

# エアー・ホッケー とサンタさんの 6 人企画書

徳島大学 工学部 知能情報工学科 27 班

木村 薫

好川 汰一

近藤 彰人

長瀧谷 大輝

2016 年 11 月 10 日

## 1 ゲーム内容

### 1.1 概略

ネットワーク対戦型のホッケーゲームである。プレイヤーは 2 人ずつに分かれ「アタッカー」と「サポーター」のどちらかを操作する。コンピュータが操作する「ディフェンダー」と協力し、先に一定の点数を取ったほうが勝ちである。得点は、相手のゴールにパッドを入れるか、相手チームのプレイヤーを 2 人倒すことで入る。プレイヤーには、それぞれ異なる能力が設定されており、仲間との協力プレイと戦略が勝利の鍵を握る。

ゲーム画面は、OpenGL を使った 3D である。また、SDL を用いて、ステータス表示や、効果音・BGM など実装する。

ジャンル        スポーツゲーム

プレイ人数    4 人

使用デバイス   ゲームパッド

また、目標は「総合優勝」である。

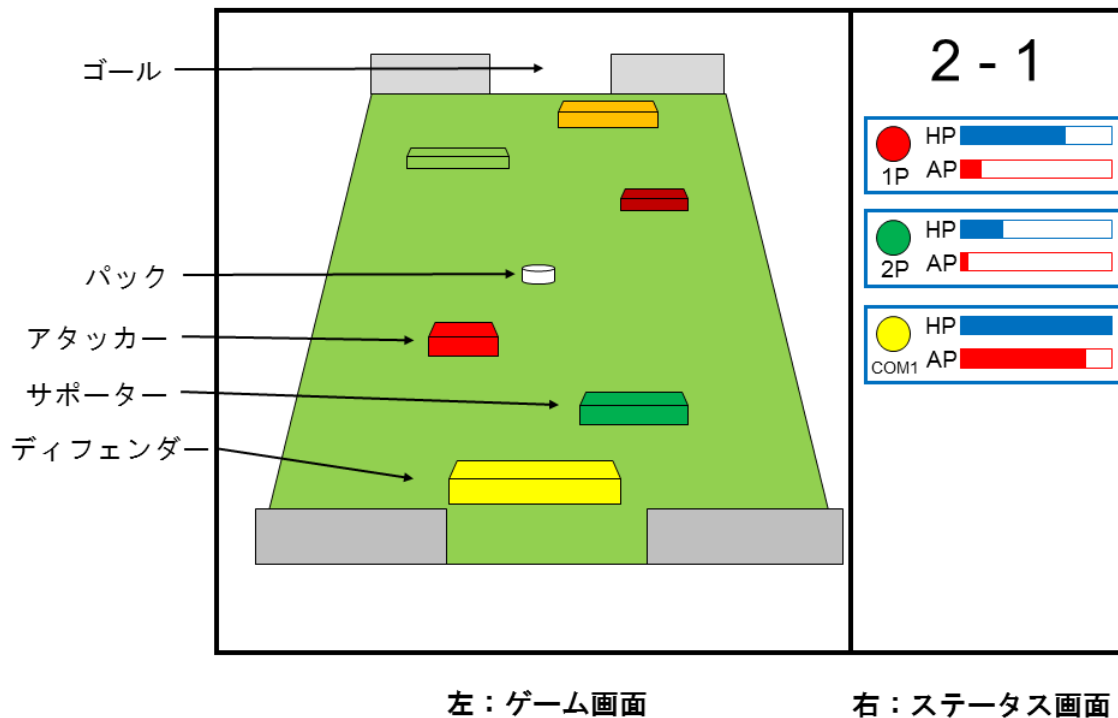


図 1: イメージ図

## 1.2 コンセプト

コンセプトは「新感覚アクションゲーム」である。単なるホッケーゲームではなく、基本能力と必殺技を駆使して戦うという点が「新感覚」である。また、「分かりやすい操作」と「やっているうちに理解できるルール」を目標としている。

## 1.3 各プレイヤーの説明

### 1.3.1 アタッカー（人間）

アタッカーは、攻撃を得意とするプレイヤーである。  
アタッカーが打ったパッドは速くなる。

### 1.3.2 ディフェンダー（コンピュータ）

ディフェンダーは、コンピュータが操作するプレイヤーでゴールを守る。また、アタッカーとサポーターの体力が0になったときに、身代わりになる。  
ディフェンダーが打ったパッドは遅くなる。

### 1.3.3 サポーター（人間）

サポーターは、アタッカーとディフェンダーの体力回復をするプレイヤーである。サポーターが攻撃を受けると、ディフェンダーとアタッカーの体力が回復する。サポーターの体力は、時間が経つと回復する。

## 1.4 ゲームの流れ

各クライアントはサーバーに接続する。接続した順に 1P, 2P, 3P, 4P となる。「1P と 2P」「3P と 4P」がチームになる。各プレイヤーは「アタッカー」か「サポーター」を選択するが、もし被ってしまった場合はランダムに選択される。1P はゲーム開始前に「何点先取で勝ちにするか」などのルール設定をする。設定が終わると、カウントダウンと共にゲームが始まる。

各プレイヤーは左右方向に移動し、パッドを跳ね返す。ただ単に跳ね返すだけでなく、仲間と協力をしたり、必殺技を使うことで点数を取る。

点数を取ると、体力と必殺技ゲージはリセットされる。点数を取られた方からパッドを発射する。開始前に設定した点数に達したほうが勝ちとなる。

## 1.5 世界観

みなさんは、サンタさんがクリスマス以外は何をして過ごしていると思いますか？長年にわたる調査の結果、僕たちはエアホッケーをしていることを突き止めました。そのエアホッケーは、普通の人間たちがプレイしているものとは全然違ったものでした。私達はサンタさんがプレイしている珍しいエアホッケーをゲームで再現することにしました。最後に、皆さんに忠告があります。このゲームをクリスマスの日にプレイしないでください。エアホッケーに熱中したサンタさんがプレゼントを届けることを忘れてしまいます。

## 2 構造体とモジュールの説明

### 2.1 構造体

構造体 PLAYER

データ型	変数名	内容
int	type	プレイヤーの種類 (0:アタッカー 1:サポーター 2:ディフェンダー)
int	hp	プレイヤーの体力
int	ap	必殺技ポイント
int	x	プレイヤーの X 座標

構造体 PAD

データ型	変数名	内容
int	speed	パッドの速度
int	x	パッドの X 座標
int	y	パッドの Y 座標

## 2.2 モジュールの説明

1. ネットワークモジュール  
サーバ・クライアント間の通信に関する処理を行う。
2. ゲーム処理モジュール  
ゲームパッドからの入力を受け取る・パッドの動きを計算する。
3. グラフィック・サウンドモジュール  
ゲーム画面・ステータス表示を描画する。また、BGM や効果音等の処理も行う。

## 2.3 各モジュールの外部関数

### 2.3.1 サーバー

1. ネットワークモジュール

void setup_server(u_short port)		
機能	サーバーの初期設定を行う	
引数	port	ポート番号
返回值	なし	

void recv_data(int cid, void *data, int size)		
機能	クライアントからデータを受信する	
引数	cid	クライアント ID
	data	受信データ
	size	受信データのサイズ
返回值	なし	

void send_data(int cid, void *data, int size)		
機能	クライアントへデータを送信する	
引数	cid	クライアント ID
	data	送信データ
	size	送信データのサイズ
返回值	なし	

void terminate_server(void)	
機能	クライアントとの接続を切断する
引数	なし
返回值	なし

## 2. ゲーム処理モジュール

void field_set(void)	
機能	受信したデータを元に、パッドの動きを計算する
引数	なし
返回值	なし

### 2.3.2 クライアント

#### 1. ネットワークモジュール

void client_server(char *server_name, u_short port)	
機能	クライアントの初期設定を行う
引数	server_name      サーバのホスト名 port              ポート番号
返回值	なし

void recv_data(void *data, int size)	
機能	サーバからデータを受信する
引数	data      受信データ size      受信データのサイズ
返回值	なし

void recv_data(void *data, int size)	
機能	サーバへデータを送信する
引数	data      送信データ size      送信データのサイズ
返回值	なし

void terminate_client(void)	
機能	サーバとの接続を切断する
引数	なし
返回值	なし

## 2. ゲーム処理モジュール

void gamepad_input(void)	
機能	ゲームパッドの入力を受け取る
引数	なし
返回值	なし

## 3. グラフィック・サウンドモジュール

void draw_field(void)	
機能	ゲーム画面を描画する
引数	なし
返回值	なし

void draw_status(void)	
機能	ゲームのステータスを描画する
引数	なし
返回值	なし

## 3 各々の担当

木村	ネットワーク担当
好川	グラフィック担当
長滝谷・近藤	ゲーム担当

## 4 ガントチャート

