Mini-Game Console

Introduction

- The project consists of a portable gaming console built around the T-HMI LilyGO ESP32-S3 module with a 2.8” LCD display.
- The goal of the project is to create a portable, programmable, and extensible gaming console that offers an accessible platform for developing and running simple games. The console allows both playing predefined games and the possibility to create and upload your own games, thus transforming it into an educational tool for programming and electronics.
- The idea originated from a passion for retro games and the desire to create a portable gaming platform that could simultaneously serve as a learning tool.
- It is useful as educational material for learning programming and electronics through game creation, as well as an alternative to commercial gaming consoles.

Descriere generală

This portable game console effectively integrates ESP32-S3 technology with physical controls and audio output to create a compact, programmable gaming device. Its simple yet functional design makes it accessible for both gaming enthusiasts and those learning electronics and programming.

Hardware Design

- **LilyGO T-HMI ESP32-S3 Module with 2.8″ LCD**

1. **Microcontroller:** ESP32-S3
2. **Flash:** 16 MB
3. **RAM:** 8 MB PSRAM
4. **Display:** 2.8″, 320 × 240 pixels, touchscreen
5. **Connectivity:** Wi-Fi and Bluetooth
6. **USB-C port** for programming

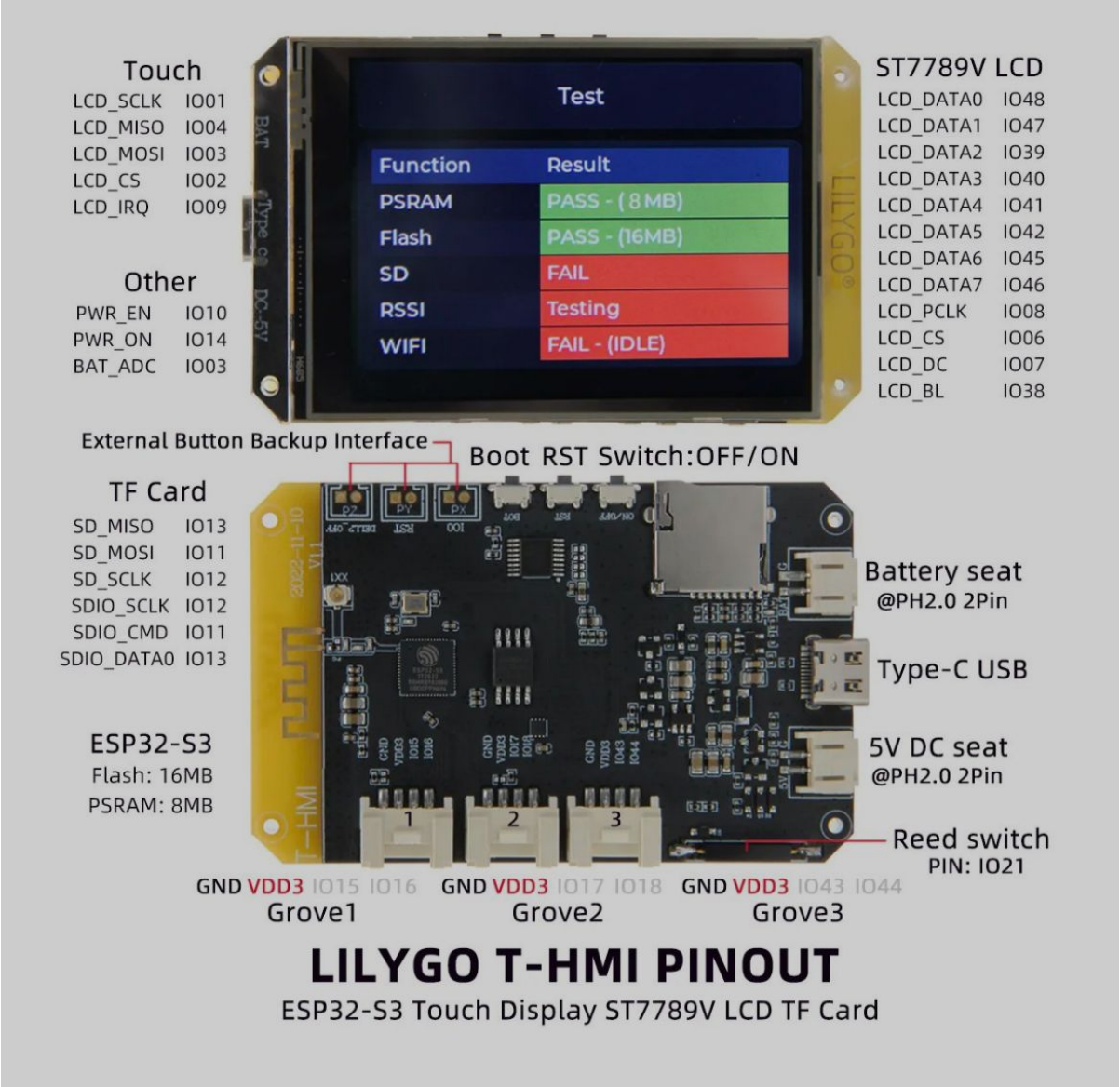
7. **SD-CARD** 32GB

- **Joystick Development Board**

1. 4 action buttons
2. 1 analog joystick

- **5 V Passive Buzzer Module**

- **SD-Card reader module + 32GB SD-Card**



The buzzer, buttons, and joystick will be connected to the I/O pins. (IO15, IO16, IO17, IO18, IO43, IO44).

The SD-CARD reader uses IO11, IO12, IO13 pins.

Chapters from laboratories used:

1. Joystick (ADC lab)

2. Buzzer driven by PWM signals tone (pin,freq,duration) handles waveform generation. (PWM lab)

3. Display using TFT_eSPI for ESP32-optimized rendering (SPI lab)

Bill of Materials (BOM)

#	Part / Module	Main Function	Qty.	Purchase Link	Datasheet / Additional Info
1	**Passive Buzzer Module – 5 V**	Generates audible tones/buzzer alerts when driven by a microcontroller pin.	1	eMAG product page	Typical 5 V passive buzzer datasheet (example): < https://cdn-shop.adafruit.com/product-files/160/160_datasheet.pdf>
2	**LilyGO T-HMI (ESP32-S3 + 2.8″ LCD)**	Core MCU board that provides Wi-Fi, Bluetooth, and a 2.8-inch touch LCD for HMI.	1	eMAG product page	Official LilyGO GitHub (schematic & datasheet bundle): < https://github.com/Xinyuan-LilyGO/T-HMI>
3	**I²C General-Purpose Joystick Board (4 buttons + 1 joystick)**	User input module offering a mini-joystick and push-buttons over I²C.	1	eMAG product page	Example joystick module datasheet: < https://cdn-learn.adafruit.com/assets/assets/000/008/495/original/joystick-datasheet.pdf >
4	**ADATA microSDHC 32 GB Class 10 + SD Adapter**	Provides removable storage for data logging or firmware assets.	1	[eMAG product page](https://www.emag.ro/card-de-memorie-microsdhc-adata-32gb-adaptor-sd-class-10-ausdh32guicl10-ra1/pd/D6Z2S6BBM/)	

Software Design

Currently Developing a mini-game Space Defender type.

The player is a pilot of a spaceship through an endless asteroid field.

The joystick is used to move on the oX, oY axis and the button of the joystick is used to start/pause the game. When starting the console the game will initially be paused. (When paused all the objects within the game are grey) After hitting an asteroid the buzzer will produce a 'Game Over' sound.

Software used in developing the game console:

1. Arduino Framework

- Provides hardware abstraction for ESP32

- Handles basic GPIO operations

- Manages timing and delays

- Enables serial communication

2. TFT_eSPI

- Controls TFT LCD display output

- Manages display settings

3. SD_MMC

- Interfaces with SD card storage

- Reads game asset files

- Handles filesystem operations

- Manages data transfers

4. BSR Framework

- Provides 3D rendering capabilities

- Processes vertex data

- Handles texture mapping

- Manages depth buffering

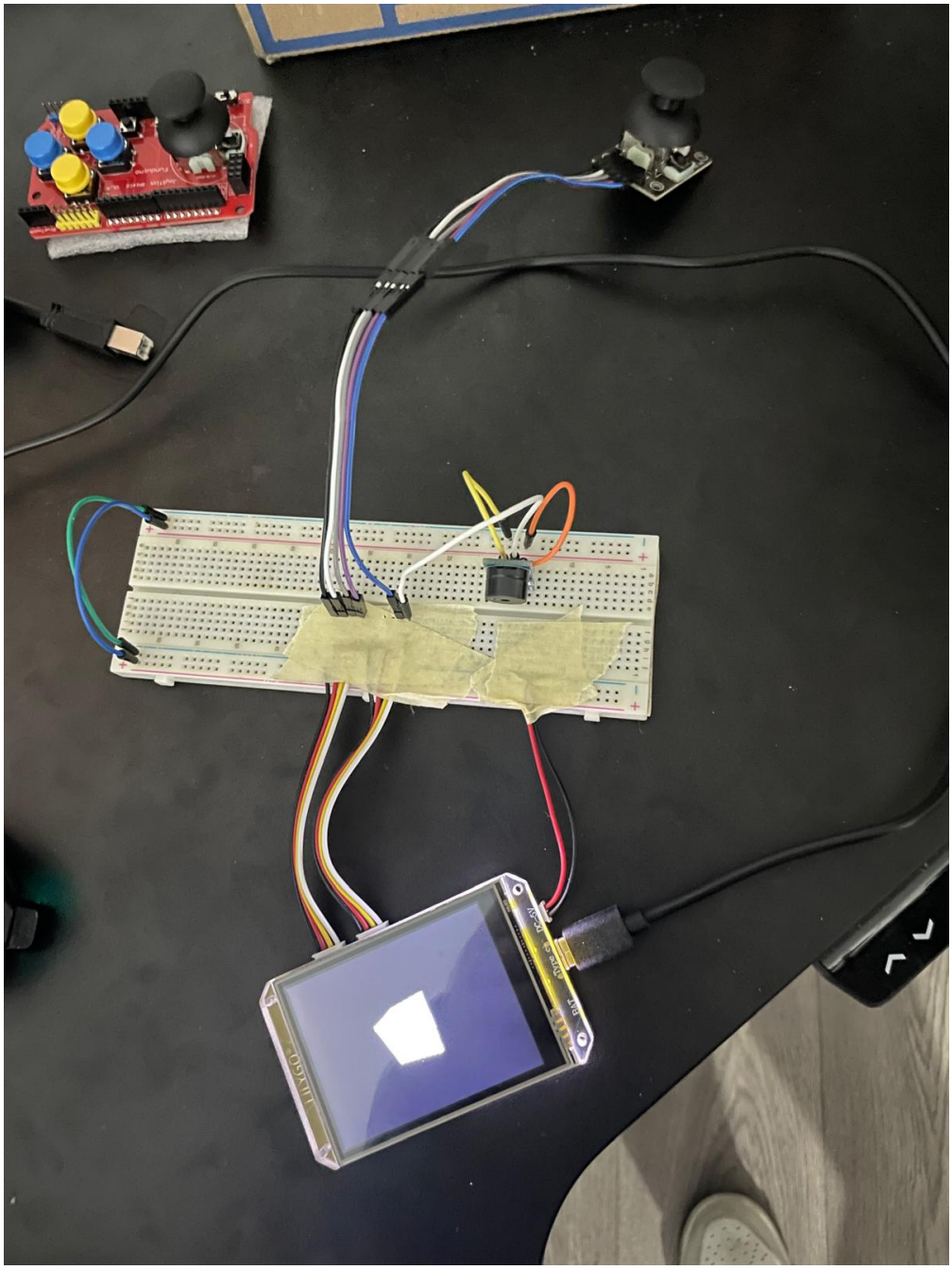
- BSR framework used to render graphics on the ESP32 microcontroller(open source project - GitHub link: https://github.com/Bogdy555/BSR)

DEMO version:

The first version of the game! - https://www.youtube.com/watch?v=BtcSVL-yEIg&ab_channel=CartofulAlbastru

Results

A hands-on console featuring a small playable game with graphics!



Concluzii

Download

consolegame.zip

Bibliography/Resources

https://github.com/Xinyuan-LilyGO/T-HMI

https://handsontec.com/dataspecs/module/Arduino%20Shield/Joystick%20Shield.pdf

https://projecthub.arduino.cc/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-cf4191

https://github.com/espressif/arduino-esp32/blob/master/libraries/SD_MMC/examples/SDMMC_Test/SDMMC_Test.ino

https://github.com/Bodmer/TFT_eSPI

Export to PDF