# Voltage Control System for Liquid-Crystal Based Reconfigurable Intelligent Surfaces

**Nazar Vasyliev**
**Supervisors: Arash Asadi, Fabian Portner**
EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Nazar Vasyliev
Final project course: CSE3000 Research Project
Thesis committee: Arash Asadi, Fabian Portner, Ranga Rao Venkatesha Prasad

**Abstract**

Liquid-crystal reconfigurable intelligent surfaces (LC-RIS) need hundreds of stable bias voltages, yet most existing controllers are slow, expensive, or hard to scale. This thesis prototypes a lean 64-channel driver that combines one 1 kHz NE555 square-wave source with per-channel amplitude control via daisy-chained AD5263 digital potentiometers and OPA4197 buffers. Each channel supplies a continuous $\pm 10$ V swing in 256 steps, and a Raspberry Pi Pico streams updates in 31 $\mu$s, fast enough not to limit the LC's millisecond-scale response. The work is still a proof of concept: it has been validated only with 64 channels on the bench, long-term drift and true large-panel scaling remain open questions. Nevertheless, the prototype points toward a practical, low-cost path for future LC-RIS control.

# 1 Introduction

Over the last decade, cellular networks have pushed into millimetre-wave (mmWave) frequencies to meet ever-growing data demands. Unfortunately, mmWave signals behave more like light than the familiar sub-6 GHz waves - they struggle to bend around corners and are easily blocked by walls, furniture, and even the human body. The result is a patchwork of signal "dead zones" that can only be fixed today by installing many costly and power-hungry base stations.

**Reconfigurable Intelligent Surfaces (RIS)** are emerging as a new way to shape the wireless propagation environment rather than merely adapting to it [14]. An RIS comprises a thin, planar array of unit cells, whose reflection coefficients can be programmed electronically [5]. RIS elements can be tuned in real time to reroute radio waves and achieve beam steering or focusing, all without active amplification or additional energy consumed compared to conventional wireless systems [14]. For example, RIS deployments can enlarge coverage [4], mitigate interference [14], and enable high-accuracy localization and sensing functions [13], all while adding almost no additional power consumption [14]. Thanks to these capabilities, RISs are envisioned as a cornerstone of energy-efficient 6G wireless networks [5].

To build a reconfigurable surface, each element must be controllable in how it reflects an incoming wave. In essence, an RIS acts like a "smart mirror" composed of many tiny adjustable reflectors. Each element induces a tunable phase shift (and possibly amplitude change) on the incident signal, analogous to a pixel in a reflective display that can be set to redirect light in a controlled manner. By coordinating thousands of such elements, the RIS can shape the wavefront - for instance, reflecting a planar wave toward a target receiver in the same way a phased array antenna or a mirror mosaic can direct a beam of light. Crucially, this is achieved passively: the elements simply scatter the existing signals with a modified phase, rather than actively retransmitting new energy [6]. This passive operation underpins the ultra-low power nature of RIS technology.

**Competing hardware technologies.** A variety of hardware implementations for RIS elements have been explored, each with different trade-offs in tunability, speed, power, and scalability. Common approaches include using voltage-tunable semiconductor devices (varactor or PIN diodes), Micro-Electro-Mechanical Systems (MEMS) switches/mirrors, or tunable materials like liquid crystals [5]:

- *Varactor diodes* provide continuous phase control with nanosecond switching, but their junction capacitance limits high-frequency efficiency and increases loss [5].

- *PIN diodes* offer ultra-fast reconfiguration (nanosecond switching), but only discrete phase levels per diode (multiple diodes per element are needed for multi-bit control). They also draw bias currents, and typically, each element's bias network consumes moderate power [5].
- *MEMS switches/mirrors* use micro-scale mechanical movement to achieve reconfigurability. In MEMS switches, a tiny movable capacitor or cantilever toggles the circuit state, whereas MEMS mirrors physically move a reflective surface or patch. These approaches can operate at high frequencies (mmWave to terahertz) due to the very small feature sizes, and they exhibit low RF (Radio Frequency) loss when engaged [5].
- *Liquid crystal (LC)* is an electrically tunable material widely used in display screens, and it has recently been adopted for RIS designs. LC-based RIS elements achieve continuous phase tuning by applying a voltage across a cell filled with liquid crystal, which changes the orientation of the LC molecules and thus the effective permittivity "seen" by the RF wave [6]. Unlike diodes or MEMS, the LC itself is a passive dielectric layer that draws virtually no current - only a bias voltage needs to be maintained, and each element's power consumption is extremely low [5]. LC-RIS designs are also highly scalable and cost-effective because they can leverage mature Liquid Crystal Display (LCD) fabrication techniques: the tunable elements can be built on glass substrates with thin-film transistor (TFT) backplanes to address many elements in a matrix, much as millions of pixels are addressed in an LCD panel. This makes it feasible to integrate thousands or even millions of tunable elements on a single surface. However, a key limitation of LC technology is its slower switching speed - reorienting the LC molecules typically takes on the order of milliseconds or more, which is orders of magnitude slower than semiconductor or MEMS alternatives [5].

LC-RIS advantages include continuous tunability, low power operation, and potential for large-scale integration. Notably, the same LC technology that enables modern LCD screens to control millions of pixels at video frame rates (tens of milliseconds per update) has the potential to drive large RIS arrays. We aim to develop a scalable, low-latency control system that can harness the full potential of LC-based RIS hardware. In particular, the goal is to design a hardware-software architecture capable of delivering high-resolution, high-speed bias control to large numbers of LC elements simultaneously [4, 6].

**Research questions.**

*What is a scalable, low-latency hardware-software solution for controlling LC-RIS systems?*

1. **Hardware selection** - Which driver can serve $\geq 50$ channels, each reaching at least $\pm 10$ V (ideally $\pm 15$ V) with $\geq 7$-bit resolution?
2. **Waveform generation** - How do we synthesise clean, synchronised 1 kHz (AC) bias waveforms on all channels with minimal skew and jitter?
3. **Latency** - Which hardware-software stack ensures the delay from command to voltage change stays below 1 ms?
4. **Scalability** - How can we preserve those guarantees as the channel count scales to hundreds or thousands?

5. **Validation** - Which experiments and metrics convincingly demonstrate that the controller meets these goals?

# 2 Control Systems for LC-RIS

Existing LC-RIS prototypes in the literature reveal limitations in their control systems, particularly in terms of scalability, latency, cost, and overall architecture. Current designs typically use a centralised controller (often a microcontroller or FPGA) linked to each tunable element through drive circuits, but the way this is implemented imposes serious performance bottlenecks for large-scale surfaces. Both hardware and software aspects of these control systems constrain the reconfiguration speed and practicality of RIS, indicating that more advanced architectures are needed for next-generation, scalable LC-RIS.

**Scalability and Channel Count:** A fundamental challenge is scaling the control hardware to hundreds or thousands of LC-RIS elements. In conventional designs, each element (or group of elements) requires a control line or analog bias. To reduce the explosion of wiring, researchers have resorted to multiplexing and matrix-addressing schemes. For example, one $10\times10$ RIS board uses a microcontroller with a grid of row/column address lines, requiring only 20 connections instead of 100 direct lines [8]. This bus architecture significantly cuts down wiring and is nominally scalable, but it comes at the cost of sequentially updating one element at a time. In that 100-element prototype, programming each unit cell took about 0.35 ms, amounting to 35 ms to refresh the entire array, even at this modest scale [8]. Such latencies would grow linearly with element count under a similar scheme, quickly reaching impractical levels for larger panels. A more parallel approach was demonstrated in a $16\times16$ RIS with binary-phase PIN diodes, where an FPGA and cascaded shift registers broadcast new states to all 256 elements quasi-simultaneously [12]. As the number of elements N grows, the sheer amount of control data and the complexity of fan-out drive circuits become a bottleneck. Clearly, the lack of a truly scalable channel count is a major shortcoming of current control systems. A practical LC-RIS may require thousands of independently tunable elements, which is far beyond the channel counts handled so far. Providing each element with a dedicated high-precision analog voltage in a cost-effective manner remains unsolved with existing architectures.

**Latency and Reconfiguration Speed:** Even if a control system can theoretically support many channels, it may not update them fast enough for advanced applications. Liquid crystal elements themselves are relatively slow to respond (on the order of tens of milliseconds to seconds for large tuning ranges). Recent work achieved a 72 ms response with an ultrathin LC layer, but this is still five orders of magnitude slower than semiconductor alternatives like MEMS or PIN diodes, which can switch in microseconds [6]. This intrinsic latency of LC technology means the control system must introduce minimal additional delay.

**Architecture and Cost:** The above limitations stem from underlying architectural choices. Hardware-wise, today's RIS controllers are largely adapted from lab equipment or small-scale prototypes, rather than designed holistically for massive arrays. They often consist of off-the-shelf microcontrollers or FPGA boards plus many discrete driver chips (shift registers, DACs, op-amps, etc.). This modular approach is flexible for initial research, but it does not scale to high density at low cost.

**In summary**, the limitations of current LC-RIS control systems in scalability, latency, and architecture motivate the development of a new, advanced control system. Such a system should overcome the bottlenecks by supporting a far higher analog/digital channel

count with fine voltage control, delivering updates to all elements with minimal latency and proper synchronisation, and doing so with a hardware architecture optimised for low cost per element. Only by addressing these gaps, such as increasing the number of controllable channels, speeding up reconfiguration, and leveraging a more integrated, scalable architecture, can future LC-RIS deployments fulfil their promise as large, continuously tunable metasurfaces for beyond-5G communications. Our contribution is a low-cost multichannel platform paired with a real-time software stack that meets these requirements.

# 3 Experimental Setup

## 3.1 Hardware Selection and Design

To meet the ±10V output swing requirement, high-resolution control, and low-latency response, we carefully selected each hardware component of the 64-channel LC-RIS control platform. Key components include an analog square-wave generator, digitally controlled attenuators, buffering amplifiers, and control interfaces. The rationale for each selection is as follows:

- **LTM8049 (±12V Rail Generation)**: To create the isolated ±12V rails that power the analogue section, we add the Analog Devices LTM8049. This power-module integrates the controller, switches, inductors and compensation into a 15 mm × 9 mm package, needing only a few capacitors and resistors externally. These rails directly power every OPA4197 and also serve as the input to the secondary ±5V regulators. By generating both polarities in one module, the LTM8049 guarantees matched dynamic performance and simplifies layout compared with two separate converters [2].
- **LT3032-5 (±5V Rail Generation)**: Clean ±5V rails are critical for the AD5263 digipots and the NE555 timing core, where supply noise would translate into amplitude or jitter errors. The LT3032-5 furnishes a fixed +5 V and -5 V output, each capable of 150 mA, with only 0.3 V dropout from the incoming ±12V rails [3].
- **NE555 Timer (1 kHz Square-Wave Generator)**: We chose the NE555 timer IC to generate a stable ∼1 kHz square wave as the common drive signal for all channels. The NE555 offers a simple, low-jitter oscillation source and can be powered to produce a ±5V peak output ($10V_{pp}$) with a symmetric duty cycle. This guarantees that each RIS element receives an identical frequency and phase drive, which is crucial for uniform control. The NE555's simplicity and reliability make it well-suited for this task, and its output amplitude can be adjusted via supply rails to meet the ±5V drive needed [10].
- **AD5263 Digital Potentiometers (Amplitude Control)**: Each channel's square-wave amplitude is digitally controlled using the AD5263, a quad 256-position digital potentiometer. The AD5263 was selected for its combination of high resolution (8-bit, 256 steps) and high-voltage handling. It supports dual ±5V supplies, meaning it can directly handle the ±5V AC waveform swing required for the LC cells. Using the AD5263 as a voltage divider allows us to attenuate the 1 kHz waveform from ∼0 up to full amplitude with fine granularity. Each chip provides four channels in one package, reducing board footprint and ensuring channel-to-channel consistency. Multiple AD5263s are daisy-chained on a 3-wire SPI bus, so all 64 channels (16 chips × 4 channels) can be updated through a common data stream. This daisy-chain SPI

configuration minimises the number of control lines and enables simultaneous updating of many channels with a single burst of clocked data [1].

- **OPA4197 Quad Op Amps (Buffer Amplifiers)**: Following each digipot, an OPA4197 operational amplifier is used as a buffer and output driver. The OPA4197 is a precision quad op amp that supports a wide supply range up to ±18V. This allows us to power the op amp at ±12V, comfortably achieving the desired ±5V output swing with headroom to spare. The OPA4197 was chosen for its ability to drive the load presented by the RIS elements and maintain signal integrity. These features ensure the 1 kHz square wave edges remain sharp and undistorted even after amplitude modulation. Additionally, the op amp's low output impedance and high output current (up to ±65mA sourcing/sinking) mean each channel can drive the LC-RIS element reliably without crosstalk or amplitude droop. The rail-to-rail output capability further guarantees the full voltage span is delivered to the load. In summary, the OPA4197 provides the necessary voltage range, drive strength, and precision to serve as the final stage amplifier for each channel [9].

- **Control Microcontroller (Raspberry Pi Pico 2)**: For initial prototyping, a Raspberry Pi Pico was used to orchestrate the system. The Pico's microcontroller provides flexible I/O (SPI/GPIO) and dual-core processing, which allowed testing of various control schemes. We leveraged its SPI hardware interface to achieve fast serial communication with the 16 daisy-chained AD5263 devices. The Pico was programmed with a high-level interface to set channel amplitudes and verify timing. It was chosen due to its ease of use, low cost, and adequate performance for driving 1 kHz updates [7].

Overall, the hardware is designed to achieve ±10V analog output per channel, 7-bit resolution amplitude control, and millisecond-scale update latency. The use of analog signal generation combined with digital amplitude control provides a robust and scalable solution: the single oscillator guarantees coherent drive across all channels, and the digital potentiometers with buffers allow independent amplitude tuning with minimal interference and fast response.

## 3.2 Software Architecture

**Low-level control layer.** Beneath the possible user-facing API, we implement a performance-critical control layer in MicroPython, responsible for direct hardware interactions such as Serial Peripheral Interface (SPI) transfers, data-frame marshalling, and microsecond-level timing. Early profiling showed that a minimalist, procedural style outperformed an object-oriented design: class-based wrappers added more latency on the Raspberry Pi Pico microcontroller.

The host communicates with the control layer either (i) via direct General-Purpose Input/Output (GPIO) when the Pico is connected over USB, or (ii) through a wireless link when using the Pico W / Pico 2 W. In both scenarios, updates are batched to amortise transaction overhead.

**Architectural takeaway.** This separation cleanly isolates high-level application logic from deterministic, real-time control tasks, yielding both maintainability and predictable timing.

## 3.3 Control Algorithms and Timing Logic

The control logic for the LC-RIS platform is designed to efficiently manage waveform generation and amplitude modulation across all channels. A fundamental design choice is that waveform generation is entirely offloaded to analog circuits (the NE555 oscillator and analog distribution network), while the digital control adjusts the amplitude of that waveform per channel. This division of work greatly simplifies the control algorithm: the system does not need to compute or store waveform samples, it only sets the desired amplitude level (as a digital code for the potentiometer) for each channel's analog path.

Each channel's amplitude can be updated on-the-fly by writing a new 8-bit value to the corresponding digipot (0 = minimum/grounded, 255 = maximum amplitude, with a roughly linear attenuation in between). The update logic ensures that writing new values does not disrupt the ongoing waveform beyond the intended amplitude change. Thanks to the buffer op amp, the output transitions cleanly from old to new amplitude within a fraction of the 1-ms period. If multiple channels need updating simultaneously, the control algorithm coordinates a synchronised update: the software packages all the new values into one SPI sequence so that all digipots are updated in quick succession. In the ideal case, the SPI update is so fast ($<$0.2 ms for all channels) that effectively all outputs attain their new amplitudes within the same waveform cycle.

Overall, the control logic is event-driven: when a new setting command is received, it triggers the hardware update sequence promptly. The worst-case latency from command to output update is kept below 1 ms by design. This includes communication, digital potentiometer settling (which is virtually instantaneous in electronic terms, on the order of microseconds), and op amp output settling. By keeping the waveform generation analog and continuous, we avoid any overhead of real-time waveform computation, allowing all processing budget to go into fast I/O updates. The result is a control system that can rapidly modulate 64 channels in parallel with minimal delay, fulfilling the real-time requirements of dynamic RIS operation.

## 3.4 Measurement and Validation Plan

A comprehensive measurement plan is in place to evaluate the platform's performance against the design criteria. The key metrics of interest are update time, amplitude accuracy, waveform integrity, and overall scalability of the system. We will use a laboratory oscilloscope and to gather precise data:

- **Reconfiguration Measurement**: Update time is defined as the interval between receiving a command on the Raspberry Pi Pico (to swing one or more channel amplitudes) and the moment the analog output settles at the new level. In the revised setup, the wave master line is powered by a steady 5 V supply rather than a square wave. A Raspberry Pi Pico simultaneously toggles a GPIO line high as it sends each SPI update, then drives every channel from full-scale high to full-scale low as quickly as possible. An oscilloscope records the repetition rate of these transitions, providing a measure of the maximum update frequency and the update time (half-period of the wave).

- **Amplitude Accuracy and Resolution**: To ensure each channel's output accurately reflects the commanded amplitude, we will use the oscilloscope to measure the peak-to-peak voltage of the square wave at various digipot settings. By sweeping a single channel's code from 0 to 255 (or a set of representative values), we can record the

resulting output amplitudes. These will be compared to the theoretical expected values (assuming ideal linear scaling). We expect a near-linear relationship, but any deviation (e.g., due to digipot resistor tolerance or op amp saturation at extremes) will be quantified.

- **Waveform Integrity and Stability**: Using the oscilloscope, we will examine the quality of the square wave across different amplitude settings and channels. Key aspects include the duty cycle of low and high periods, the flatness of the high and low levels, and the absence of overshoot or ringing. The NE555-based generator's frequency stability will also be measured over time - the frequency should remain ∼1 kHz with minimal drift.

  From [10]: $T = t_H + t_L = 0.693\,R_1C_1 + 0.693\,R_1C_1 = 1.386\,R_1C_1$;
  $f = \frac{1}{T} \approx \frac{0.721}{R_1C_1}$; $D = \frac{t_H}{T} = 0.5$; where $T$ is the period, $f$ the frequency, and $D$ the duty cycle.

- **Scalability and Throughput**: Finally, we will assess how the system performs as the number of channels and update frequency scale up. Although our prototype has 64 channels, the design could be extended by adding more digipot chips in the SPI chain or by using multiple parallel SPI buses. We will measure the update time for the 64 channels, and then calculate a higher channel count scenario by increasing the amount of data transferred per update, as the time will rise linearly.

# 4 Results

The implemented voltage control system successfully met most of the design criteria, as demonstrated by a series of measurements evaluating latency, accuracy, waveform integrity, and scalability.

## 4.1 Power Supply Performance

Figure 1 shows the simulation results for the LT3032-5 and LTM8049 power rails: -5 V, +5 V, -12 V, +12 V. The simulation results are precisely -5 V, +5 V, -12 V, +12 V.

The custom ±12 V and ±5 V rails remained mostly stable at the expected voltages, validating the power design. As we can see in Table 1, under no-load conditions, the rails measured very close to their nominal outputs (e.g. +5.04 V and -5.02 V for ±5 V). Under full load, a drop was observed on the +5 V rail after warm-up (down to +4.15 V) due to the on-board regulator hitting its 150 mA current limit. This was resolved by switching to an external 5 V supply, which restored the rails to ≈ ±5V even at maximum load. The ±12 V rails remained rock-steady around ±12.1 V in all cases, indicating the LTM8049 maintained the required output. These results confirm that the power delivery network could provide the necessary voltages without a significant drop, if we handle the 150 mA current limit for LT3032-5 (by decreasing the consumption from the NE555 or changing LT3032-5 to a more powerful part).

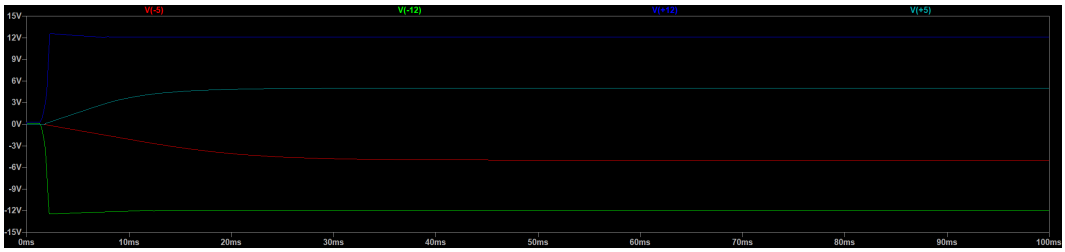



Figure 1: LTSpice simulation results for the LT3032-5 and LTM8049 power rails: -5 V, +5 V, -12 V, +12 V

Table 1: Power-rail voltages under various operating conditions

| **Condition** | $V_{-5}$ **(V)** | $V_{+5}$ **(V)** | $V_{-12}$ **(V)** | $V_{+12}$ **(V)** | **External supply** |
|---|---|---|---|---|---|
| Expected | -5.00 | +5.00 | -12.00 | +12.00 | - |
| No load (cold) | -5.02 | +5.04 | -12.10 | +12.10 | +12 V 42 mA |
| Full load, cold | -5.04 | +5.06 | -12.10 | +12.10 | +12 V 387 mA |
| Full load, hot | -5.02 | +4.15 | -12.20 | +12.10 | +12 V 335 mA |
| Full load, ext. +5, cold | -4.99 | +4.94 | -12.10 | +12.10 | +12 V 226 mA; +5 V 125 mA |
| Full load, ext. +5, hot | -5.00 | +4.94 | -12.10 | +12.10 | +12 V 228 mA; +5 V 123 mA |

## 4.2 Waveform integrity and stability

The NE555-based master oscillator produced a clean ∼1 kHz square wave that drives all channels. Simulation had predicted an ideal 1 kHz, 50 % duty cycle swing of ±5 V, and the measurements showed the real oscillator performing well within acceptable bounds. Figure 3a shows that at cold startup with the internal 5 V supply, the output waveform was 826 Hz with a 58.7% high duty cycle and 41.3% low duty cycle, and amplitude levels of about -5.2 V (low) to +4.88 V (high). Figure 3b shows that after the system warmed up, the frequency remained stable at 826 Hz, while the high level dropped to +4.08 V due to the aforementioned +5 V rail sag. Upon using the external 5 V source, Figure 3d shows that the cold system ran at 820 Hz with a 58.2/41.8% duty cycle and full ±5 V swing (measured -5.28 V low, +5.39 V high), and with almost no difference the warmed up system, as shown in Figure 3d, ran at 833 Hz with a 58.33/41.67% duty cycle and full ±5 V swing (measured -5.27 V low, +5.41 V high) In all cases, the waveforms were stable in frequency and had nearly flat tops and bottoms with no significant overshoot or ringing noted. The slight amplitude variation under different supply conditions reflects the load on the NE555 and regulator, but with a stable supply, these effects disappeared. The shifted duty cycle and frequency are probably a result of RC tolerance (the values of resistors and capacitors may differ by 10% from the expected). Overall, the master square-wave maintained its integrity across conditions, providing a reliable AC drive reference for all channels.

Figure 2 shows the simulated waves from the NE555. It is perfect 1 kHz, 50% duty cycle, ±5 V square waves.

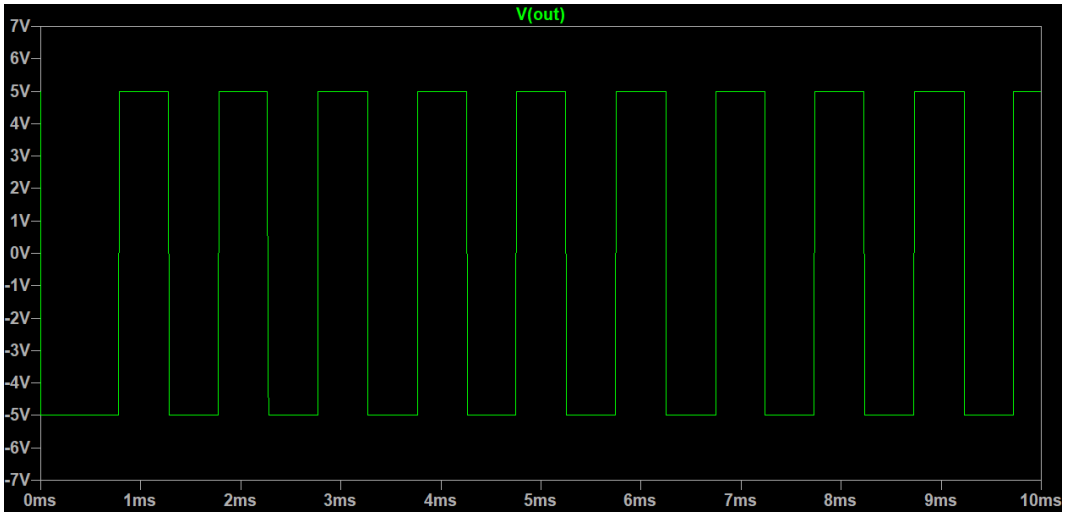
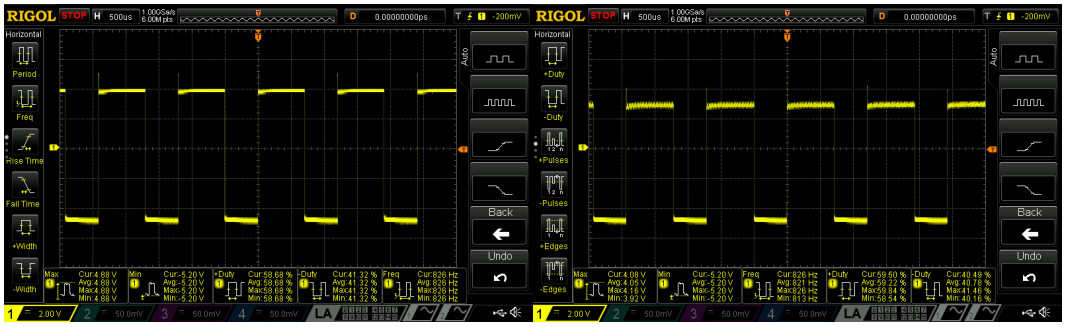


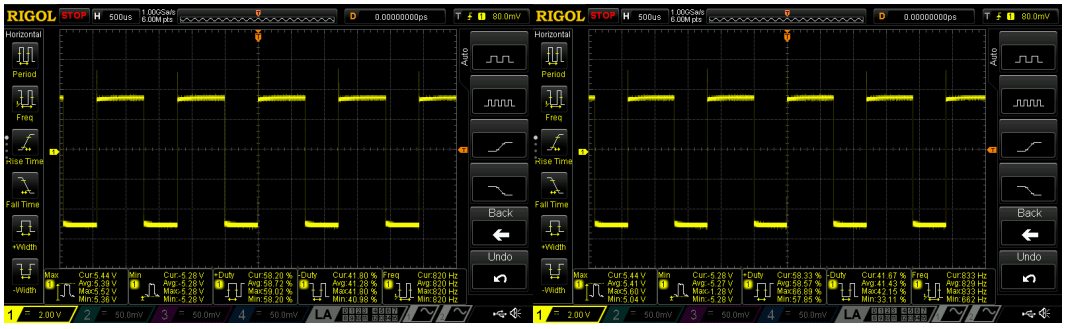

Figure 2: LTSpice simulation results for the NE555 wave master rail.



(a) Cold with internal +5 V



(b) Hot with internal +5 V



(c) Cold with external +5 V



(d) Hot with external +5 V

Figure 3: Overview of the master wave measurements under different scenarios. (a) Cold with internal +5 V; (b) Hot with internal +5 V; (c) Cold with external +5 V; (d) Hot with external +5 V.

### 4.3 Amplitude accuracy and resolution

The per-channel amplitude control proved to be highly linear and precise across the 8-bit range. At the maximum digipot code (255), channels output approximately ±10.5V peak-to-peak, closely approaching the design goal of ±10V. For example, in the Figure 5a Channel 3 at full scale produced a square wave swinging from -10.7 V to +10.4 V, and a similar range was seen on Channel 12 at the same setting (Figure 5b). This confirms the amplifiers (powered at ±12 V) can comfortably drive the ±10 V output range. At a mid-scale setting (128), in the Figure 5c, the output amplitude was roughly half of full-scale: Channel 11 showed about -5.86 V to +4.72 V swing for code 128, which is near the expected ∼50% amplitude level. At the minimum setting (0), the output was essentially zero-biased as intended - the measured waveform in Figure 5d of Channel 3 at code 0 had an average of about -16 mV (effectively 0 V) with only a tiny ∼±68V residual ripple. These results indicate an almost linear mapping between the digital control value and the analog amplitude. Any deviation from ideal linearity (for instance, the high output being ∼10.4 V instead of exactly 10 V) is minor and can be attributed to component tolerances (digipot resistor ratios, op amp saturation limits at the rails, etc.) and imperfections from the wave master rail. Crucially, the consistency across channels is high - multiple channels at the same setting yielded very similar voltages (e.g. two different channels both gave ∼ ±10.5 V at full scale), demonstrating uniform behaviour of all 64 channels. The 256-step resolution (0-255 codes) allows fine-grained control; the smallest amplitude increment is on the order of a few tens of millivolts, which is sufficient for precise tuning of LC cell bias levels.

Figure 4 shows the simulated output for one channel, with the digital potentiometer set to the minimum value = 0. It is a ±68.61 mV square wave at 1 kHz, which is very close to the ±0 V.

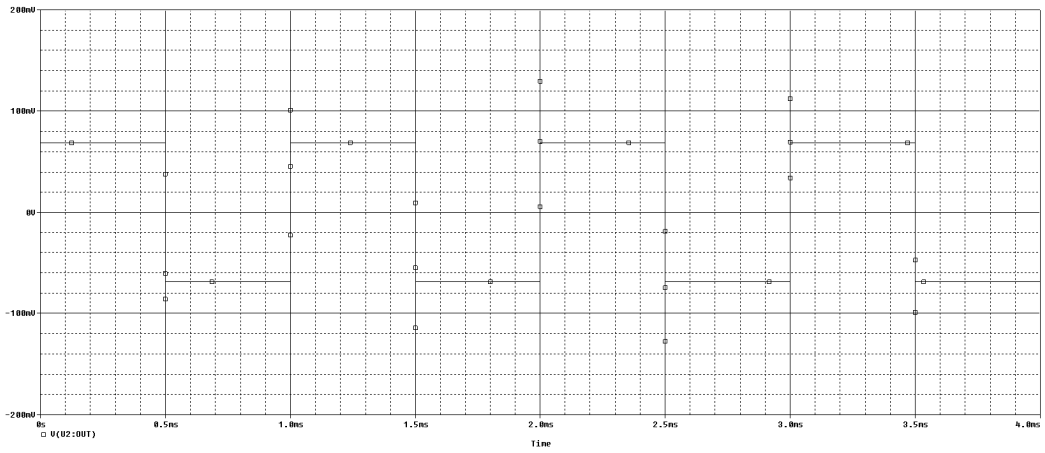
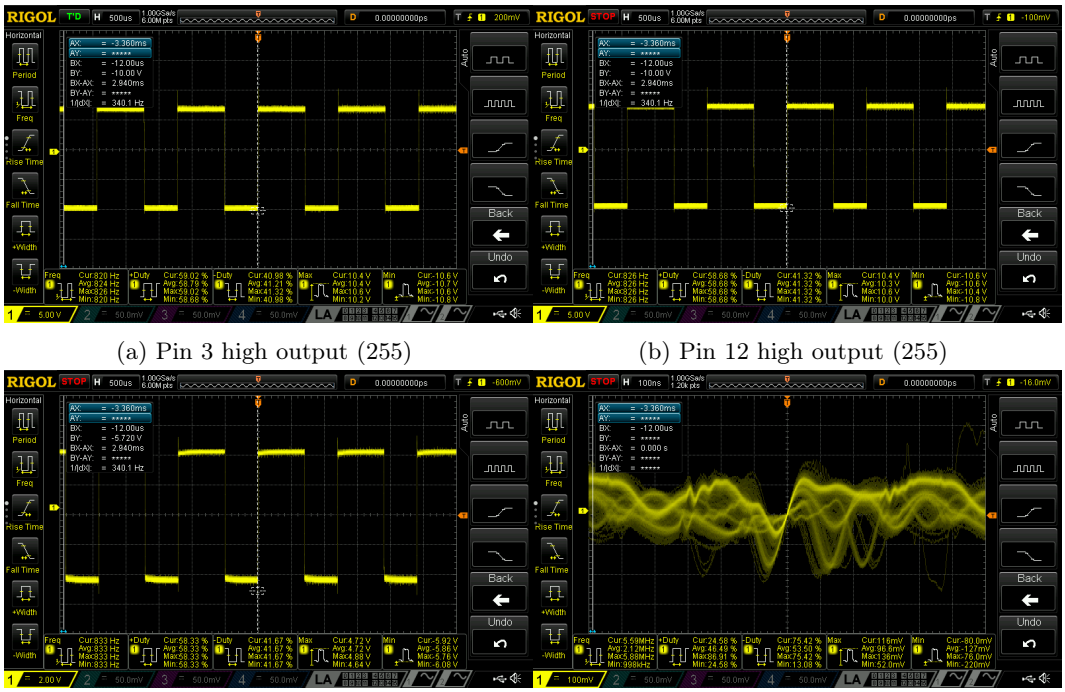



Figure 4: PSpice-Probe simulation results for one channel, with the digital potentiometer set to the minimum value = 0.



(a) Pin 3 high output (255)

(b) Pin 12 high output (255)

(c) Pin 11 medium output (128)

(d) Pin 3 low output (0)

Figure 5: Overview of the output channels with different control parameters. (a) Pin 3 high output (255); (b) Pin 12 high output (255); (c) Pin 11 medium output (128); (d) Pin 3 low output (0).

## 4.4 Reconfiguration latency and throughput

A key result is that the control system achieves very low latency updates, far below the 1 ms target. We measured the time from issuing a new amplitude command to the analog output settling by toggling channel values at the maximum possible rate. The Raspberry Pi Pico was clocked at 25 MHz for SPI and alternated a channel's digipot code between 0 and 255 as fast as possible. Figure 6 shows that the observed output toggle frequency was about 16.2 kHz, meaning the system could complete ∼16,200 full amplitude updates per second on that channel. This corresponds to a period of approximately 61.7 $\mu$s for a full high-low-high cycle, so the effective latency per update is half that period ∼30.9 $\mu$s. In other words, the new voltage level appeared on the output in well under 0.1 ms from the command, comfortably satisfying the real-time requirement. Even when updating all 64 channels in one burst, the operation is extremely fast - the Pico's SPI can stream all 64 channel updates in under 0.2 ms (at 25 MHz, sending 16 chips $\times$ 4 channels $\times$ 10 bits each $\approx$ 640 bits). Practically, this means the controller can refresh every channel's amplitude within a single 1 kHz waveform cycle, achieving simultaneous reconfiguration. The minimal skew between channels (on the order of microseconds or less) is governed by the SPI propagation delay, which is negligible for our purposes. This level of performance represents a significant improvement over prior LC-RIS controllers that often reconfigured elements sequentially with millisecond-scale delays per element. In summary, the platform demonstrated sub-50

$\mu$s latency for one voltage batch updates, enabling truly dynamic adjustments of the RIS with negligible control-induced delay. So the full 64 channels change would take $\sim 31 \times 4 = 124$ $\mu$s, so the driver itself is unlikely to limit system throughput. In practice, the USB link between the host PC and the Pico is slower than this; consider replacing it with a wireless link to avoid that bottleneck.

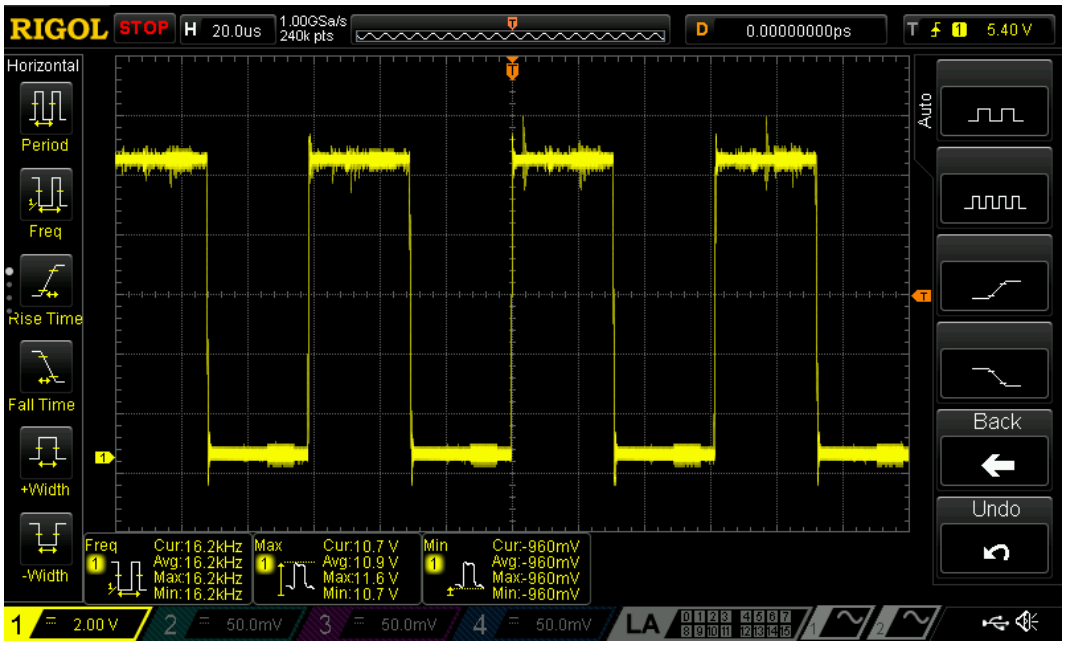



Figure 6: Frequency for one channel to change between high and low states at maximum clock speed.

## 4.5 Scalability

Although the prototype was built and tested with 64 channels, the architecture is inherently scalable to much larger arrays. All channels are driven in parallel by the same master oscillator, ensuring any number of added channels would still receive a synchronous AC wave. Scaling up primarily involves adding more AD5263 digipot chips (each providing four channels) daisy-chained on the SPI bus. The daisy-chain approach means a single serial command can update a large array of channels, at the cost of a longer shift register. In our current design, 16 chips (64 channels) are chained, and adding, say, another 16 chips would double the channel count to 128 with a proportional doubling of the SPI data length. The measured throughput indicates that even 128 channels could be updated in roughly 0.25 ms if using the same single bus (since update time increases linearly with channel count). In practice, for extremely large RIS panels with hundreds or thousands of elements, multiple SPI buses or controllers could operate in parallel to divide the load and maintain low latency per update. The control software is already designed to batch-update all channels in one cycle, so it would naturally accommodate a longer update frame for more channels. Additionally, the choice of inexpensive, modular components means scaling to hundreds of channels is economically feasible. Overall, the experiments confirm that the system not

only handles 64 channels with ease but also provides a clear path to high-channel-count scalability without fundamental performance degradation.

# 5 Responsible Research

**Ethical and safety considerations**: Developing a voltage control platform for LC-RIS required careful attention to safety and ethical use. On the hardware side, the design was kept within safe voltage and power limits to protect both users and devices. The output drive is $\pm 10$ V at small currents, which is below hazardous levels; nevertheless, the circuit includes proper insulation and power regulation to prevent accidental surges. The LC-RIS system's passive nature also means it does not emit any RF energy by itself - it only biases the RIS elements, which avoids interference with other devices. This contributes to safety in electromagnetic terms, as the platform will not inadvertently transmit or jam signals. In terms of potential misuse, one should consider that a reconfigurable surface could be employed to steer or focus signals in unintended ways. However, our platform is a tool for research and requires physical access to the hardware, so the risk of malicious use is low. Still, responsible deployment would involve secure access control on the software (to ensure only authorised commands adjust the RIS) and adherence to communication regulations so that any beamforming does not violate spectrum licenses or privacy. We also reflect on the broader ethical context: Reconfigurable Intelligent Surfaces are envisioned to improve wireless coverage and energy efficiency, which is a positive societal impact. Ensuring our control system functions reliably contributes to these benefits. We must also be mindful of environmental impacts - our hardware uses lead-free components and reasonable power levels, aligning with sustainability. By enabling passive, low-power RIS technology (which can reduce the need for power-hungry base stations), our work supports the goal of more energy-efficient wireless networks.

**Reproducibility and open science**: We endeavoured to make the research as reproducible as possible by using open-source tools and widely available hardware. The firmware for the controller is written in MicroPython running on a Raspberry Pi Pico, a platform chosen for its low cost and accessibility. MicroPython and the Pico's SDK are open-source, allowing other researchers to easily inspect, modify, and run the control software on their own hardware. All core components of the system, from the AD5263 digital potentiometers to the OPA4197 op amps, are off-the-shelf parts with publicly available data sheets. This means anyone can obtain the same parts and replicate the circuit. We provide detailed documentation of the circuit design and wiring so that the hardware setup can be reproduced without proprietary elements. In addition, our measurement and validation procedures are described step-by-step (capturing how we measured latency, waveform quality, etc.), which aids in independent verification of results. By sharing schematics, code, and test procedures openly, we adhere to the principles of responsible research: transparency, reproducibility, and accessibility. This approach lowers the barrier for other groups to build upon our work, for example, extending the controller to larger surfaces or integrating it into their own LC-RIS experiments, thereby accelerating progress in the field. We believe that an open hardware/software methodology not only benefits the academic community but also ensures that any potential issues (ethical or technical) can be quickly identified and addressed by a broader group of engineers.

# 6 Conclusions and Future Work

## 6.1 Conclusions

This project set out to answer the question: "What is a scalable, low-latency hardware-software solution for controlling LC-RIS systems?" Through the design, implementation, and testing of our 64-channel control platform, we have shown that such a solution is both feasible and effective. The proposed architecture combines a centralised analog waveform generator with distributed digital amplitude control to meet the requirements of LC-based metasurfaces. In hardware terms, we selected components that satisfied the sub-questions: an array of AD5263 digital potentiometers (each providing 4 channels, 256 levels) was used as the voltage driver for $>$50 channels, achieving the needed $\pm$10V swing with 8-bit resolution. These digipots, together with precision OPA4197 op-amps as buffering amplifiers, allowed us to drive each LC cell with an independent bias voltage while maintaining signal integrity. For waveform generation, we offloaded the task to a simple NE555 oscillator, creating a 1 kHz AC reference shared by all channels, thus guaranteeing synchronised bias across the entire surface without per-channel jitter. This analog waveform approach, coupled with fast digital control, proved to simplify the overall design while ensuring timing alignment. On the software side, a Raspberry Pi Pico microcontroller (running MicroPython) handled the control logic and SPI communication. By optimising the code for batch updates and leveraging the Pico's SPI at 25 MHz, we achieved a worst-case latency well below 1 ms from receiving command to output change - in fact, on the order of only tens of microseconds in our tests. This fulfils the latency sub-question and demonstrates that real-time reconfiguration of a multi-channel LC-RIS is practicable with a low-cost microcontroller. Scalability was addressed by using a daisy-chained SPI topology and modular channel expansion; our 64-channel prototype can be seen as a building block for larger systems, and we discussed how the design could scale to hundreds of channels without fundamental changes. Finally, a comprehensive validation was performed (power stability, waveform fidelity, amplitude linearity, update speed), and the results confirmed that the controller meets its design goals on most metrics. In summary, the research question is answered by our demonstrated platform: a scalable (64+ channels), low-latency ($<$ 1 ms) hardware/software solution that effectively controls LC-RIS elements. This solution contributes a novel architecture to the LC-RIS field, emphasising that analog-assisted waveform distribution with digital precision control is a viable path toward large, fast reconfigurable surfaces. It offers a reference design for future RIS controllers in terms of both circuit methodology and software strategy, bridging the gap between small-scale laboratory setups and the demands of large-scale, real-time tunable metasurfaces.

## 6.2 Future Works

Several improvements and extensions can be pursued to enhance the system's performance and adaptability:

**Enhanced Power Supply Design**: The most immediate improvement is to upgrade the +5 V supply to prevent droop under load. This can involve using a higher-current LDO or a switching DC-DC converter in place of the LT3032's 5 V output. By fortifying the power supply, the system will exhibit improved stability and can be scaled to more channels or higher drive strengths as needed.

**Oscillator Stability and Precision**: To achieve a more precise and reliable 1 kHz output, future iterations might replace or augment the NE555 timer. One option is to use a CMOS 555 variant (such as the TS555 or LMC555), which offers much lower power con-

sumption and smaller current spikes during transitions. A CMOS timer would significantly reduce the load on the 5 V rail and minimise interference, thus improving frequency stability. It can also operate at the needed frequency with better accuracy and less temperature variation. Another approach is to use a microcontroller or a crystal-controlled oscillator to generate the 1 kHz square wave. A small microcontroller could output a precise 1 kHz PWM (and even allow programmability of frequency or duty cycle if ever required). This digital approach would virtually eliminate the RC tolerance issues and provide exact 50% duty cycle by design. The trade-off is added complexity and noise from high-frequency switching, but given the low frequency (1 kHz) needed, a microcontroller's internal timer could directly toggle an output pin without introducing significant jitter. For our application, the simplicity of the analog 555 was initially attractive; however, ensuring frequency accuracy might justify a more modern oscillator in the next design revision. Any chosen oscillator should have a start-up stabilisation time considered if phase synchronisation across many channels is important.

**Thermal Drift Mitigation**: Although the system's output stabilises after warm-up, it would be beneficial to reduce sensitivity to temperature changes. Future work could include temperature compensation measures (using heat sinks and fans).

# A Schematics

This appendix contains the schematic set for each separate module and the system as a whole. There is a repository with all the code and design documents [11].

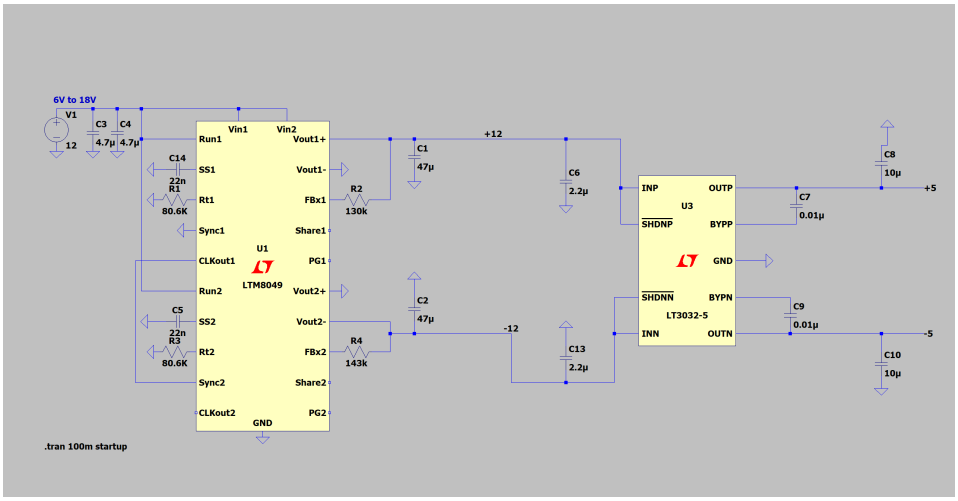



Figure 7: LTspice schematic of the LTM8049 and LT3032-5.

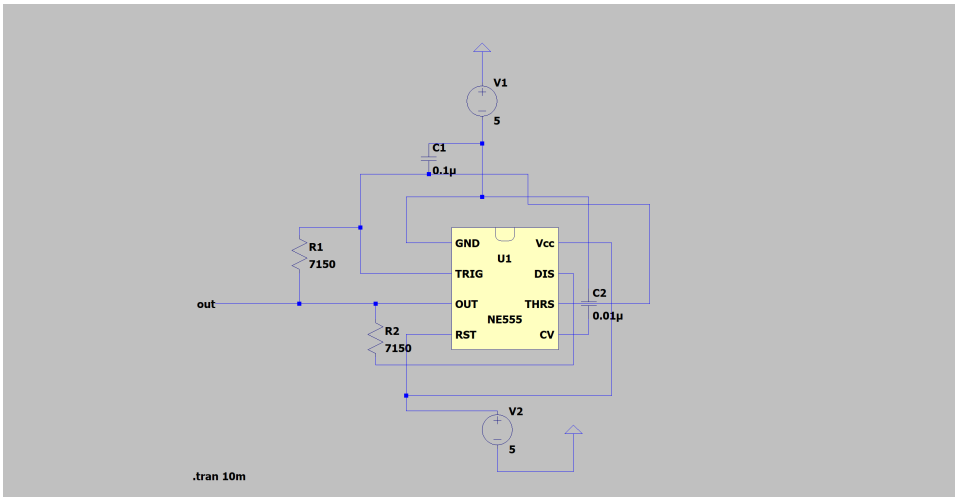



Figure 8: LTspice schematic of the NE555.

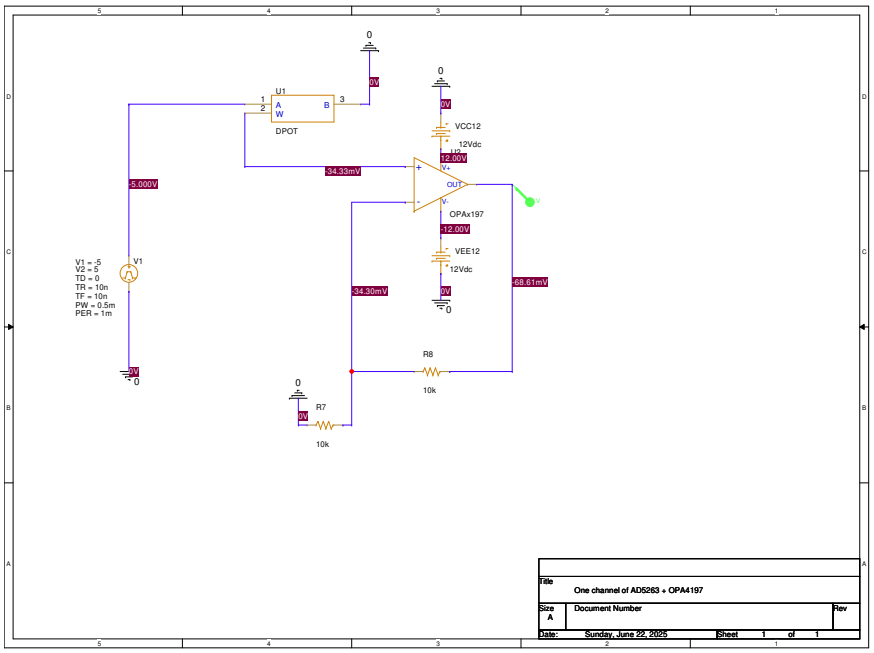



Figure 9: OrCAD Capture schematic of the AD5263 and OPA4197.

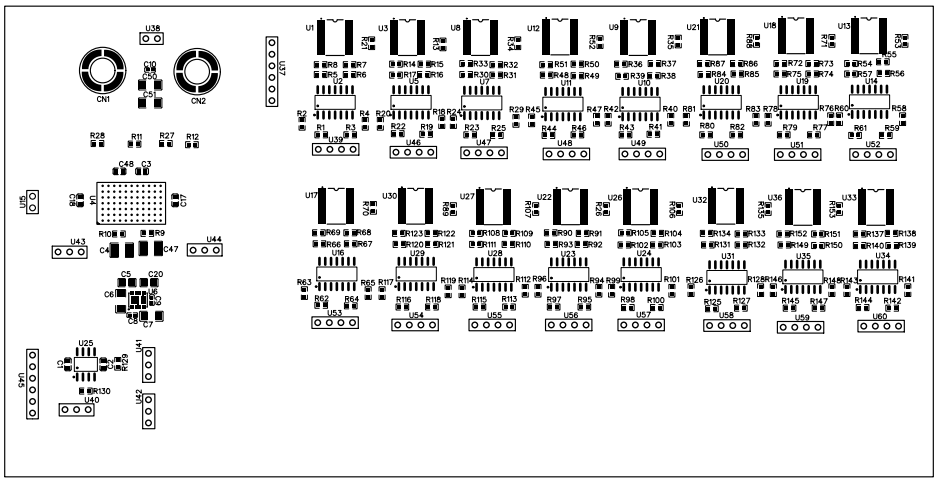


Figure 10: EasyEDA schematic of the PCB.

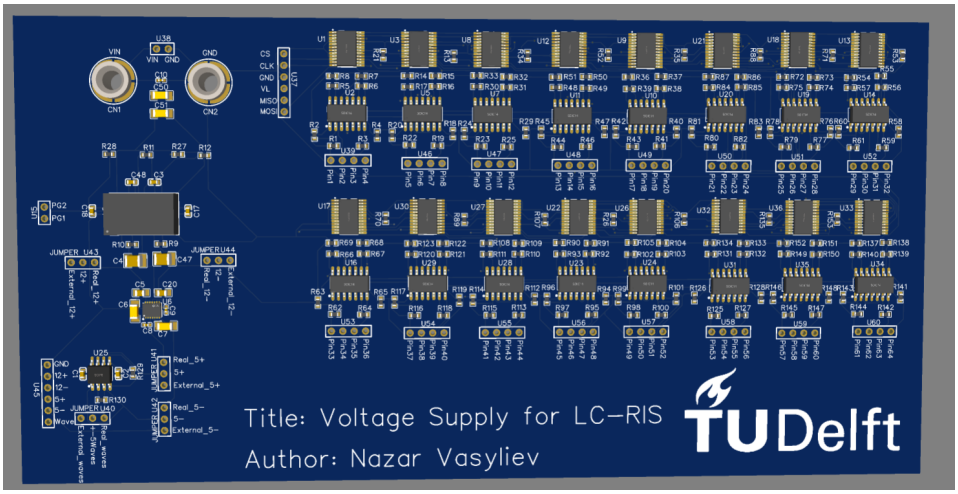



Figure 11: Rendered picture of soldered PCB.

# B Use of Large-Language-Model Assistance (ChatGPT)

1. Early stage: Brainstormed ideas and asked the model to clarify specialised concepts. I always requested at least one primary source for every claim, to read further.

2. Outline design: Requested feedback on the logical flow, ordering of sections and possible "missing links" between topics. Incorporated suggestions selectively while redrafting the outline manually; the final structure is my own.

3. Stylistic refinement: Asked the model to rephrase draft paragraphs for greater clarity, concision and a tone appropriate for a Bachelor's thesis in Computer Science. Used rephras-

ings as stylistic hints, took some good words or phrases; no large, uninterrupted blocks of AI text.

4. Proofreading: Queried ChatGPT for potential grammatical inconsistencies, ambiguous wording, and overly informal phrasing. Reviewed each recommendation; accepted only those that aligned with programme style guidelines.

5. LaTeX troubleshooting: Sought help with LaTeX issues: figure placement, table formatting, citation commands. Shared snippets of my own LaTeX code to diagnose errors and improve visual layout. Tested every code snippet locally; adopted solutions that compiled correctly and matched required formatting rules.

# References


[1] Analog Devices. *AD5263: Quad, 256-Position, Digital Potentiometer with SPI Interface*, 2002. Data Sheet.

[2] Analog Devices. *LTM8049: Dual SEPIC or Inverting μModule DC/DC Converter*, 2016. Data Sheet, Rev. B.

[3] Analog Devices. *LT3032: Dual 150 mA Positive/Negative Low-Noise Low-Dropout Linear Regulator*, 2017. Data Sheet.

[4] Mohamadreza Delbari, Robin Neuder, Alejandro Jiménez-Sáez, Arash Asadi, and Vahid Jamali. Fast reconfiguration of lc-riss: Modeling and algorithm design. *arXiv preprint arXiv:2504.08352*, 2025.

[5] Alejandro Jiménez-Sáez, Arash Asadi, Robin Neuder, Mohamadreza Delbari, and Vahid Jamali. Reconfigurable intelligent surfaces with liquid crystal technology: A hardware design and communication perspective. *arXiv preprint arXiv:2308.03065*, 2023.

[6] Robin Neuder, Marc Späth, Martin Schüßler, and Alejandro Jiménez-Sáez. Architecture for sub-100 ms liquid crystal reconfigurable intelligent surface based on defected delay lines. *Communications Engineering*, 3(70), 2024.

[7] Raspberry Pi Ltd. *Raspberry Pi Pico: RP2040-Based Microcontroller Board*, 2024. Data Sheet.

[8] Marco Rossanese, Placido Mursia, Andres Garcia-Saavedra, Vincenzo Sciancalepore, Arash Asadi, and Xavier Costa-Pérez. Open experimental measurements of sub-6ghz reconfigurable intelligent surfaces. *arXiv preprint arXiv:2404.01796*, 2024.

[9] Texas Instruments. *OPA4197: Quad, 36-V, Precision, Rail-to-Rail I/O, Low-Offset-Voltage Operational Amplifier*, 2018. Data Sheet, Rev. C.

[10] Texas Instruments. *NE555: Precision Timer*, 2025. Data Sheet, Rev. J.

[11] Nazar Vasyliev. https://github.com/bomberkomar/rp-lc-ris. GitHub repository, (22 June 2025).

[12] Jinghe Wang, Wankai Tang, Jing Cheng Liang, Lei Zhang, Jun Yan Dai, Xiao Li, Shi Jin, Qiang Cheng, and Tie Jun Cui. Reconfigurable intelligent surface: Power consumption modeling and practical measurement validation. *arXiv preprint arXiv:2211.00323*, 2022.

[13] Ziang Yang, Haobo Zhang, Hongliang Zhang, and Lingyang Song. Reconfigurable intelligent surface-enabled integrated sensing and communication: A sensing-assisted communication framework. *Frontiers in Communications and Networks*, 4(1176322), 2023.

[14] Xiaojun Yuan, Ying-Jun Angela Zhang, Yuanming Shi, Wenjing Yan, and Hang Liu. Reconfigurable-intelligent-surface empowered wireless communications: Challenges and opportunities. *arXiv preprint arXiv:2001.00364*, 2020.