

# **Sungka using ESP32**

by

Ron Paolo Molejona

Gerianne Burgos

Angelica Custodio

Darah Via Moscoso

Christine Jean Pagunsan

CMSC 197 Physical Computing

Project Documentation

December 2025

## **1. Introduction**

Physical computing focuses on the interaction between hardware and software to create systems that respond to real-world inputs and produce meaningful outputs. One way to understand this concept is through the digitization of traditional games, where physical actions are translated into electronic signals and processed through programmed logic.

This project, Sungka, is a physical computing implementation of the traditional Filipino board game also called Sungka using ESP32 microcontroller. Instead of using shells and manual counting, the game logic, turn management, and score tracking are all done electronically. User inputs are provided through buttons and switches, while game information is displayed on an LCD. To enhance usability and aesthetics, the electronic components are enclosed in a custom-made wooden box that houses the LCD and conceals internal wiring. This project demonstrates how physical inputs, embedded systems, and programmed logic can work together to recreate a culturally significant game in a modern, interactive form.

## **2. Objectives**

### *2.1. General*

The general objective of this project is to design and develop an electronic version of the Sungka game using physical computing principles.

### *2.2. Specific*

Specifically, the project aims to:

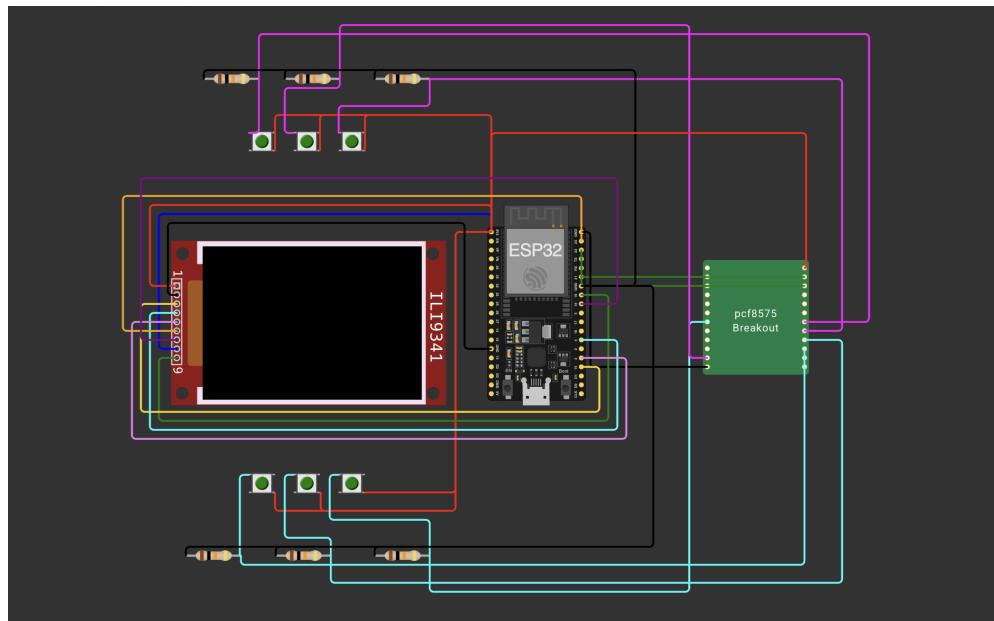
- Implement the rules and gameplay logic of Sungka using Arduino and ESP32.
- Display game status, turns, and results using an LCD module.
- Design a physical enclosure to protect components and improve presentation.

### 3. System Design

#### 3.1. Hardware Design

##### 3.1.2 Schematic Diagram

The schematic diagram displays the wiring connections, pin assignments, and component layout of the project.



<https://wokwi.com/projects/450575275859127297>

**Figure 1.** Schematic Diagram for Wiring Connections

##### 3.1.2 Materials

This section presents the materials used in the project and their respective functions.

| Material              | Description/Purpose  |
|-----------------------|--|
| ESP32 Microcontroller | Main microcontroller that runs the game logic and manages all inputs and outputs |

|                      |   |
|----------------------|---|
| ST7789 LCD           | Displays real-time game information including seed counts, player turns, and winner |
| Pcf8575 I/O Expander | Adds extra pins to ESP32 to handle all the button inputs                            |
| Buttons              | Used by players to control game actions and select houses                           |
| Resistors            | Stabilizes button signals   |
| Connecting Wires     | Connects all components together  |
| Plastic Box          | Encases the device for a cleaner presentation and simulates traditional game box    |
| Breadboard           | Used for connecting and organizing circuit components                               |

**Table 1.** Materials used for Sungka Game and its Purpose



**Image 1.** Sungka Device with the Plastic Box

### 3.2. Software Design

The rules of the traditional Sungka game were analyzed and translated into a logical flow that could be implemented programmatically. This included stone distribution, player turns, capturing mechanics, and end-game conditions. Button inputs represent player actions such as selecting a house or confirming a move. The system updates the number of stones in each house, determines whether a player retains or passes a turn, and checks for valid captures based on Sungka rules. The game continues until an end condition is met, after which the winner is determined and displayed on the LCD.

The LCD is continuously updated to reflect the current game status, allowing players to follow the game progression clearly. The program structure follows a loop-based execution model, where inputs are read, logic is processed, and outputs are displayed in real time.

|                   |  |
|-------------------|--|
| Board Setup       | Game start/init() <ul style="list-style-type: none"><li>- <i>Player 1</i>: Houses [1-7], Store at index [15] (left)</li><li>- <i>Player 2</i>: Houses [8-14], Store at index [0] (right)</li><li>- Each house starts with 7 seeds. Stores start at 0 seeds</li></ul> |
| Round transition  | After both moves are WAITING in round 1, switch to round 2 and P1's turn.<br><u>checkRoundOneEnd</u>   |
| Seed distribution | Pick up all seeds from the selected house and distribute them counterclockwise one by one to   |

|                        |  |
|------------------------|--|
|                        | subsequent houses and your own store, skipping the opponent's store.   |
| Extra turn (own store) | Last seed in own store; only extra turn in round 2+<br><u>animateP1, animateP2</u>   |
| Capture                | Last seed in your own empty house, then capture that seed plus all seeds in the directly opposite pit.<br><br>All captured seeds go to your score. |
| End turn on opponent   | Last seed in empty opponent house  |
| No valid moves         | Can't move on turn, opponent goes again or game ends   |
| Game end               | After both sides are empty or no moves left, count score, and declare winner   |