

# Eduon Controller Application 설명서

Xamarin Android Appliction

## UI

### Splash 뷰

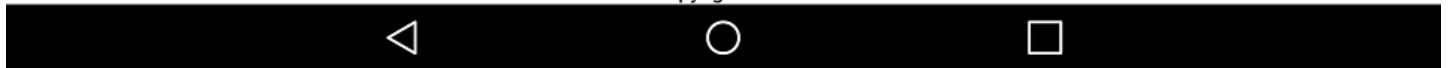
Welcome Controller

DROGEN dev

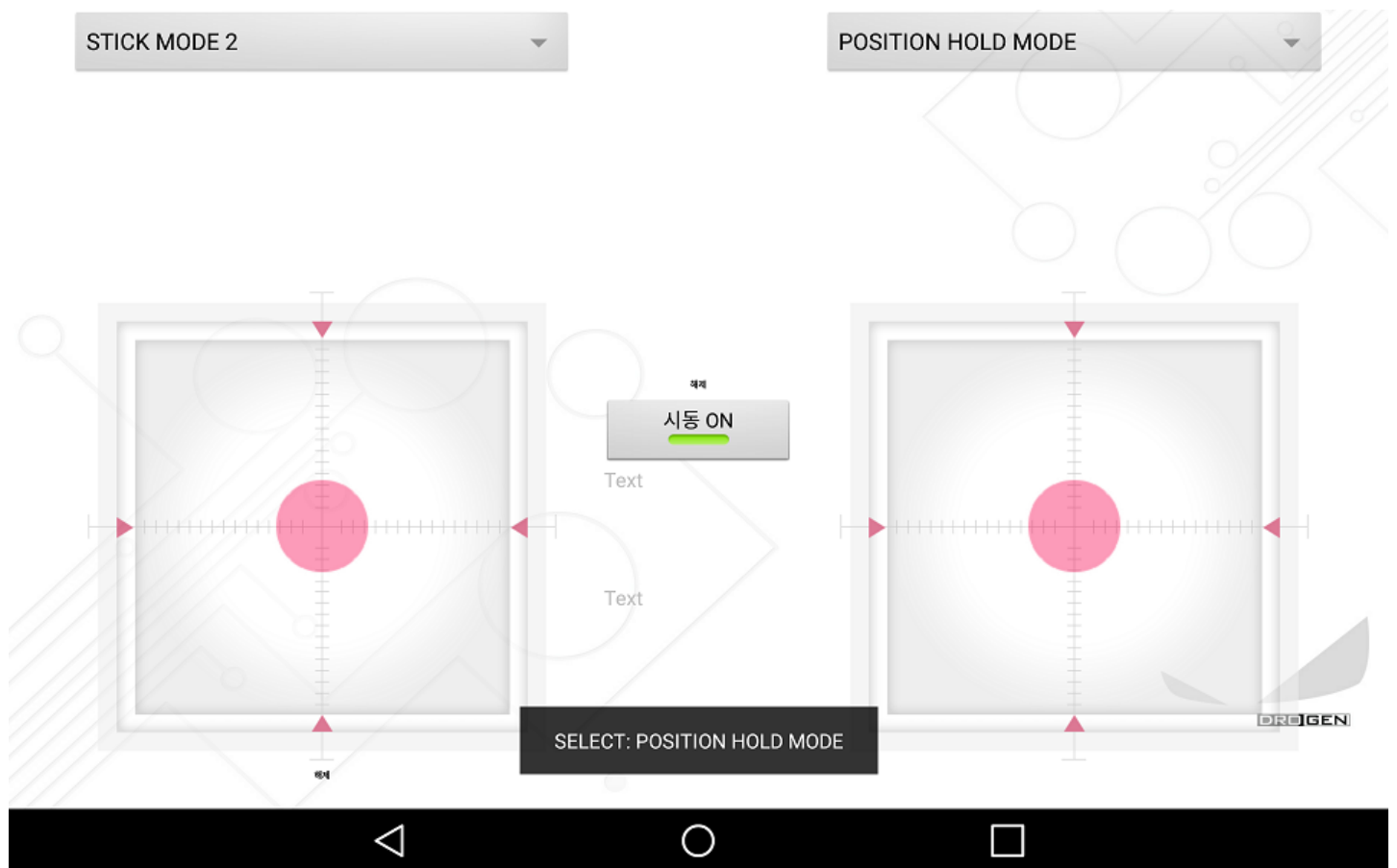


Dual Shork & Android Controller

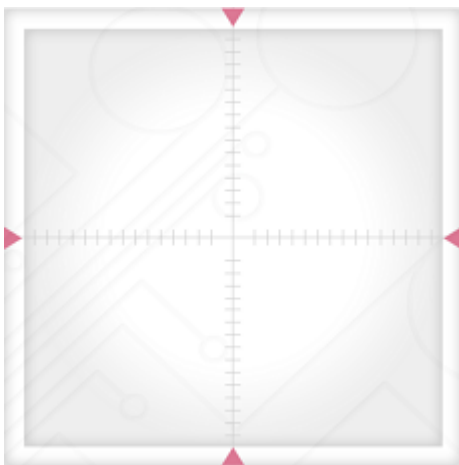
Copyright 2018



### 메인 UI 뷰



- 조종 패드 이미지



- 조종 스틱 이미지



# MainActivity Class

- 어플리케이션 시작 클래스

## Activity 설정

어플리케이션 이름, 로고이미지, 화면테마, 화면 가로세로 설정

```
[Activity(Label = "EDURON", Name = "controller.MainActivity", MainLauncher = true,
    ConfigurationChanges = ConfigChanges.Orientation | ConfigChanges.KeyboardHidden | Cor
    Theme = "@style/Theme.AppCompat.Light.NoActionBar", ScreenOrientation = ScreenOrienta
```

- 이름: Eduron
- 테마: NoActionBar
- 화면: 가로
- 로고: Logo.png(Drogen Img)

## Splash

Application 실행 시 인트로 화면 역할

```
config = new EasySplashScreen(this)
    .WithFullScreen()
    .WithTargetActivity(Java.Lang.Class.FromType(typeof(controller.ControllerActivity)))
    .WithSplashTimeOut(1500) //1.5sec
```

- Splash 뷰 적용(Component: EasySplashScreen)
- 1.5초 동안 위와 같이 설정한 텍스트와 이미지 출력 후 **ControllerActivity** 로 화면전환

## ControllerActivity Class

- 8채널 데이터 (T, A, E, R, Aux1, Aux2, Aux3, Aux4) 처리
- 메인 UI 뷰와 axml 연결해 주는 클래스
- axml의 엘리먼트를 이벤트 등록 및 제어
- 터치 이벤트 처리
- 조이스틱 이벤트 처리

## OnGenericMotionEvent

```
override bool OnGenericMotionEvent(MotionEvent e)
```

- 조이스틱 조작이 일어날 때마다 실행되는 **이벤트 함수**
- 조이스틱 조작 시 화면에 그리는 조종 스틱 이미지 X, Y의 좌표를 갱신
- 시동 버튼이 ON 일때만, 조종 스틱 이미지 Draw

Parameters	Explanation
e	조이스틱 작동 이벤트 객체

## OnKeyDown

```
override bool OnKeyDown([GeneratedEnum] KeyCode keycode, KeyEvent e)
```

- 조이스틱 버튼을 누를 때 마다 실행되는 **이벤트 함수**
- OnKeyDown = 1, OnKeyUp = 0

Parameters	Explanation
keycode	누르는 조이스틱 버튼 코드 객체
e	이벤트를 발생 시키는 조이스틱 정보 객체

## OnKeyUp

```
override bool OnKeyUp([GeneratedEnum] KeyCode keycode, KeyEvent e)
```

- 조이스틱 버튼을 땔 때 마다 실행되는 **이벤트 함수**
- OnKeyDown = 1, OnKeyUp = 0

Parameters	Explanation
keycode	누르는 조이스틱 버튼 코드 객체
e	이벤트를 발생 시키는 조이스틱 정보 객체

## OnTouchListenerAnonymousInner Class

- 터치 이벤트 리스너 등록 클래스

## OnTouchListenerAnonymousInner 생성자

```
OnTouchListenerAnonymousInnerClass(ControllerActivity outerInstance, VirtualController virtualC
```

- 메인 UI 뷰(ControllerActivity)에서 일어나는 터치 이벤트를 등록

Parameters	Explanation
outerInstance	이벤트가 일어나는 메인 UI 뷰 객체
virtualController	X, Y 좌표 및 조종 스틱 이미지 Draw 객체

## OnTouch

```
bool onTouch(View v, MotionEvent e)
```

- 터치를 할때마다 실행되는 **이벤트 함수**

Parameters	Explanation
v	View 정보 객체
e	터치 이벤트(Down, Move, Up), 터치한 X, Y 좌표 정보 객체

## VirtualController Class

- 메인 UI 뷰의 *조종 패드 이미지* 터치 or 외부디바이스 조종 관리(조종 스틱 이미지 Draw)
- 메인 UI 뷰의 *조종 스틱 이미지* drawable 영역(Min, Max) 및 스틱이 점프하는 범위 설정
  - 조이 패드 이미지* 사이즈에 따라 X, Y의 최대, 최소값 좌표가 유동적 (ex: 사이즈[650 \* 650]  
MIN: 105, MAX: 545)
  - 과격한 스틱 점프 방지를 위해 최대 거리값 50으로 제한

## VirtualController 생성자

```
VirtualController(Context context, ViewGroup layout, int stickResID)
```

- 메인 UI 뷰 *조종 패드 이미지* 설정 및 *조종 스틱 이미지* 설정

Parameters	Explanation
------------	-------------

Parameters	Explanation
context	실행하고있는 현재 App Context 정보
layout	axml과 연결된 <i>조종 패드 이미지</i> 객체
stickResID	<i>조종 스틱 이미지</i> 아이디

## DrawStick

```
void DrawStick(MotionEvent arg1)
```

- 터치되는 X, Y좌표 저장 및 *조종 스틱 이미지* Draw 함수

Parameters	Explanation
arg1	터치 이벤트(Down, Move, Up) 및 X,Y 좌표 정보 객체

- Touch 상태 3가지
  - Touch Down: 화면을 누를 때 발생
  - Touch Move: 화면을 움직일 때 발생
  - touch Up: 화면에서 손을 뗄때 발생

```
if (arg1.Action == MotionEventActions.Down)
{
    //점프방지
    drawRenamed.Position(positionX, positionY);
    Draw();
    touchState = true;
}
```

- Touch Down: 화면을 누를 때 발생
- 점프방지
- 터치 된 지점의 X, Y좌표 저장
- 조종 스틱 이미지* Draw

```
else if (arg1.Action == MotionEventActions.Move && touchState)
{
    drawRenamed.Position(positionX, positionY);
    Draw();
}
```

- Touch Move: 화면을 움직일 때 발생
- Touch Down인 상태에서 움직일 때만 실행
- 터치 된 지점의 X, Y좌표 저장

- 조종 스틱 이미지/Draw

```
else if (arg1.Action == MotionEventActions.Up)
{
    touchState = false;
}
```

- touch Up: 화면에서 손을 뗄때 발생

## JoystickController Class

- 외부 디바이스는 디바이스ID로 관리
- 연결 된 외부 디바이스가 조이스틱 or 게임패드 검증 후 디바이스ID 저장
- 조이스틱 Min, Max 설정(Default: -1 ~ 1) -> (Set: -0.8 ~ 0.8)
- 조이스틱 데이터를 UI좌표로 변환

## DualShock Joystick Controller 조이스틱 중요 키맵

왼쪽 패드 값: X,Y 오른쪽 패드 값: RX, RY (각 X,Y축 범위는 -1 ~ 1)

```
//왼쪽 조종 스틱
public static Axis AXIS_X = MotionEvent.AxisFromString("AXIS_X");
public static Axis AXIS_Y = MotionEvent.AxisFromString("AXIS_Y");
//오른쪽 조종 스틱
public static Axis AXIS_RX = MotionEvent.AxisFromString("AXIS_RX");
public static Axis AXIS_RY = MotionEvent.AxisFromString("AXIS_RY");
//4개 버튼
public static Keycode BUTTON_A = Keycode.ButtonB;
public static Keycode BUTTON_B = Keycode.ButtonC;
public static Keycode BUTTON_C = Keycode.ButtonY;
public static Keycode BUTTON_X = Keycode.ButtonA;
```

## JoystickController 생성자

JoystickController(VirtualController LeftPad, VirtualController RightPad)

- 좌, 우 조종 패드 이미지의 각 X, Y축 좌표 저장을 위해 입력

Parameters	Explanation
LeftPad	UI 왼쪽 패드

Parameters	Explanation
RightPad	UI 오른쪽 패드

## CheckGameControllers

```
void CheckGameControllers(int[] getDevicesID)
```

- InputManager로 저장된 Device List 중에 Joystick or Gamepad 체크 후 디바이스ID 등록

Parameters	Explanation
getDevicesID	InputManager에 저장된 있는 Device ID 목록

## KeyMapping

```
void KeyMapping(MotionEvent e)
```

- 조이스틱 LeftStick RightStick 각 X, Y축 데이터의 범위(-1 ~ 1)를 **Draw 가능한 좌표로 변환**
- *조종패드 이미지* 크기에 따라 Min, Max 값 결정(Ex 사이즈(650 \* 650)경우 MIN: 105, MAX: 545 )

Parameters	Explanation
e	조이스틱 X, Y축 정보를 가지고 있는 객체

## IsGamepad

```
IsGamepad(InputDevice device)
```

외부장치 device가 Joystick or Gamepad 체크 후 True or False 리턴

Parameters	Explanation
device	안드로이드 시스템과 연결된 외부 장치 객체