, Q12). Additionally, because of the number of targets highlighted in the bring up task, especially with all the pins associated with the GND net, four participants indicated the board was overpopulated with highlights, making the probe points more difficult to discern because of the relative brightness and busyness of the projections. “It’s mostly just because there’s so many things being highlighted. Versus [when] I’m just trying to look for a component, it’s just highlighting one thing... dealing with smaller things it gets a little busy” (P2, Q13). “It messes with how I see it” (P9, Q14). These participants indicated that they would like the option to turn off the GND highlight since they would often solder a separate lead or have a dedicated test point. Following the study, we added a toggle to turn off the highlight for GND.

While most participants indicated that this feature would be useful in their own workflow, it is also worth noting that four of the 10 participants mentioned that this bring-up feature (and more generally the pre-specified measurement feature) is not useful because they do not perform bring-up tasks in a fixed sequence, instead performing their net selection on-the-fly. The split in opinion was divided by complexity of the design they usually worked with: these participants generally focused on miniature low-net count designs, while two other participants who work in industry on more complex designs acknowledged the benefits of having the measurements automatically recorded.

**5.2.3 Findings for Task 3: Free Form Debugging.** The process of debugging again required the participants to cross-reference between the layout/schematic and the PCB. We observed that the findings from the two previous tasks are a common thread through Task 3 as well. Notably, the benefits of ARDW in highlighting when locating the components were again resonated from all 10 participants in this task. Of which, five of them mentioned that it reduced the amount of effort required. For the Sound Detector board, the power cord caused the typical orientation of the board on the table to rotate from that of the layout. Three participants specifically mentioned that the augmentation was particularly helpful in this



new board. P10, who mentioned that “ease wise, and confidence... I think both are directly correlated”, gave a Likert score of 7 for both the questions on ease and confidence when using ARDW in Task 2: Bring Up.

Four out of 10 participants sought to further reduce context switching by integrating the system with automated hypothesis generation and guidance. “I almost wanted it to tell you what pin it is for me [instead of having to] look up and [specify] which pin on my program” (P8, Q21). “If [the system] can say hey, check this out, it’s lower than it’s supposed to be, that would be helpful, kind of warns you. Or you could go for the diode and [it might say], hey, is your [diode in] reverse? So that’d be helpful... helps you check against the schematic” (P4, Q22). Taking this thread further, future work could allow for ARDW to assist in physical computing environments that consist of both code and embedded hardware portions. As users step through code, sub-circuits related to that code could be augmented directly on the board. Paired with the ability to code step, this could make a compelling way to more directly pinpoint errors that cross code and hardware.

**6.1.2 Directness of Probe Interaction.** Participants appreciated the use of a probe as interactive tool, commenting that the directness of interaction on the board made for a more seamless navigation experience across representations. “It’d be really useful to just like, instead of looking back and forth trying to find a component, it’d be really easy to just click on it and it highlights [on the schematic] so I don’t have to search in the schematic because... often I use schematics way bigger than [Arduino Uno] and [it’s] really time consuming” (P4, Q23). “I definitely like the backwards from board into schematic better, because then if I’m looking something where the heck is this, and then I just [find] it right there. And that’s really useful” (P6, Q24).

Some suggested building out the augmented interface to allow for greater interactivity on the anti-static mat itself. **The user’s attention should generally remain at the point of interest, in this case the mat where the PCB is placed.** Through observation, users’ attention typically jumps between the schematic or layout view, the PCB, the digital multi-meter, and the measurement panel. By making use of the real estate that the mat has, more relevant information can be provided to the user, without them having to take their attention away from the PCB. P5 said: “My top dislike is having my attention in two different places. Or if the tension is split, like fifty-fifty and there’s more movement.” One participant expressed the desire for measurement information to be shown right next to the PCB. Another participant expressed that they would love to have an API to generate interactive graphics directly on the mat itself, so they could quickly change test modes by simply tapping on the mat. Future work could explore making the mat surface a more interactive element of the system.

**6.1.3 Screen Interface.** Apart from the augmented cross-probing, a few participants commented that they appreciated the utility of screen interface alone. They indicated that the cleanliness and smoothness of the interface (which is comparatively lightweight to the busyness of a typical ECAD tool) enabled a better experience. They also indicated that the interface encouraged them to cross-probe between schematic and layout, which was helpful for

debugging, something that traditional ECAD programs do not directly emphasize.

## 6.2 Limitations of ARDW and Future Work

**6.2.1 Precision and Clarity of Projected Augmentations.** While users expressed that the greatest benefit for such a system would be for large densely-packed boards, dense boards created the greatest challenge to augment precisely. **While some users appreciate augmentations that can direct to a general target area, greater precision in augmentation would unlock finer targets further reducing context switching.** While the highlights generally worked well for larger components and targets (0603 packages and up), smaller targets such as 0201 package components or dense IC pins proved challenging to highlight effectively due to imprecision of the highlight. While the projected pixel size is approximately 0.25 mm by 0.25 mm across the 43 cm by 27 cm area, the ability of our relatively budget projector’s optics to resolve those pixels was limited. Furthermore, users did not benefit from the large projected area as they debugging happened within one location in the mat. Future systems would benefit from concentrating resolution in a tighter space and from better-resolving projection optics to maintain sharpness. Future work could benefit from a focus-free projector such as a laser scanning projector which could help to maintain sharpness across the component z-heights. This could also allow for in-focus augmentation even if the board is held in mid-air.

While most augmentations were easily visible especially for larger components or diffusive IC packages, some highly specular components, like polished metal housings or glossy soldermask, resulted in changing augmentation clarity from different viewpoints. Additionally, the system is extremely sensitive to perturbations. A combination of the workbench being on wheels and the weighty projector being mounted on a cantilevered arm causes the projected augmentations to wobble a couple millimeters when the table is bumped. This made using the system a more delicate procedure. For future systems, ensuring rigidity of the projector mounting structure or high speed correction of projection would make the augmentation alignment more tolerant to vibration.

To our surprise, only three participants brought up blocking the projection as they looked to take a measurement. Generally, users approached probe targets from an angle to avoid occlusions. However, when the probe point required a vertical approach, participants expressed reservations: “Sometimes, like, you got to really come in... basically vertically and the way this is working from above kind of restricts that motion” (P6, Q25). The fact that the board sat between the user and the projector’s throw axis helped avoid occlusion to some extent since the light came from an angle beyond where the user’s body typically was rather than directly straight down on the board. However this introduced a new issue, where taller components’ projections were translated because the projection arrived off-axis. In future systems, this could be remediated by performing distortion correction relying on a 3-D model of the PCB or depth map to apply the correct shift for tall components.

**6.2.2 Precision of Probe Tracking.** Another major challenge for the usage of the system is the precision of the tracked probes in relation to the board. We find that **good alignment and calibration of**

### tracked probes is crucial for users to want to use the system.

In a number of instances, especially for smaller targets users would need to hunt for the hitbox or use probe cursor dot to adjust the position of the probe to achieve their desired selection. There were two sources of imprecision: (1) the physical board moved in relation to the virtual board due to bumping the board, and (2) the inherent system inaccuracy of the probe point due to mis-calibration of the tracking or the transform between projection and tracking.

To address these challenges, participants suggested a few methods. The simplest was to fix the board in place with an adjustable vice or putting the board on standoffs (some participants naturally did this by taping down the sound detector board in Task 3). Future work should explore this avenue as a low-hanging fruit to alleviate probing issues.

Another method would be to continually track and update the board position. We implemented continuous board tracking by applying retroreflective stickers to the board surface. Translational augmentation registration accuracy (combining board tracking accuracy and projection accuracy) is  $\pm 0.4\text{mm}$  MAE as implemented. At this accuracy, qualitatively, 0402 pads and 0201 packages are accurate, with 0201 pads and dense IC pins (e.g. on QFN packages) being close but imperfect (e.g. see 1:15 of demo video). For the user study, board tracking was not enabled as our pilot participants probed the board such that it did not tend to move (applying little lateral force on the PCB with friction from ESD mat). This turned out to be the exception amongst participants in the user study. We brought back two participants for an informal, post-hoc survey to compare their experience with board tracking enabled. Both participants provided strongly positive feedback indicating that it allowed for both more precise selections but also a more ergonomic and comfortable debugging experience, but confirmed it fell short of selecting the smallest targets such as IC pins. “So overall I definitely feel this is a lot more helpful, just because I trust here what it generate more than I trust myself when lining it up.” (P2, Q26) Future work could look into having the board sit on a tray with markers or placing the board markers on a crown connected to the board.

We found that users generally did not engage with the projected disambiguation menu list because of the additional friction in brought in trying to select a targets in a dense area. The list contained a set of textual names, but would be likely more effective with a spatial selection scheme to avoid the users from having to think about the type of element they’re looking to select. “Instead of just giving me the net names of the two pads, show me an enlarged display of the region under the probe and let me select the pad more visually. Having net names on this is still nice though” (P8, Q27). That being said, it is likely that better precision that can do away with the disambiguation menu entirely would be ideal.

**6.2.3 Other considerations. A lightweight set up process is required for electrical engineers to choose to employ this system over their current workflow.** The value the system brings during work must significantly exceed the effort to set it up. Regardless of whether the participant works at a fixed workbench or needs the system to be portable, minimum set up effort is required for users to choose to use ARDW. These factors include the physical set up, software set up (calibration of the projector, probes, and

tracking cameras), and setting up the design files. “If it could be movable from bench to bench...without much tweaking, that would make it way more practical and usable. Because I think anytime I have to grab something and be like, oh man, I’ve got to calibrate this thing, it just hinders me wanting to use it at all” (P8, Q28).

System cost is 33,390 USD, mainly driven by motion capture rig price (Optitrack cameras and equipment: 32k USD, projector: 700 USD, monitor: 300 USD, probes: 40 USD, desk: 200 USD, cage: 150 USD). From a cost viewpoint, participants agreed that the system should not significantly exceed the price of other professional-grade test bench equipment such as bench top DMMs and oscilloscopes (2.5k USD to 35k USD). Shedding the motion capture system for a cheaper RGB camera solution can help to make the system more practical. We are particularly encouraged by the recent work in PCB component segmentation via deep learning [18], and works exploring precise probe tracking with RGB cameras [22].

Lastly, while we did not get to test the remote collaboration capability due to the long study length, participant gave feedback that the the system would be beneficial for remote collaboration, especially with people who are new to the board or for non-electrical engineers who have to perform basic debugging. “We have engineers [in another office] but none of them are electrical engineers... but they deal with hardware. So they’ll get a board, they’ll boot it up, do some stuff and find out, stop working. But that’s the time where I need to help them debug over the phone. And they don’t have the technical know how to read a schematic where this would be wildly beneficial, if someone could just show them like where it was on this board. So I don’t have to pull up the assembly drawing myself or go through that process” (P8, Q29).

**6.2.4 Study Limitations.** Participants generally agreed that the tasks provided within the study were realistic and representative of the items they address in their own work. However, for users who focused on smaller, simpler boards, they found that bring-up task to be less applicable to their own workflow. For Task 3 (Debugging), we used the same tasks across all users. Therefore, the difficulty of a given task differed based on the participant’s own experience and concentration area. For this task, we collected only qualitative feedback due to unconstrained nature, but it would have been interesting to collect qualitative rating scores to (1) understand the difference in perceived benefit across differing experience levels and (2) quantify the magnitude of perceived benefit against the other tasks which were more directed. Finally, because of the length of the study, participants did not directly engage in a collaboration task. Future work should explore this area further.

## 7 CONCLUSION

In this paper, we build and test ARDW, a workbench that leverages projected augmented reality and tracked probes to assist electrical engineers in debugging PCBs. By enabling the ability to select and highlight elements across the schematic, layout, and physical board, ARDW reduces context-switching between board representations, allowing for users to more efficiently localize items on the PCB and capture measurements. Users appreciated the directness of interaction afforded the tracked selection and measurement probes and the performance of our in-browser application. We provide design considerations and recommend paths of inquiry for future systems,

including techniques to minimize imprecision of augmentation and to explore remote collaboration scenarios.

Users are excited to see further development in this area. When we mentioned compensation logistics after our study, a number of participants expressed that they signed up for the study just because it “sounded cool”, with one participant exclaiming “man, I wish to have something like this in my workplace to be honest!” By sharing our findings and open sourcing our code, we hope this work inspires further research into tools to support hobbyists and industry professionals alike.

## REFERENCES

- [1] [n.d.]. GitHub - openscopeproject/InteractiveHtmlBom: Interactive HTML BOM generation plugin for KiCad. <https://github.com/openscopeproject/InteractiveHtmlBom>
- [2] [n.d.]. inspectAR Augmented Reality PCB Tools. <https://www.inspectar.com/>
- [3] [n.d.]. Mascot | Robotas. <https://www.robotas.com/mascot/>
- [4] [n.d.]. pyrealtime/filter\_layers.py at master · ewhitmire/pyrealtime · GitHub. [https://github.com/ewhitmire/pyrealtime/blob/master/pyrealtime/filter\\_layers.py](https://github.com/ewhitmire/pyrealtime/blob/master/pyrealtime/filter_layers.py)
- [5] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2 (1 2006), 77–101. <https://doi.org/10.1191/1478088706qp0630a>
- [6] Ishan Chatterjee, Olga Khvan, Tadeusz Pforte, Richard Li, and Shwetak Patel. 2021. Augmented Silkscreen: Designing AR Interactions for Debugging Printed Circuit Boards. In *DIS 2021 - Proceedings of the 2021 ACM Designing Interactive Systems Conference: Nowhere and Everywhere*. <https://doi.org/10.1145/3461778.3462091>
- [7] Daniel Drew, Julie L. Newcomb, William McGrath, Filip Maksimovic, David Mellis, and Björn Hartmann. 2016. The Toastboard. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 677–686. <https://doi.org/10.1145/2984511.2984566>
- [8] Pragnu Goyal, Harshit Agrawal, Joseph A. Paradiso, and Pattie Maes. 2013. Board-Lab. In *Proceedings of the adjunct publication of the 26th annual ACM symposium on User interface software and technology - UIST '13 Adjunct*. ACM Press, New York, New York, USA, 19–20. <https://doi.org/10.1145/2508468.2514936>
- [9] Saul Greenberg and Bill Buxton. 2008. Usability Evaluation Considered Harmful (Some of the Time). (2008).
- [10] Jürgen Hahn, Bernd Ludwig, and Christian Wolff. 2015. Augmented reality-based training of the PCB assembly process. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*, Vol. 30–Novembe. ACM, New York, NY, USA, 395–399. <https://doi.org/10.1145/2836041.2841215>
- [11] Hamraz Javaheri, Olaf Pfeiffer, and Paul Lukowicz. 2018. HolOsci. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. ACM, New York, NY, USA, 1006–1010. <https://doi.org/10.1145/3267305.3274125>
- [12] Mitchell Karchemsky, J D Zamfirescu-Pereira, Kuan-Ju Wu, François Guimbretière, and Bjoern Hartmann. 2019. Heimdall: A Remotely Controlled Inspection Workbench For Debugging Microcontroller Projects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3290605.3300728>
- [13] Rushil Khurana and Steve Hodges. 2020. Beyond the Prototype: Understanding the Challenge of Scaling Hardware Device Production. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–11. <https://doi.org/10.1145/3313831.3376761>
- [14] Yoonji Kim, Youngkyung Choi, Hyein Lee, Geehyuk Lee, and Andrea Bianchi. 2019. VirtualComponent. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–13. <https://doi.org/10.1145/3290605.3300407>
- [15] Clayton Lewis. 1982. Using the “thinking Aloud” Method in Cognitive Interface Design. *IBM Research Report, RC-9265 9265* (1982).
- [16] Richard Lin, Rohit Ramesh, Antonio Iannopollo, Alberto Sangiovanni Vincentelli, Prabal Dutta, Elad Alon, and Björn Hartmann. 2019. Beyond Schematic Capture Meaningful Abstractions for Better Electronics Design Tools. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (2019). <https://doi.org/10.1145/3290605>
- [17] Yoichi Ochiai. 2014. Visible Breadboard: System for Dynamic, Programmable, and Tangible Circuit Prototyping with Visible Electricity. In *Virtual, Augmented and Mixed Reality. Applications of Virtual and Augmented Reality. VAMR 2014. Lecture Notes in Computer Science*. Vol. 8526 LNCS. Springer, Cham, 73–84. [https://doi.org/10.1007/978-3-319-07464-1\\_7](https://doi.org/10.1007/978-3-319-07464-1_7)
- [18] Abinai Pasunuri, Nathan Jessurun, Olivia P. Paradis, and Navid Asadizanjani. 2022. A Comparison of Neural Networks for PCB Component Segmentation. (2 2022), 113–123. <https://doi.org/10.1109/HOST49136.2021.9702286>
- [19] Evan Strasnick, Maneesh Agrawala, and Sean Follmer. 2017. Scanalog: Interactive Design and Debugging of Analog Circuits with Programmable Hardware. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 321–330. <https://doi.org/10.1145/3126594.3126618>
- [20] Evan Strasnick, Maneesh Agrawala, and Sean Follmer. 2021. Coupling Simulation and Hardware for Interactive Circuit Debugging. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA. <https://doi.org/10.1145/3411764.3445422>
- [21] Evan Strasnick, Sean Follmer, and Maneesh Agrawala. 2019. Pinpoint. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–11. <https://doi.org/10.1145/3290605.3300278>
- [22] Po-Chen Wu, Robert Wang, Kenrick Kin, Christopher Twigg, Shangchen Han, Ming-Hsuan Yang, and Shao-Yi Chien. [n.d.]. DodecaPen: Accurate 6DoF Tracking of a Passive Stylus. ([n. d.]). <https://doi.org/10.1145/3126594.3126664>
- [23] Te-Yen Wu, Jun Gong, Teddy Seyed, and Xing-Dong Yang. 2019. Proxino. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 121–132. <https://doi.org/10.1145/3332165.3347938>
- [24] Te-Yen Wu, Hao-Ping Shen, Yu-Chian Wu, Yu-An Chen, Pin-Sung Ku, Ming-Wei Hsu, Jun-You Liu, Yu-Chih Lin, and Mike Y Chen. 2017. CurrentViz: Sensing and Visualizing Electric Current Flows of Breadboarded Circuits. (2017). <https://doi.org/10.1145/3126594.3126646>