

DENSO



High-powered Handy Scanner

DENSO SP1 SDK for Android Reference Manual

(E-Paper Tag)

Copyright © 2020 DENSO WAVE INCORPORATED
All rights reserved.

BHT, QR code, iQR code, SQRC and Qbdirect are registered trademarks of DENSO WAVE INCORPORATED.

Bluetooth® is registered trademarks of trademarks rights holders, DENSO WAVE INCORPORATED use Bluetooth® based on license.

All products and company names in this document are trademarks of their respective holders.
™, ® mark is not always specified in this document.

Contents

0. Introduction	2
1. Precaution	3
1.1. Precautions for RF Tag Processing.....	3
1.2. Precautions for the response from the scanner.....	3
1.3. Precautions for the authentication with tags	3
1.4. Precautions for E-Paper Tag	3
2. Application Development environment.....	5
3. Recommended scanner settings.....	5
3.1. SP1	5
4. RFID	6
4.1. How to use E-Paper Tag.....	6
Update display of E-Paper Tag.....	6
4.2. E-Paper API	7
RFIDScanner#getEPD	7
RFIDScanner#EPD#createWorkPanel.....	8
RFIDScanner#EPD#editWorkPanel.....	9
RFIDScanner#EPD#savePanel	11
RFIDScanner#EPD#loadPanel	12
RFIDScanner#EPD#downloadPanel	13
RFIDScanner#EPD#uploadPanel	14
RFIDScanner#EPD#writeEpaper	15
EPDBitmap class.....	17
5. Sample code	18
5.1. Update display of E-Paper Tag	18

0. Introduction

This manual explains how to use the RFID scanner function to develop applications for E-Paper tags. This manual is written for people who have programming experience using Android Studio. For information on how to use and develop Android products and the grammar of each language, refer to the commercial books.

The included library may have functions that are planned to support in the future. However, the functions not described in this manual are not guaranteed.

The information in this document is subject to change without notice.

For information on the common functions of RFID scanners, see "Denso SP1 SDK for Android Reference Manual".

1. Precaution

1.1. Precautions for RF Tag Processing

Since contactless communication is performed with RF tags, it is necessary to keep the following precautions in mind.

- (1) If a RF tag does not respond due to its defect or it does not receive sufficient power due to wave interference, the scanner may be unable to read it even if it is located inside the reading area.
- (2) If an irrelevant RF tag is located inside the reading area, the scanner may read it.
- (3) In case of iterative reading, RF tags shall be read again if it still remain inside the reading area.
- (4) If two or more RF tags have the same Ull, they will be handled as a single RF tag.

Therefore, you may get a different result than expected if you perform the reading process in a way such as to complete it for the required number of tags. To solve this problem, take the measures described below for the upper-level application as required to read from RF tags.

In cases (1) to (3):

- Constructing a system capable of checking the number of read RF tags and the Descriptions. The results of reading tags (number of RF tags and Descriptions) should be checked by the operator. Make sure that unread RF tags are read by the scanner or registered separately.

In case (4):

- Do NOT use RF tags having the same Ull data.

1.2. Precautions for the response from the scanner

Depending on the host device and environment, the response from the scanner may be delayed. Make sure that the scanner can communicate without problem in advance.

1.3. Precautions for the authentication with tags

When the access password for openRead() and writeOneTag() is set to "00-00-00-00", Reading process and Writing process are executed even when the access password is set for the tag. (RFID control is performed without performing authentication process.)

1.4. Precautions for E-Paper Tag

It is necessary to pay attention to the following points due to the characteristics of E-Paper Tag.

- E-Paper Tag uses the area of USER bank after 4096(1000h) to rewrite the display.
- Do not lock the USER bank permanently.
E-Paper display cannot be rewritten.
- Rewriting the display of E-Paper Tag takes longer time than writing normal tags.
Specify a timeout value of writeEpaper method with margin.

■ Update history

Version	Description	Supported SDK Ver.	Supported OS Ver.
1st	-	1.0.7	1.12

2. Application Development environment

Refer to the "Denso SP1 SDK for Android Reference Manual".

3. Recommended scanner settings

The recommended settings for each scanner when using the E-Paper Tag are as follows.
(Refer to the SDK manual for the settings for using the SDK.)

3.1. SP1

Recommended RFID Settings	
powerLevelRead / powerLevelWrite	SP1 :30dBm
polarization	Vertical or Horizontal

4. RFID

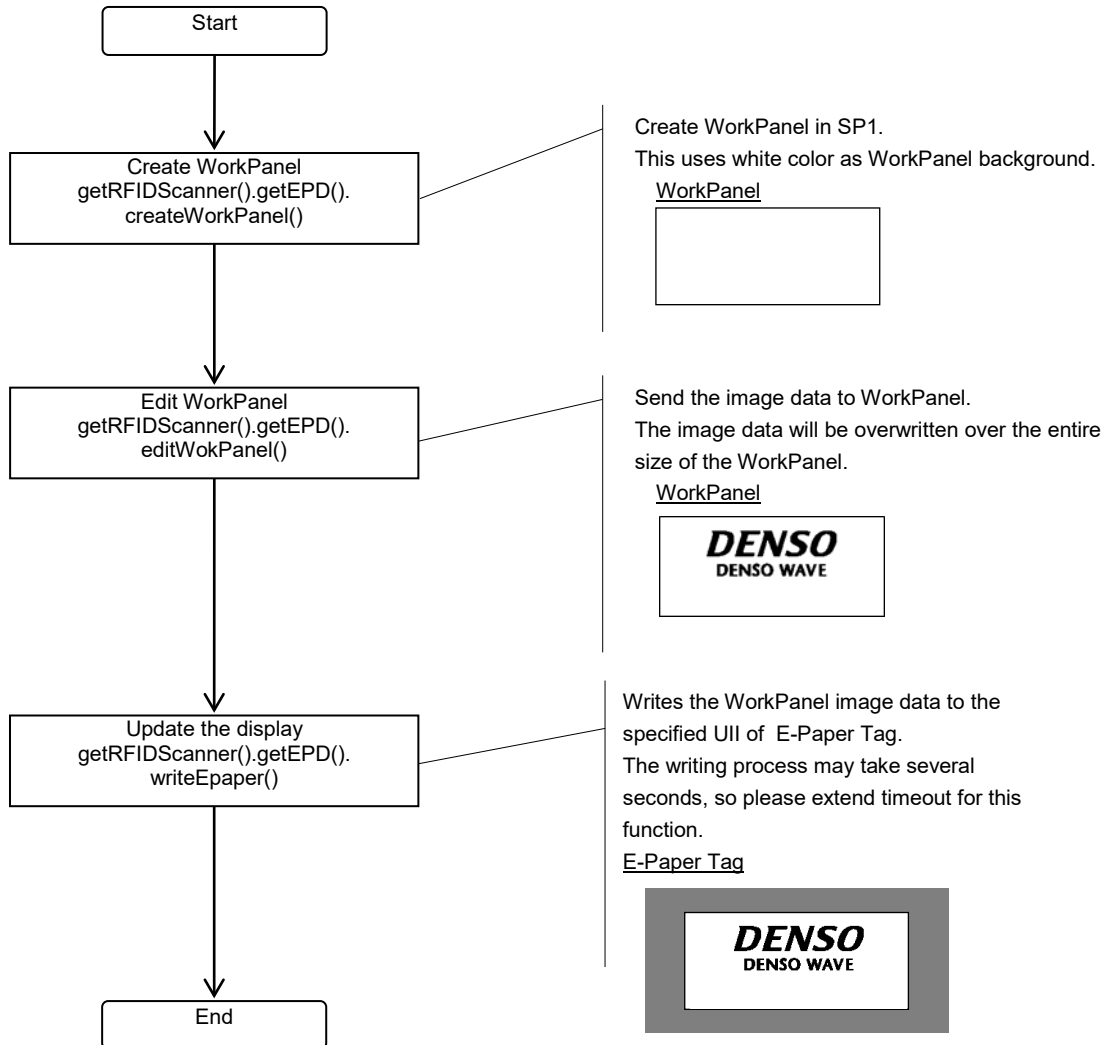
4.1. How to use E-Paper Tag

An example of the procedure of rewriting E-Paper Tag is as follows.

Update display of E-Paper Tag

To update the display of E-Paper Tag, follow the steps below.

(The flow chart is described with API argument and CONTINUOUS2 trigger mode is used)



4.2. E-Paper API

RFIDScanner#getEPD

Description

Get EPD instance

Syntax

```
EPD getEPD ();
```

Parameters

None

Return value

Get the EPD instance generated by CommScanner#claim.

Throws

This API does not throw an exception (no error occurs).

If you call this API before #claim method, the API returns Null.

RFIDScanner#EPD#createWorkPanel

Description

Create a new WorkPanel for E-Paper Tag.

When new WorkPanel is generated, the previous WorkPanel information shall be deleted.

Syntax

```
void createWorkPanel (  
    int width,  
    int height,  
    RFIDColor color  
);
```

Parameters

width

WorkPanel horizontal size(1 to 296)

height

WorkPanel vertical size(1 to 128)

color

RFIDColor.WHITE Create WorkPanel with white background

RFIDColor.BLACK Create WorkPanel with black background

Return value

None

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.INVALID_STATUS	Invalid status error
ErrorCode.NOT_SUPPORT_COMMAND	unsupported command

Notes

This method does not check the maximum size.

RFIDScanner#EPD#editWorkPanel

Description

Overwrite WorkPanel area for E-Paper Tag with the specified image data.

The specified color can be treated as a transparent color.

Syntax

```
void editWorkPanel (  
    int x,  
    int y,  
    int width,  
    int height,  
    RFIDColor tcolor,  
    byte[] data  
);
```

Parameters

x

Drawing start X coordinate(0 to 295)

y

Drawing start Y coordinate(0 to 127)

width

Image horizontal size (1 to 296)

height

Image vertical size (1 to 128)

tcolor

RFIDColor.NONE Overwrite image data(no transparency)

RFIDColor.WHITE Treat white as transparent color

RFIDColor.BLACK Treat black as transparent color

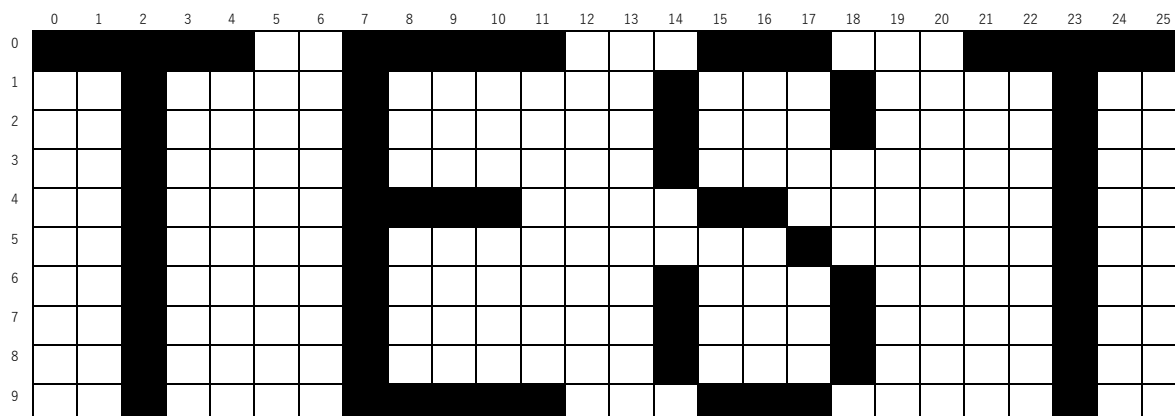
data

Image data

Image data format:

The data is stored pixel by pixel in MSB in horizontal order, begin from the upper left corner at coordinate (0,0). 0 stands for white, 1 stands for black. If the last pixel of the row is reached, the data shall be stored continuously into the next row. The redundant pixel shall be set to 0.

In the case of the following image data (26 width, 10 height), the specified binary data is as follows.
F9h,F1h,C7h,C8h,40h,88h,42h,10h,22h,10h,84h,08h,04h,21h,E1h,81h,08h,40h,10h,42h,10h,22h,10h,84h,08h,84h,21h,02h,21h,08h,7Ch,70h,40h (The last 4 bits are padding data.)



Return value

None

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.INVALID_COMMAND	Invalid parameters or syntax
ErrorCode.INVALID_STATUS	Invalid status error
ErrorCode.NOT_SUPPORT_COMMAND	unsupported command

Notes

This method does not check the maximum size.

RFIDScanner#EPD#savePanel

Description

Save WorkPanel data to SP1. It is retained even if the power of SP1 is turned off.

Syntax

```
void savePanel (  
    int index,  
);
```

Parameters

index

Storage number (1 to 4 can be specified for SP1)

Return value

None

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.INVALID_COMMAND	Invalid parameters or syntax
ErrorCode.INVALID_STATUS	Invalid status error
ErrorCode.NOT_SUPPORT_COMMAND	unsupported command
ErrorCode.EXECUTE	ROM save failed

RFIDScanner#EPD#loadPanel

Description

Load the panel data saved in SP1 to WorkPanel.

Syntax

```
void loadPanel (  
    int index,  
);
```

Parameters

index

Read storage number (1 to 4 can be specified for SP1)

Return value

None

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.INVALID_COMMAND	Invalid parameters or syntax
ErrorCode.INVALID_STATUS	Invalid status error
ErrorCode.NOT_SUPPORT_COMMAND	unsupported command

RFIDScanner#EPD#downloadPanel

Description

Download image data directly to the specified area.

Syntax

```
void downloadPanel (  
    int index,  
    int width,  
    int height,  
    byte[] data  
);
```

Parameters

index

Download area number 0: WorkPanel
1 to 4: 1 to 4 Storage number

width

Image horizontal size (1 to 296)

height

Image vertical size (1 to 128)

data

Image data

Return value

None

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.INVALID_COMMAND	Invalid parameters or syntax
ErrorCode.NOT_SUPPORT_COMMAND	unsupported command
ErrorCode.EXECUTE	ROM save failed

Notes

This method does not check the maximum size.

RFIDScanner#EPD#uploadPanel

Description

Get WorkPanel image data or saved image data in SP1.

Syntax

```
EPDBitmap uploadPanel (  
    int index  
);
```

Parameters

index

Area number 0: WorkPanel

1 to 4: 1 to 4 Storage number

Return value

EPDBitmap

WorkPanel or saved image data in SP1

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.INVALID_COMMAND	Invalid parameters or syntax
ErrorCode.INVALID_STATUS	Invalid status error
ErrorCode.NOT_SUPPORT_COMMAND	unsupported command

RFIDScanner#EPD#writeEpaper

Description

Writes WorkPanel data to one specified Ull E-Paper Tag.

This is a synchronous function. Throws an exception if an error occurs.

Trigger mode needs to be AUTO_OFF or CONTINUOUS2.

Syntax

```
void writeEpaper (  
    int type, byte[] pwd, byte[] UII, long timeout  
);
```

Parameters

type

Specify 0

pwd

Access password

UII

Target tag's Ull

timeout

Timeout value (millisecond)

Return value

None

Throws

RFIDException

RFIDException#getErrorCode() can get the error.

The errors of this API are as follows.

Error label	Description
ErrorCode.COMMUNICATION_TIMEOUT	Communication timeout
ErrorCode.INVALID_STATUS	Invalid status error
ErrorCode.NOT_SUPPORT_COMMAND	Unsupported command
ErrorCode.SCANNER_CHARGING	Error while charging
ErrorCode.INVALID_COMMAND	Invalid parameters or syntax
ErrorCode.ACCESS_OUT_OF_RANGE_MEMORY	Access out of range memory
ErrorCode.SCANNER_ACCESS_LOCK_MEMORY	Access lock memory
ErrorCode.SCANNER_POWER_SHORTAGE_OF_WRITE_MEMORY	Writing power is not enough
ErrorCode.SCANNER_WRITE_LOCK_KILL_TIMEOUT	Write or Lock or Kill's timeout
ErrorCode.SCANNER_WRITE_LOCK_KILL_UNKNOWN	Write or Lock or Kill unknown error
ErrorCode.SCANNER_TRIGGER_MODE	Trigger mode error (Implement WRITE or LOCK or KILL in unsupported trigger mode)
ErrorCode.SCANNER_TEMP_ERROR	Temperature error. After a while, write E-Paper Tag again
ErrorCode.SCANNER_EPD_LOST	The target E-Paper Tag is lost during the writing process
ErrorCode.SCANNER_EPD_CHARGE_TIMEOUT	Timeout of power supply to E-Paper Tag
ErrorCode.SCANNER_EPD_OPERATION_TIMEOUT	E-Paper Tag processing timeout

EPDBitmap class

Description

E-Paper image data class

(✓:Support)

Property : description	Setting value		SP1
EPDBitmap			
width : Hroizontal size	1- 296(int)	Horizontal size	✓
height :Vertical size	1- 128(int)	Vertical size	✓
data :Image data	byte[]	Image data	✓

5. Sample code

5.1. Update display of E-Paper Tag

```
try {
    // create WorkPanel
    scanner.getRFIDScanner().getEPD().createWorkPanel(296,128, RFIDColor.WHITE);

    // Edit WorkPanel
    byte[] image = new byte[] { /* specify byte dat of image */ };
    scanner.getRFIDScanner().getEPD().editWorkPanel(0, 0, 296, 128, RFIDColor.NONE,image);

    // Update display(Writing images on E-Paper Tag)
    byte[] pwd = { 0, 0, 0, 0 }; // No authentication
    byte[] uii = { /* Specify UII of tag */ };

    scanner.getRFIDScanner().getEPD().writeEpaper(0, pwd, uii, 15000);
} catch (RFIDException e) {
    e.printStackTrace();
}
```

DENSO SP1 SDK for Android Reference Manual(E-Paper Tag)

1st Edition. Mar 2020

DENSO WAVE INCORPORATED AUTO-ID Business Unit

No part of this publication may be reproduced in any form or by any means without permission in writing from the publisher

Specifications are subject to change without prior notice.

In no event will DENSO WAVE be liable for any direct or indirect damages resulting from the application of the information in this manual.