

2D ゲームと PAD の制作

植田 峻生 榎本 堪太
門原 柊平 高橋 哲平

1. 研究概要

ピクセルアートのマルチエンディング 2DRPG ゲームと本ゲームを操作するためのコントローラー (PAD) の作成を目的とする。本研究では、ハードウェア設計とゲームデザインの開発に取り組むことで、ユーザビリティの知見を深めることを目的としている。

2. 研究の具体的内容

(1) ゲーム概要

私たちが開発したゲームは、登場人物やストーリー構成を含め、完全オリジナルで制作した作品である。さらに、マルチエンディング形式を採用しており、プレイヤーの選択によって異なる結末を迎えるため、何度でも楽しめる仕様となっている。

(2) 制作スケジュール

2DRPG ゲームと PAD 作成に当たって表 1 のような計画を立てた。初めてのことが多く、予定通りに進まないことがあった。

表 1 制作スケジュール

4 月	PAD、ゲームの基本設計
5 月	環境構築
6 月	コントローラー作成開始
7 月	ゲームストーリー構成
8 月	ゲーム作成開始、マップ作成
9 月	マップ作成
10 月	その他システム実装
11 月	バグ修正
12 月	報告書等資料作成
1 月	課題研究発表準備

(3) 開発環境

ゲームと PAD 制作で使用した開発環境は次のとおりである。

- Unity 2022.3.8f
- Visual Studio
- GitHub
- Blender
- Creality Slicer
- 123D Circuits
- Arduino

(4) 制作にあたる役割分担

ゲーム作成には、ストーリー構想やプログラミング、PAD 作成には、3D モデルの作成や配線作業が必要となる。そこで、表 2 のように分担し、作業を進めた。なお、ゲームでの家などのオブジェクトやキャラクターのイラストは、著作権フリーのものを使用している。

表 2 役割分担

植田	マップ作成、発表資料の作成
榎本	日報、ストーリー構想
門原	Unity プログラム、PAD 外装作成
高橋	ストーリー構想

(5) PAD の詳細と機能

(a) 使用部品

- Arduino R4 MICRO
- ジョイスティック
- タクトスイッチ
- PAD カバー
- ブレッドボード
- 抵抗

(b) PAD の設計

私たちが本研究で作成した PAD は、ジョイスティック 1 個とタクトスイッチ 4 個から構成されている (図 1・2)。一般的なコントローラーを参考にし、操作方法が誰にでも理解しやすいように設計した。

ジョイスティックは、キャラクターの移動操作 (上下左右) に対応しており、プレイヤーは直感的にキャラクターを操作することができる。

タクトスイッチは、選択肢の決定、キャンセル、攻撃などの操作を割り当てている。タクトスイッチを 4 個にすることで、ゲームに慣れている人でもそうでない人でも操作方法を覚えやすく、扱いやすい PAD となるように工夫をした。



図 1 ジョイスティック



図 2 タクトスイッチ

(c) PAD の外装

PAD の外装には、Blender を使用し 3D モデルを作成した。完成した 3D モデルを Crealty slicer を使用してスライス処理を実行し、3D プリンタで印刷した (図 3)。私たちが重要視

している、「ユーザビリティ」の観点からプロトタイプの評価をし、角を丸め、手にフィットするように改良を行った (図 4)。今後もさらに改良を重ねる予定だ。



図 3 改良前



図 4 改良後

(d) プログラム

ジョイスティックとタクトスイッチの入力をキーボードの入力に変換するコードを、それぞれ Arduino を使用して作成した (図 7・8)。この結果から、タクトスイッチが対応したキーを出力しており、正常に動作していることがわかった。

なお、使用したマイコンは Arduino R4 MINIMA である (図 5)。



図 5 Arduino R4 MINIMA

```

#include <Keyboard.h>

void setup() {
  Keyboard.begin();
  delay(1000); // 起動時の誤入力防止

  pinMode(2, INPUT_PULLUP); // 決定
  pinMode(3, INPUT_PULLUP); // キャンセル
  pinMode(4, INPUT_PULLUP); // 攻撃
}

void loop() {

  // 決定ボタン (Space)
  if (digitalRead(2) == LOW) {
    Keyboard.press(' ');
    delay(300);
    Keyboard.releaseAll();
  }

  // キャンセルボタン (Esc)
  if (digitalRead(3) == LOW) {
    Keyboard.press(KEY_ESC);
    delay(300);
    Keyboard.releaseAll();
  }

  // 攻撃ボタン (J)
  if (digitalRead(4) == LOW) {
    Keyboard.press('j');
    delay(300);
    Keyboard.releaseAll();
  }

  delay(10);
}

```

図6 タクトスイッチ制御の一部

```

#include <Keyboard.h>

const int joyX = A0;
const int joyY = A1;

// 中立付近のしきい値
const int centerLow = 400;
const int centerHigh = 600;

void setup() {
  Keyboard.begin();
  delay(1000);
}

void loop() {
  int xValue = analogRead(joyX);
  int yValue = analogRead(joyY);

  // --- 左右 ---
  if (xValue < centerLow) {
    Keyboard.press('a'); // 左
  } else if (xValue > centerHigh) {
    Keyboard.press('d'); // 右
  } else {
    Keyboard.release('a');
    Keyboard.release('d');
  }

  delay(10);
}

```

図7 ジョイスティック制御の一部

(e) 回路設計

123D Circuits を使用して配線を考えた(図8)。123D Circuits では、今回使用しているジョイスティックと Arduino R4 MICRO がなかったため、似ている部品で代用している。

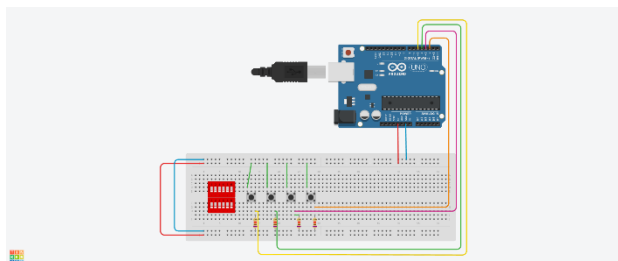


図8 123D Circuits で作成した回路

(6) ゲームの詳細と機能

ゲームを構成するシステムとして、プレイヤーが操作する主人公の名前の入力や登場キャラクターとの会話、敵モンスター・味方キャラクターの能力値、コマンド操作による戦闘、インベントリ機能などがある。

(a) ストーリー概要

ゲームを作成するにあたって、オリジナルのストーリーを構想した。下図は、そのストーリーのあらすじである(図9)。

主人公は 16 歳の王子で、次期国王として剣術や魔法、政治を学びながら日々を過ごしている。許嫁である公爵令嬢ノエル・ヴァーミリオンは、相手の資質を見抜く能力を持ち、幼い頃から主人公を支えてきた存在である。

ある地方視察で、主人公は魔族に滅ぼされた村を訪れ、不思議な既視感を覚える。その後、王都に魔物が襲来し、王国は大きな危機に陥る。ノエルから「英雄の資質」を持つことを告げられた主人公は、悩みながらも勇者として戦う決意を固める。城外での戦いの中、主人公は四大魔官リュウ

・タロウ・ラムノノと対峙し、戦闘の途中で自分が 11 年前に滅ぼされた村の生き残りであること、そして両親を魔族に殺されていることを思い出す。

図9 ストーリーのあらすじ

(b) 名前の入力

名前入力では、プレイヤーの好きな名前を入力し、その後のプレイに反映させることができる(図10)。

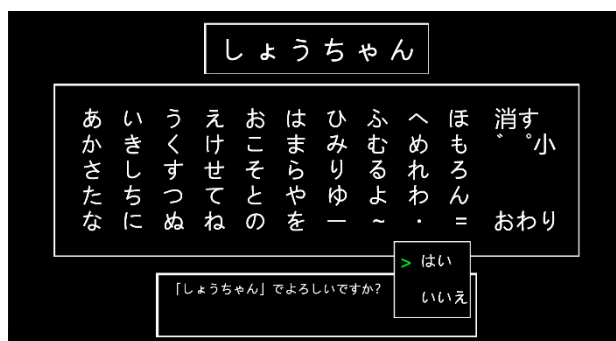


図10 名前入力画面

(c) 登場キャラクターとの会話

城内や森など各場所にいる主要なキャラクター(NPC)にキーボードの Enter、もしくはPADのタクトスイッチを押下することで話することができる。

(d) 能力値について

能力値は体力、攻撃力、防御力、魔力、マジックポイント、素早さの6つの項目がある。攻撃力と防御力によって物理攻撃のダメージ計算を行う。魔法攻撃のダメージ計算は攻撃を受ける側の魔力が防御力として計算する。敵モンスターは種類ごとにレベル1時点での能力値が決まっている。敵モンスターのレベルによって演算を行い、能力値を決定する。味方キャラクターの能力値はレベルアップとステータスポイントの使用で上昇する。ステータスポイントはレベルアップごとに得ることができ、各項目に割り振ることができる。

(e) コマンド操作による戦闘

戦闘時には攻撃、アイテムの使用、逃げるなどの行動を、ジョイスティックを用いた操作で決定する。行動順は能力値の素早さの値を参照する。

(f) インベントリ機能

インベントリ機能では所持しているアイテムの使用ができる。アイテムには消費アイテム、装備、宝物がある。消費アイテムには回復アイテムなどが含まれる。装備には剣や杖などの武器と防具が含まれる。宝物には使用しても消費しないアイテムや使用はできないがルート選択に関係のあるアイテムなどが含まれる。

(g) マップ

マップの作成には、Tilemap と呼ばれる Unity のコンポーネントを使用している。これは、分割されている建物や木などのオブジェクト、地面のパーツを Unity 側に出力されるタイルごとに組み合わせてマップを作成する。これを使用するメリットは、外部サイトで、著作権フリーで配布されている素材を取り込むだけで使用することができるため、自分たちでパーツを用意する必要がなくなり、取り込んだパーツを組み合わせるだけでマップを作成できる点とそのパーツを他の建物に流用することができる点である（図 11）。一方で、使用したいパーツがなく、新たに素材を取り込む必要があるというデメリットもあった。作成したマップは次のようになった（図 12・13）。



図 11 使用したパーツの一部



図 12 村の一部

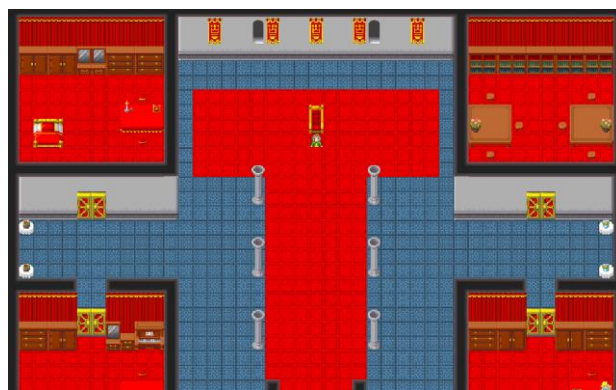


図 13 城内の一部

3. 研究のまとめ

本研究を通じて、2DRPG ゲームと専用 PAD の制作を実際に行い、ゲーム開発やハードウェア制作の難しさと面白さを体験することができた。制作を始めた当初は、ゲーム開発やコントローラー制作のどちらも初めてのことが多く、一つの作業に想像以上の時間がかかってしまった。そのため、思うように作業が進まず、不安やあせりを感じる場面も多かった。

また、班で役割分担をして制作を進めてい

たが、班員への指示が十分にできておらず、作業の進み具合を把握できていないこともあった。その結果、当初に立てていた計画から遅れてしまい、計画性の大切さを強く実感した。これらの経験を通して、自分たちの技術面や計画面での未熟さを痛感するとともに、今後の制作では事前の準備や話し合いをより丁寧に行う必要があると感じた。

今回の研究で得た反省点や経験を、今後の授業や制作活動に活かしていきたい。

4. 参考文献

・使用した素材サイト

ドット絵世界

<https://yms.main.jp/>

びぼや倉庫

https://pipoya.net/sozai/assets/map-chip_tileset32/

DOT ILLUSTRATION

<https://dot-illustration.net/>

CRAFTPIX.NET

<https://craftpix.net/>

Material Forward

<https://materialforwardvfx.wixsite.com/materialforward>

効果音ラボ

<https://soundeffect-lab.info/>