

Rapid Roll trên STM32F429 (TouchGFX + FreeRTOS)

1) Tên project

Rapid Roll – Game lăn bóng tránh chướng ngại vật

2) Mô tả nội dung

Project xây dựng trò chơi **Rapid Roll** chạy trực tiếp trên vi điều khiển **STM32F429**, hiển thị trên LCD 240x320 và điều khiển bằng **TouchGFX** (sử dụng 2 nút bấm trái/phải để di chuyển quả bóng).

Luật chơi tổng quát:

- Người chơi điều khiển bóng di chuyển trái/phải để rơi xuống các platform (các platform là các thanh ngang không có gai).
- Tránh spike (gai) và tránh chạm chướng ngại.
- Có các power-up: **Life**, **Score Up**, **SpeedUp**, **SlowUp**, **Shield/BlackBall** (giải thích ở trong tài liệu mô tả game: HDSD – RapidRoll.pdf).

3) Thiết kế phần cứng

Thiết lập các chân GPIO tương ứng với việc di chuyển quả bóng sang trái hoặc phải

- tín hiệu sang trái (nút LEFT) được nối với PC14
- tín hiệu sang phải (nút RIGHT) được nối với PC8

Mỗi lần đọc tín hiệu, hệ thống kiểm tra xem có tín hiệu LEFT và RIGHT được đọc không nhờ bitmask bằng việc XOR

- Nếu nhận tín hiệu LEFT thì XOR với 0x04
- Nếu nhận tín hiệu RIGHT thì XOR với 0x08

4) Thiết kế phần mềm

4.1 Kiến trúc tổng thể

- **HAL/Drivers (STM32Cube)**: cấu hình và điều khiển ngoại vi.
- **FreeRTOS**: chạy các task hệ thống, trong đó có task GUI.
- **TouchGFX**: quản lý UI, render, và tick event (frame update).
- **Game Logic**: chạy chủ yếu trong tick event của màn hình chơi (Main Screen).

4.2 Cơ chế gameplay cốt lõi

- **Gravity & rơi**: bóng rơi theo tốc độ cố định mỗi tick.
- **Cuộn màn hình**: platform/spike di chuyển lên theo gameSpeed (được chọn ở Level Screen).
- **Điều khiển ngang**: bóng di chuyển trái/phải trong phạm vi hai tường, input lấy từ nút cứng.
- **Platform/Spike**: platform là “sàn an toàn”, spike là “gai” (chạm sẽ mất mạng/trừ khi có shield).
- **Spawn/Recycle**: khi platform/spike đi lên quá màn hình sẽ được xuất hiện lại xuống dưới với vị trí ngẫu nhiên.

4.3 Quản lý mạng (Lives), hồi sinh, Game Over

- Lives/Score/Speed được quản lý trong **Model** và được Presenter/View đọc để hiển thị.
- **Mất mạng** khi bóng rơi khỏi đáy (và các điều kiện chết khác theo gameplay):
 - trừ 1 mạng
 - UI cập nhật động (“x lives”)
 - **delay hồi sinh ~60 ticks** (1 giây) trước khi bóng xuất hiện lại
- **Game Over** khi lives = 0:
 - hiện hình “Game Over”
 - dừng chuyển động và input gameplay
- **Reset game state** khi vào Main Screen từ menu (đảm bảo mỗi lượt chơi bắt đầu từ trạng thái mặc định)

4.4 Các file trọng tâm (để đọc/duy trì)

- **Input & hardware init:**
 - Core/Inc/main.h + Core/Src/main.c : Khai báo và triển khai các phần khởi tạo phần cứng
- **State (lives/score/speed):**
 - TouchGFX/gui/include/gui/model/Model.hpp + TouchGFXgui/src/model/Model.cpp :

- Lưu và quản lý trạng thái game toàn cục: số mạng (lives), điểm (score), tốc độ/độ khó (gameSpeed), best/last score.
- Mỗi tick, Model::tick() poll input (đọc keymask từ main.c) và chuyển input sang Presenter/View thông qua ModelListener.
- **Game loop & gameplay:**
 - TouchGFX/gui/include/gui/mainscreen_screen/MainScreenView.hpp + .../src/mainscreen_screen/MainScreenView.cpp
 - Là nơi chứa toàn bộ game loop chạy theo từng frame (handleTickEvent()): va chạm platform/spike, spawn/recycle platform, tính điểm, respawn, game over, power-up...
- **Hiển thị score động:**
 - TouchGFX/gui/src/scorescreen_screen/ScoreScreenView.cpp :
 - Hiển thị Best Score và Last Score lên màn hình Score.

5) Danh sách thành viên & đóng góp

Thành viên 1 – UI/TouchGFX

- **Họ tên:** Trần Sơn Tùng | **MSSV:** 20220048
- **Đóng góp chính:**
 - Thiết kế giao diện trên **TouchGFX Designer**
 - Tạo các màn hình: Home/Level/Score
 - Thiết lập widgets: texts, images, wildcards

Thành viên 2 – Hardware config & Game chạy được

- **Họ tên:** Lò Hải Long | **MSSV:** 20224873
- **Đóng góp chính:**
 - Cấu hình phần cứng.
 - Cài đặt input (GPIO) và tạo game loop chạy được
 - Thực hiện logic sinh platform/spike ngẫu nhiên, logic ball va chạm với platform/spike

Thành viên 3 – Chức năng game & Power-up

- **Họ tên:** Nguyễn Hoàng Việt | **MSSV:** 20220050
- **Đóng góp chính:**
 - Implement các power-up:

- **Life** (ăn vào tăng mạng)
- **Score Up** (nhân đôi điểm)
- **SpeedUp, SlowUp** (tăng/giảm tốc độ bóng)
- **Shield, BlackBall** (giữ quả bóng chịu được 1 mạng trong 10s)

CẤU TRÚC GIAO DIỆN GAME RAPID ROLL

I. Cấu trúc giao diện

1. HomeScreen



Giao diện màn hình trang chủ của game. Cấu trúc:

- Tiêu đề RAPID ROLL
- StartButton: Khi nhấn sẽ chuyển đến màn hình game (MainScreen)
- LevelButton: Khi nhấn sẽ chuyển đến màn hình level (LevelScreen)
- ScoreButton: Khi nhấn sẽ chuyển đến màn hình score (ScoreScreen)

2. Level Screen



Giao diện các level của game. Cấu trúc:

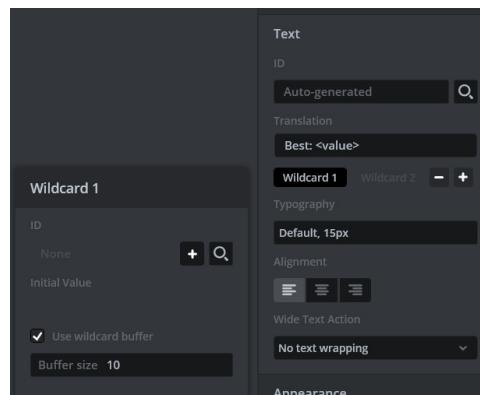
- Tiêu đề RAPiD ROLL
- EasyButton: Chuyển game về chế độ dễ. Khởi đầu với tốc độ cơ bản (người lập trình tự định nghĩa khi lập trình). Màn hình chuyển về HomeScreen.
- MediumButton: Chuyển game về chế độ trung bình. Khởi đầu với tốc độ trung bình (người lập trình tự định nghĩa khi lập trình). Màn hình chuyển về HomeScreen.
- HardButton: Chuyển game về chế độ khó. Khởi đầu với tốc độ cao (người lập trình tự định nghĩa khi lập trình). Màn hình chuyển về HomeScreen.
- BackButton: Chuyển màn hình game về HomeScreen

3. ScoreScreen



Màn hình ghi nhận điểm khi kết thúc trò chơi. Cấu trúc:

- Tiêu đề RAPiD ROLL
- Tiêu đề High Score
- Text Area:
 - + Best: Ghi nhận điểm tốt nhất trong tất cả các lần chơi.
 - + Last: Ghi nhận điểm ở lần chơi cuối.



- BackButton: Quay về màn hình HomeScreen

4. MainScreen



Màn hình chính của game. Cấu trúc:

- Thanh tiêu đề, chứa:

+ Số mạng hiện có: Có thể set tối đa là 10, tối thiểu là 1

+ Score: Điểm hiện tại

- - Ball: Quả bóng trong trò chơi. Quả bóng có thể di chuyển trái/phải. Trong khi rơi vẫn có thể nhấn trái/phải để di chuyển

- - Platform: Thanh đỡ quả bóng, xuất hiện ngẫu nhiên trong quá trình chơi

- - Spike: Thanh gai, xuất hiện ngẫu nhiên trong quá trình chơi

- - HeadSpike: Thanh gai cố định trên đầu

- Power-up: Gồm 5 power-up khác nhau xuất hiện ngẫu nhiên trong quá trình chơi:

+ - Life: Ăn vào sẽ tăng mạng của người chơi

+ - ScoreUp: Ăn vào sẽ tăng gấp đôi số điểm nhận được trong 10s

+ - SpeedUp: Ăn vào sẽ tăng gấp đôi tốc độ của quả bóng trong 10s. Nếu ăn hai lần liên tiếp power-up này thì tốc độ vẫn giữ nguyên (tức chỉ tăng đối đa 2 lần)

+  - Slow: Ăn vào sẽ giảm gấp đôi tốc độ của quả bóng trong 10s. Nếu ăn hai lần liên tiếp power-up này thì tốc độ vẫn giữ nguyên (tức chỉ giảm đôi đa 2 lần)

+  - Shield: Ăn vào sẽ giúp giữ nguyên số mạng hiện có khi chết 1 lần. Quả bóng sẽ chuyển sang màu đen (BlackBall)

+  - BlackBall: Trạng thái miễn nhiễm sau khi ăn Shield (#dadenchidontot)

- Tiêu đề GAME OVER xuất hiện khi thua

- BackButton: Màn hình quay về HomeScreen