eBeam Marker

Developer Guide

for Android

Luidia R&D S/W

2017. 11

Development > Project setting

- Copy eBeamMarkerSDK-v1.0-20180504-release.aar to libs/

- Add Permission and service in the AndroidManifest.xml

```
 - <uses-permission android:name="android.permission.WAKE_LOCK" />
 - <uses-permission android:name="android.permission.BLUETOOTH" />
 - <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />

 - <service
     android:name="com.luidia.ebeam.sdk.UartService"
     android:enabled="true" />
```

Development > Components of Test Sample (PenTest)

Folder		File	Description
$(SDKHome)/src/com/pnf/pen	calibration/	CalibrationMarkerActivity.java	Calibration Activity
	device/	PNFBluetoothViewActivity.java	Device Connect Activity
		DeviceChangeNameViewActivity.java	Device Name Change Activity
		DeviceChangeAudioViewActivity.java	Device Audio Change Activity
	drawingview/	DrawView.java	Drawing View Class
		DrawViewActivity.java	Drawview Activity
	test/	BaseActivity.java	Base Activity
		MainActivity.java	Main Activity
		MainDefine.java	Define Class
$(SDKHome)/res/layout		activity_main.xml	Main layout
		baseview.xml	Base layout
		activity_calibration_marker.xml	Layout for calibration
		bluetooth_connect_view.xml	Layout for connect
		activity_device_change_name_view.xml	Layout for change name
		activity_device_change_audio_view.xml	Layout for change audio
		drawview.xml	Layout for drawing

Methods

new		
Description	Create object of EBeamPenController	
out	N/A	
input	Context	Context of application
Device	eBeam Marker	
Usage	@Override public void onCreate(Bundle savedInstanceState) { ..... MainDefine.penController = EBeamPenController.create(getApplicationContext()); MainDefine.penController. setModelCode(ModelCode.eBeamSmartMaker); ...... }	

Development > Reference

getModelCode		
Description	Get Connected device	
Out	int	
input	N/A	
Device	eBeam Marker	

Usage	public void onActivityResult(int requestCode, int resultCode, Intent data)
	{
	.....
	if(requestCode == REQUEST_BLUETOOTH_CONNECT) {
	modelCodeValueTextView.setText(String.valueOf(MainDefine.penController.getModelCode()));
	hwVerValueTextView.setText(""+MainDefine.penController.getHWVersion());
	nameValueTextView.setText(""+MainDefine.penController.getStationName());
	}
	......
	}	

setPenEventListener		
Description	Set an object to receive the pen data	
Out	Void	
input	PenEventListener	Object pointer to receive the pen data
Device	eBeam Marker	

Usage	protected void onResume() {
	super.onResume();
	
	MainDefine.penController.setPenEventListener(new PenEventListener() {
	@Override
	public void onPenEvent(int what, int arg1, int arg2, Object obj) {
	
	}
	})
	}	

Development > Reference

setRetObjForMsg		
Description	Set an object to receive the pen data for Message	
Out	Void	
input	PenMessageListener	
Device	eBeam Marker	

Usage	protected void onResume() {
	super.onResume();
	
	MainDefine.penController.setPenMessageListener(new PenMessageListener() {
	@Override
	public void onPenMessage(int what, int arg1, int arg2, Object obj) {
	
	}
	})}	

Development > Reference

setCalibrationSendStart		
Description	Set data for calibration	
Out	Void	
input	N/A	
Device	eBeam Marker	

Usage	void setCalibrationSendStart(){
	MainDefine.penController.startCalibMode();
	……
	}	

Development > Reference

setCalibrationSendStop		
Description	Set data for calibration	
Out	Void	
input	N/A	
Device	eBeam Marker	

Usage	void setCalibrationSendStop(){
	        MainDefine.penController.stopCalibMode();
	        ……
	}	

Development > Reference

setCalibrationSendData		
Description	Set data for calibration	
Out	Void	
input	DeviceDirection	DEVICE_DIRECTION_LEFT
	PointF	Calibration Raw data
Device	eBeam Marker	

Usage	void coordinateComplete (){
	MainDefine.penController.setCalibrationSendData(
	setCaliPosition,
	m_posRestultPoint[0],
	m_posRestultPoint[3],
	m_posRestultPoint[2],
	m_posRestultPoint[1]);
	……
	}	

Development > Reference

setChangeNameSend		
Description	Set data for Name Change	
Out	Void	
input	changeDeviceName	Device name
Device	eBeam Marker	

Usage	void setChangeNameSend (){
	MainDefine.penController. setDeviceNameData(changeDeviceName);
	……
	}	

Development > Reference

setDeviceAudioData		
Description	Set data for calibration	
Out	Void	
input	DeviceAudio	DEVICE_AUDIO_LANG_ENGLISH
Device	eBeam Marker	
Usage	MainDefine.penController.setDeviceAudioData(audioCode);	

Development > Reference

getCoordinatePosition		
Description	Get calibrated coordinates	
Out	PointF	
input	N/A	
Device	eBeam Marker	

Usage	void onPenEvent(int what, int RawX, int RawY ,Object obj)
	{
	.....
	PointF ptConv = MainDefine.penController. getCoordinatePosition(RawX,RawY,penData.bRight);
	.....
	}	

Development > Guide > Connect and Initialize

➢ Overview

Create and initialize object EBeamPenController

➢ Example

1. Create EBeamPenController object

```
MainDefine.penController = EBeamPenController.create(getApplicationContext());
```

2. Set Default Device Model

```
MainDefine.penController.setDefaultModelCode(PNFDefine.eBeamSmartMaker);
```

example source: MainActivity.java, BaseActivity.java

Development > Guide > Receive data from library

➢ Overview

Internally EBeamPenController is supposed to call selector named as “PenEventListener” of object set by “setPenEventListener” whenever the pen moves.

➢ Example

```
void onPenEvent(int what, int RawX, int RawY ,Object obj)
{
                PenData penData = (PenData)obj;
                PointF ptConv = MainDefine.penController.GetCoordinatePostionXY(RawX ,RawY ,penData.bRight);

                switch(what)
                {
                case PenEvent.PEN_DOWN:
                    switch(penData.MakerPenStatus){
                                case PenEvent.MARKERPEN_RED_MARKER:
                                break;
                                case PenEvent.MARKERPEN_GREEN_MARKER:
                                break;
                                case PenEvent.MARKERPEN_YELLOW_MARKER:
                                break;
                                case PenEvent.MARKERPEN_BLUE_MARKER:
                                break;
                                case PenEvent.MARKERPEN_PURPLE_MARKER:
                                break;
                                case PenEvent.MARKERPEN_BLACK_MARKER:
                                break;
                                case PenEvent.MARKERPEN_ERASER_CAP:
                                break;
                                case PenEvent.MARKERPEN_BIG_ERASER:
                                break;
                      }
                break;
                case PenEvent.PEN_MOVE:
                break;
                case PenEvent.PEN_UP:
                break;
                }
                .....
}
```

example source : MainActivity.java

Development > Guide > Receive message from library

➢ Overview

➢ Example

1. Set messageListener

```
        protected Handler messageHandler = new Handler(){
MainDefine.penController.setPenMessageListener(this);

            @Override
            public void handleMessage(Message msg) {
                onMessageEvent(msg.what ,msg.arg1 ,msg.arg2 ,msg.obj);
            }
        };
```

1. Handler for Message

```
private PenMessageListener listener = new PenMessageListener() {

        @Override
        public void onPenMessage(int what, int i1, int i2, Object o) {
                        ......
                    if(what == PenMessage.PNF_MSG_FAIL_LISTENING){
                    }else if(what == PenMessage.PNF_MSG_CONNECTED){
                    }
                    else if(what == PenMessage.PNF_MSG_INVALID_PROTOCOL){
                    }
                    else if(what == PenMessage.PNF_MSG_SESSION_CLOSED){
                    }
                    else if(what == PenMessage.PNF_MSG_PEN_RMD_ERROR){
                    }
                    else if(what == PenMessage.PNF_MSG_FIRST_DATA_RECV){
                    }
                    packetCnt++;
                    updatePacketCnt();
                    ......
        }
}
```

Log String Message	Description
PNF_MSG_CONNECTED	Device is connected
PNF_MSG_FAIL_LISTENING	Fail to receive. Need to reconnect.
PNF_MSG_INVALID_PROTOCOL	Invalid hardware
PNF_MSG_SESSION_CLOSED	Session is disconnected
PNF_MSG_FIRST_DATA_RECV	First data is received after connecting
PNF_MSG_PEN_RMD_ERROR	Abnormal drawing data

example source : MainActivity.java

Development > Guide > Calibration

➢ Overview

Pen coordinates is converted to screen coordinates by projective matrix which is set in the calibration view.

➢ Example

1. Override interface for pen data and message event

```
@Override
void onPenEvent(int what, int RawX, int RawY ,Object obj) {
        …… }

@Override
void onMessageEvent(int what, int RawX, int RawY ,Object obj) {
        …… }
```

2. set handler for pen data

```
MainDefine.penController.setPenEventListener(this);
MainDefine.penController. setPenMessageListener(this);
```

example source : CalibrationPointActivity.java