[ARC]

Official Website: https://arc.istratiestefan.com/

Source Code: https://github.com/IstratieStefan/ARC

Hardware Repository: https://github.com/IstratieStefan/ARC-Hardware

---

Chapter I. Practical Utility

ARC is a portable device built for developers, researchers, and engineers who need tools for debugging, communication, and RF/NFC/LoRa analysis in the field. It combines a mini Linux computer (Orange Pi Zero 2 W) with a custom motherboard that includes a keyboard, 3.5” SPI display, I²S audio output, battery, and power circuits.

Problem Solved: The need for a compact, autonomous, and multifunctional terminal for field communication tests and operations (e.g. pentesting, LoRa networks, GSM AT commands, RF sniffing, NFC reading/emulation).

Efficiency vs. Alternatives: Compared to a laptop with external adapters, ARC is far more portable, has its own touchscreen and keyboard, and is designed for quick debugging or field sessions.

Chapter II. Mechanics

Section II.1. Complexity

- No active mechanical components (e.g. motors).
- Mechanical components include: compact case, 39-key tactile QWERTY keyboard, display mounting system, external ports (16-pin GPIO, power, UART, SPI, I²C, USB-C, 3.5mm jack).

Section II.2. Build Efficiency

- 3D-printed enclosure
- Custom PCB
- Battery life: up to 6–7h in mixed use

- No renewable energy source, but efficient power management (power switch, sleep button, etc.)

Chapter III. Electronics

PCB Architecture

- **Microprocessor:** Orange Pi Zero 2 W with quad-core Cortex-A53 @1.5 GHz and 4GB RAM
- **Microcontroller:** RP2040 (Waveshare Pico Mini) for keyboard, connected via I²C
- **Auxiliary circuits:**
  - Power: MCP73833 (charging), TPS61032PWP (5V boost)
  - Audio: PCM5102A (I²S DAC), PAM8302 (amplifier), 8Ω speaker
- **Modules:**
  - NFC: ESP32-S3 Mini + PN532 module
  - RF: ESP32-S3 Mini + sub-GHz CC1101
  - IR module

Connections

- I²S (DAC)
- I²C (Keyboard, touchscreen, modules)
- UART (Modules)
- SPI (Display, modules)

Active Components

#	Name	Description
1	Orange Pi Zero 2W	Single-board computer (SBC)
2	Waveshare 3.5”	480x320px SPI display
3	4x4mm tactile btn	Keyboard switch
4	MCP73833	Li-Ion charging IC
5	TPS61032PWP	5V boost converter
6	RP2040-tiny	Microcontroller for keyboard
7	PCM5102A	I²S DAC
8	PAM8302	Audio amplifier
9	8Ω Speaker	Audio output
10	Battery	Power source

Connectors

#	Name	Description
1	USB-C Port	Charging
2	USB-C Port	Data
3	3.5mm Jack	Headphone jack with detect pin for switching
4	IO Headers	Two sets of pins for power and module interface
5	Mini HDMI	Video output

Left Header

Pin	Net	SBC Signal	Function
1	GND	GROUND	Common ground for extensions
2	+5 V	5V out	Unregulated 5V power
3	GPIO12	GPIO12	General-purpose I/O
4	GPIO21	GPIO21	General-purpose I/O
5	GPIO22	GPIO22	General-purpose I/O
6	MOSI	SPI0 MOSI	SPI master-out
7	MISO	SPI0 MISO	SPI master-in
8	SCLK	SPI0 SCLK	SPI clock

Right Header

Pin	Net	SBC Signal	Function
1	GND	GROUND	Common ground for extensions
2	+3.3 V	3.3V out	Unregulated 3.3V power
3	+5 V	5V out	Unregulated 5V power
4	GPIO16	GPIO16	General-purpose I/O
5	RXD	UART0 RX	Serial data IN
6	TXD	UART0 TX	Serial data OUT
7	SDA	I²C SDA	I²C data line
8	SCL	I²C SCL	I²C clock line

Section III.1. Complexity

- Semi-autonomous device, but can execute tasks completely independently (e.g. LoRa scan, NFC emulation, GSM testing) through the graphical interface.

Chapter IV. Software

- Operating system: custom Linux distribution based on Armbian with Openbox + a lightweight Python-based desktop environment (ARC Desktop Environment).
- The interface is built using **Python + Pygame**.
- Each app is modular and runs in isolation.
- Every UI element, integrated app, script path, and more can be configured via the arc.yaml file located in ~/.config.
- The keyboard firmware is written in CircuitPython and sends matrix key codes to a Python script on the SBC, which emulates keypresses. The keyboard uses a layered layout system similar to mobile devices. The layout can be modified in the Python interpreter script.

Types of apps:

- Terminal
- WiFi tools
- Bluetooth tools
- IR tools
- RF tools
- NFC tools
- Music player
- Game launcher

+ any other Linux apps

ARC Connect

ARC can be connected to a web interface (FastAPI server) where you can:

- Monitor resource usage, uptime, IP address, and available storage
- Use a web-based SSH terminal (run commands directly from the browser)
- Transfer files to the /uploads directory

To connect, the laptop/phone must be on the same Wi-Fi network as the ARC device. Go to https://arc.istratiestefan.com/arc-connect and enter the IP, username, and password.

Chapter V. Industrial Design

- The case consists of two parts and can be assembled using only 4 screws, making it easy to repair or disassemble.
- The color scheme and design elements (like the speaker grill) are inspired by Dieter Rams' work.
- The circuit is compact, with a 4-layer PCB, integrated keyboard, and onboard power/audio circuits.
- Can be semi-industrially assembled using standard SMD components.
- Documentation includes all necessary files: schematic, PCB layout, 3D STEP, BOM, etc.

System Requirements

- **Minimum RAM:** 512 MB
- **CPU:** 1 GHz
- **OS:** Linux-based system

ARC Connect Requirements:

- A functional browser to use the ARC Connect interface

Useful Links:

- Project Website: https://arc.istratiestefan.com
- Source Code: https://github.com/IstratieStefan/ARC
- Hardware Repo: https://github.com/IstratieStefan/ARC-Hardware