Application Note
for
12” Spectra
With
the cascaded G2.1 iTC
(MTP LUT)

Description	**Interface for the 12” Spectra EPD with the cascaded G2.1 iTC**
Date	**2017/11/6**
Doc. No.	
Revision	**01**

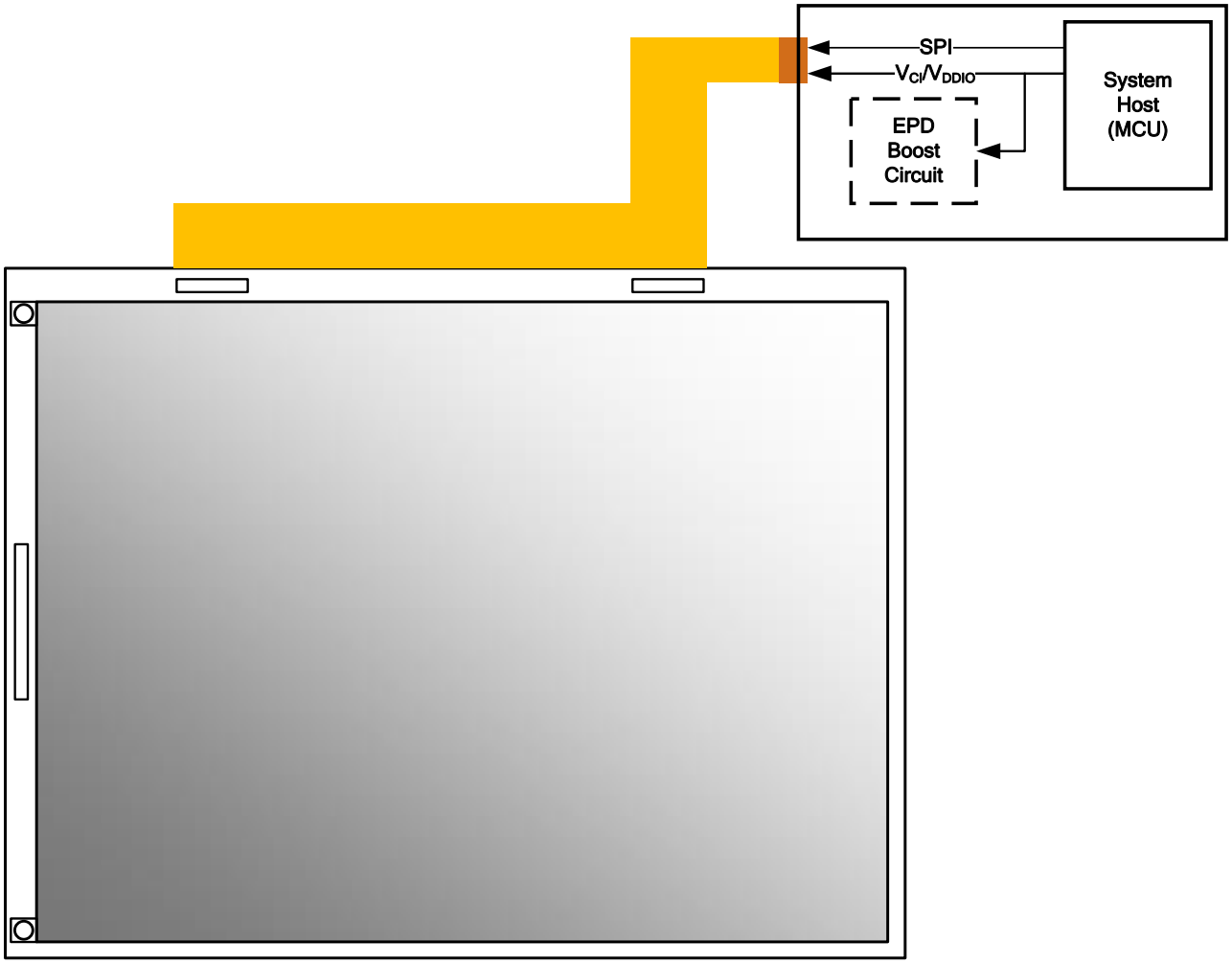
4F, No. 28, Chuangye Rd., Tainan Science Park, Tainan City 74144, Taiwan (R.O.C.)
Tel: +886-6-279-5399 Fax: +886-6-270-5857

Table of Contents

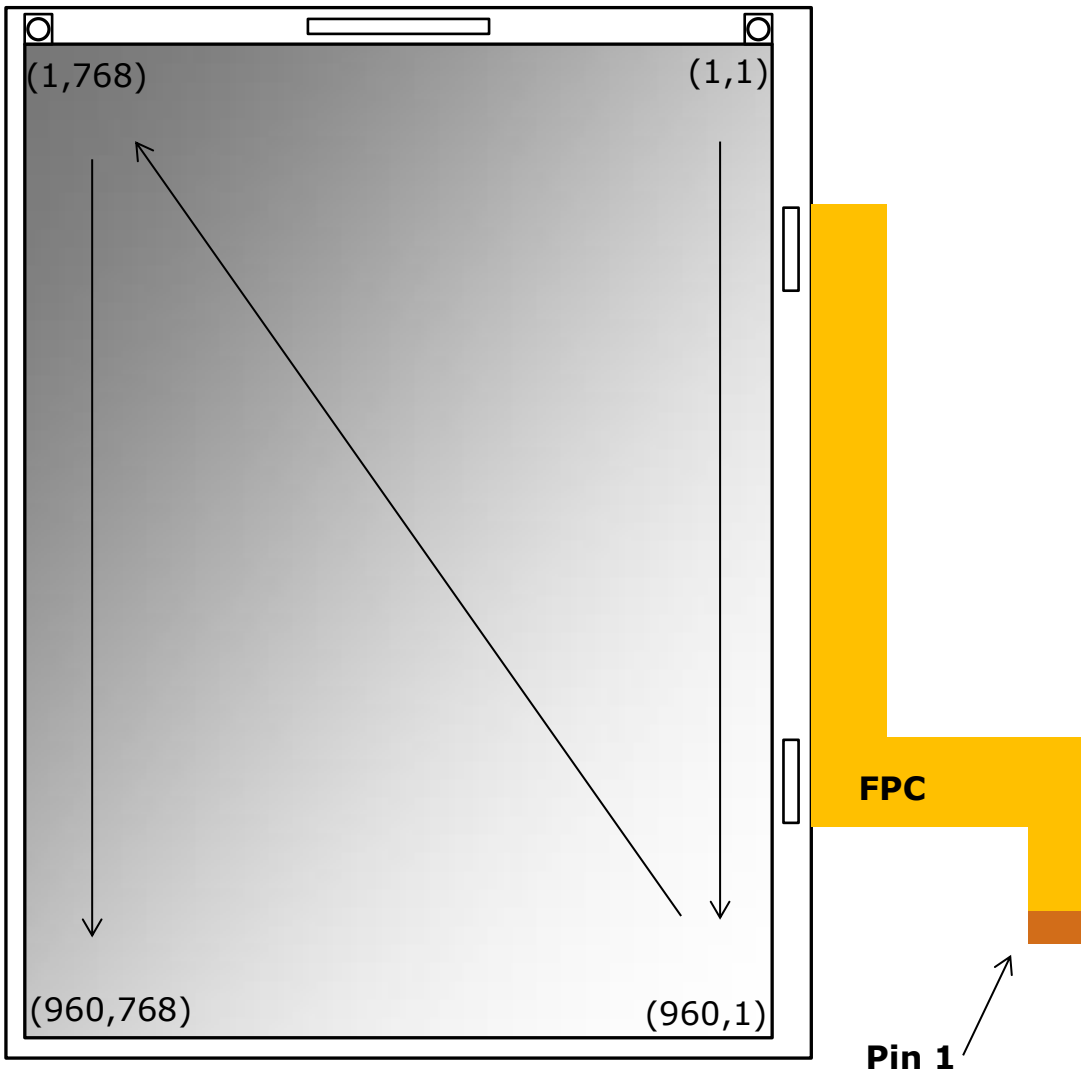
1 General Description

1.1 Overview

The document introduces how to drive the 12” EPD with the cascaded G2.1. The EPD has embedded the Tcon function. The major control interface of the driver is SPI. The host sends both the setting commands and the display image to driver through the SPI bus.



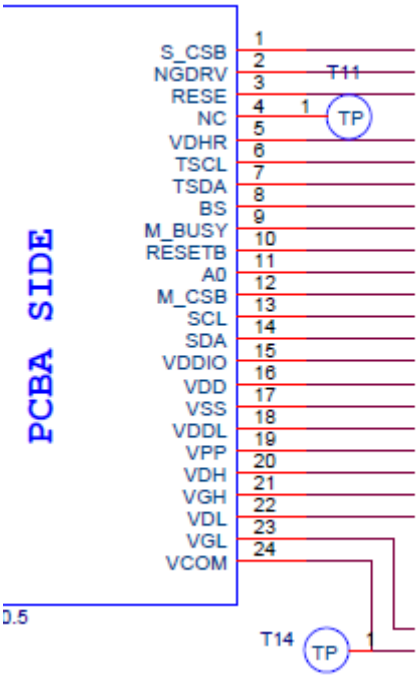
1.2 Panel drawing



1.3 FPC interface

The 12” EPD was mounted two source drivers. They are Master and Slave role respectively and share the same SPI with separate CS.

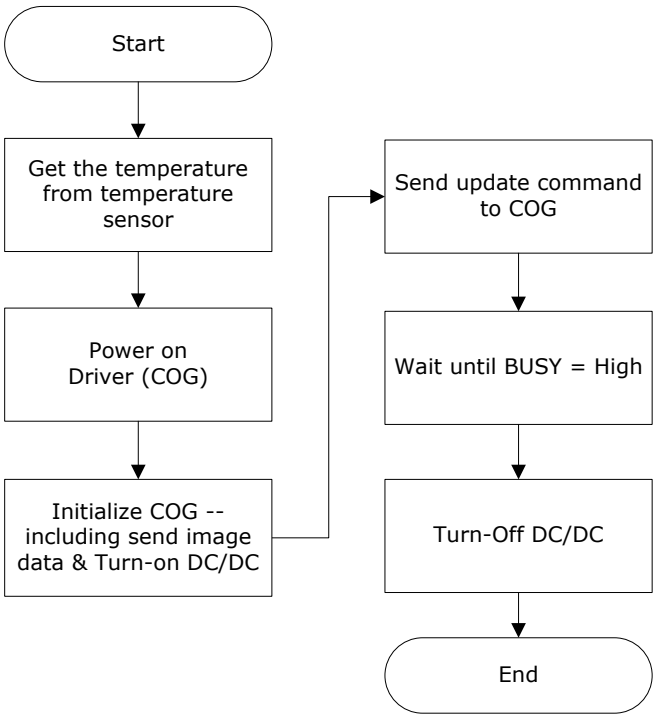
The pin assignment of FPC are as follows,



The 12th pin, M_CSB is the CS of Master. The first pin, S_CSB is the CS of Slave.

1.4 EPD Driving Flow Chart

The flowchart below provides an overview of the necessary actions to update the EPD. The steps below refer to the detailed descriptions in the respective sections.



1.5 SPI Timing Format

SPI commands are used to communicate between the MCU and the COG Driver. The SPI format used differs from the standard in that two way communications are not used. When setting up the SPI timing, PDI recommends verify both the SPI command format and SPI command timing in this section.

The maximum clock speed of the display is **5MHz**.

- Below is a description of the SPI Format:

  SPI(0xI, 0xD_1, 0xD_2, …, 0xD_n, csDS)

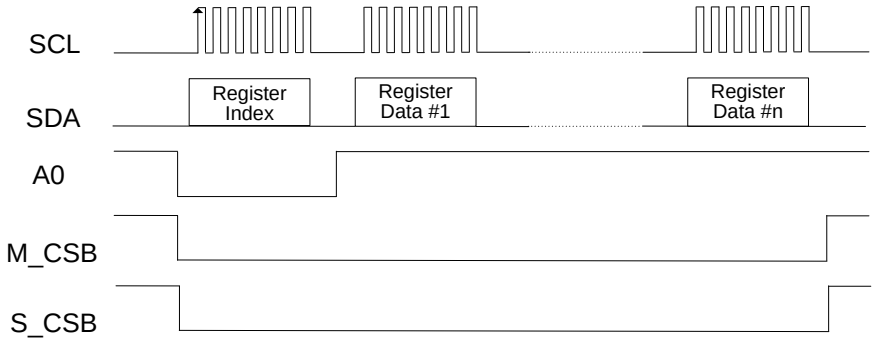
  Where:

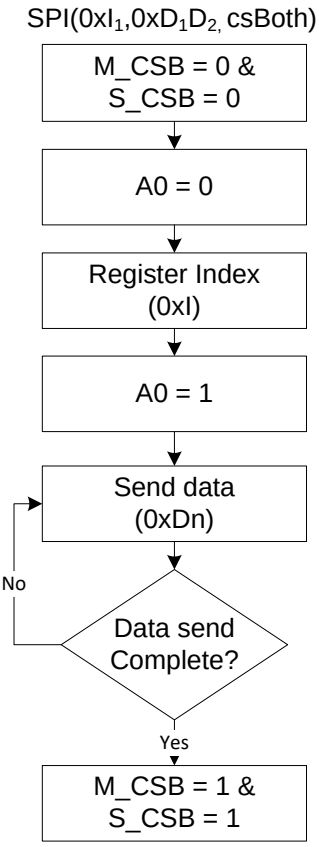
  I is the Register Index and the length is 1 byte
  $D_{1\sim n}$ is the Register Data. The Register Data length is variously.

  The csDS indicates this command is delivered to which driver or both.
  csMaster : only deliver to Master driver
  csSlave : only deliver to Slave driver
  csBoth : deliver to both Master and Slave

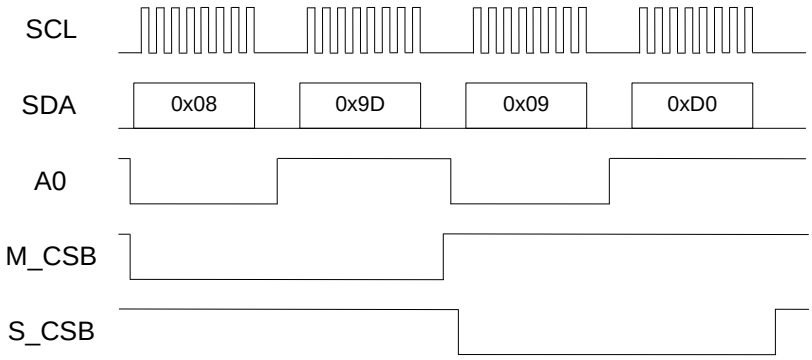
- When SPI sends the Index, the A0 has to pull LOW. When sends the data, the A0 has to pull HIGH. The next page is the detail flow chart.

- SPI command signals and flowchart:



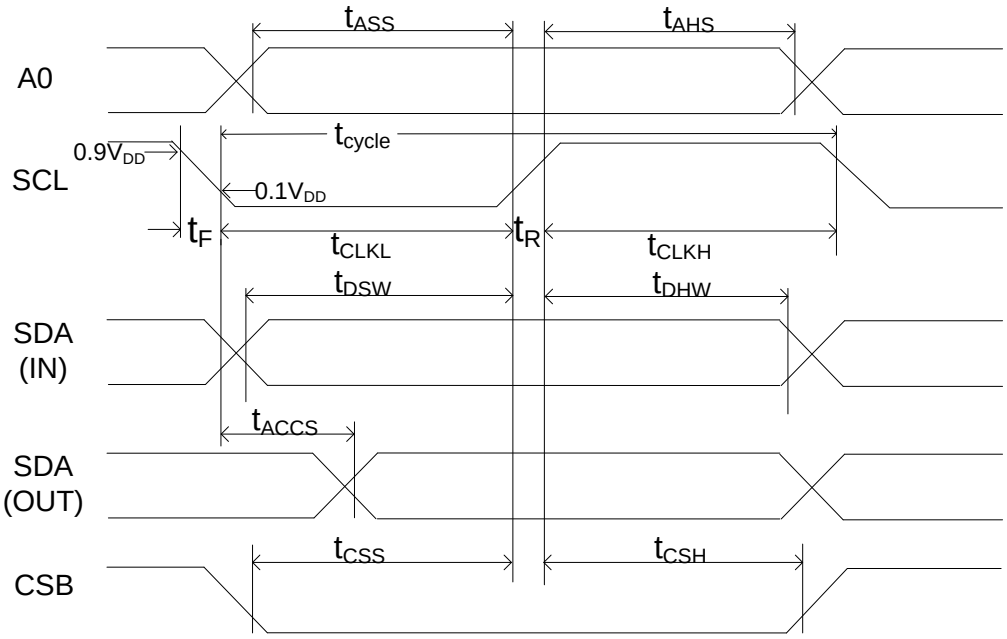


For example:
To send two SPI commands:
SPI((0x08,0x9D, csMaster) and SPI(0x09, 0xD0, csSlave)



If register data is larger than two bytes, you must input data continuously without setting Register Index again.

- SPI command timing



SPI DATA-IN

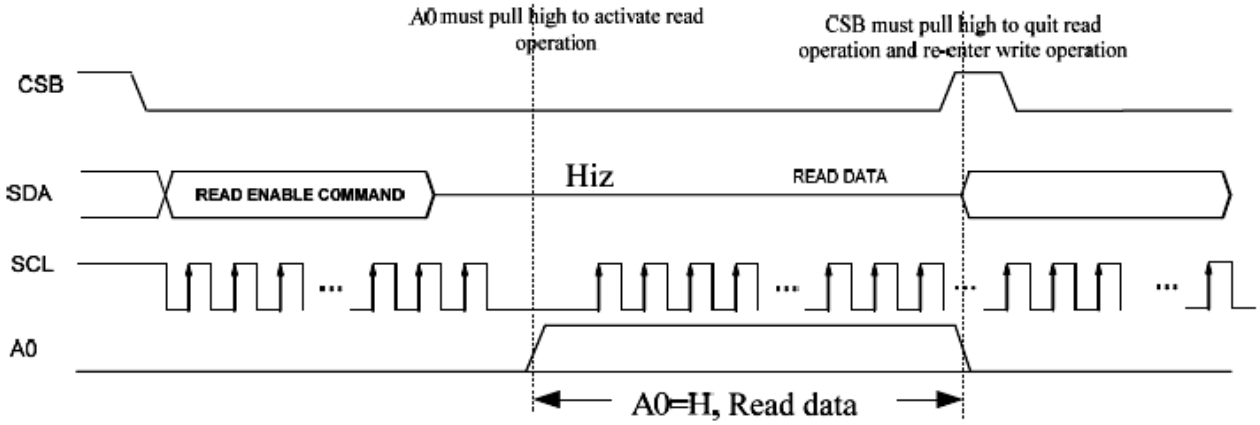
Item	Symbol	Min.	Typ.	Max.	Unit	Remark
Clock Cycle Time	t_{cycle}	200	-	-	ns	
Chip Select Setup Time	t_{CSS}	90	-	-	ns	
Chip Select Hold Time	t_{CSH}	90	-	-	ns	
Write Data Setup Time	t_{DSW}	90	-	-	ns	
Write Data Hold Time	t_{DHW}	90	-	-	ns	
A0 Setup Time	t_{ASS}	90	-	-	ns	
A0 Hold Time	t_{AHS}	90	-	-	ns	
Clock Low Time	t_{CLKL}	90	-	-	ns	
Clock High Time	t_{CLKH}	90	-	-	ns	
Rise Time [10% ~ 90%]	t_R	-	-	15	ns	
Fall Time [90% ~ 10%]	t_F	-	-	15	ns	

SPI DATA-OUT (read)

Item	Symbol	Min.	Typ.	Max.	Unit	Remark
Clock Cycle Time	t_{cycle}	600	-	-	ns	
Chip Select Setup Time	t_{CSS}	400	-	-	ns	
Chip Select Hold Time	t_{CSH}	150	-	-	ns	
Read access time	t_{ACCS}	-	-	200	ns	
A0 Setup Time	t_{ASS}	90	-	-	ns	
A0 Hold Time	t_{AHS}	90	-	-	ns	
Clock Low Time	t_{CLKL}	400	-	-	ns	
Clock High Time	t_{CLKH}	150	-	-	ns	
Rise Time [10% ~ 90%]	t_R	-	-	15	ns	
Fall Time [90% ~ 10%]	t_F	-	-	15	ns	
		VCC = 2.3 to 3.6V			**Temp = 0 to +50℃**	

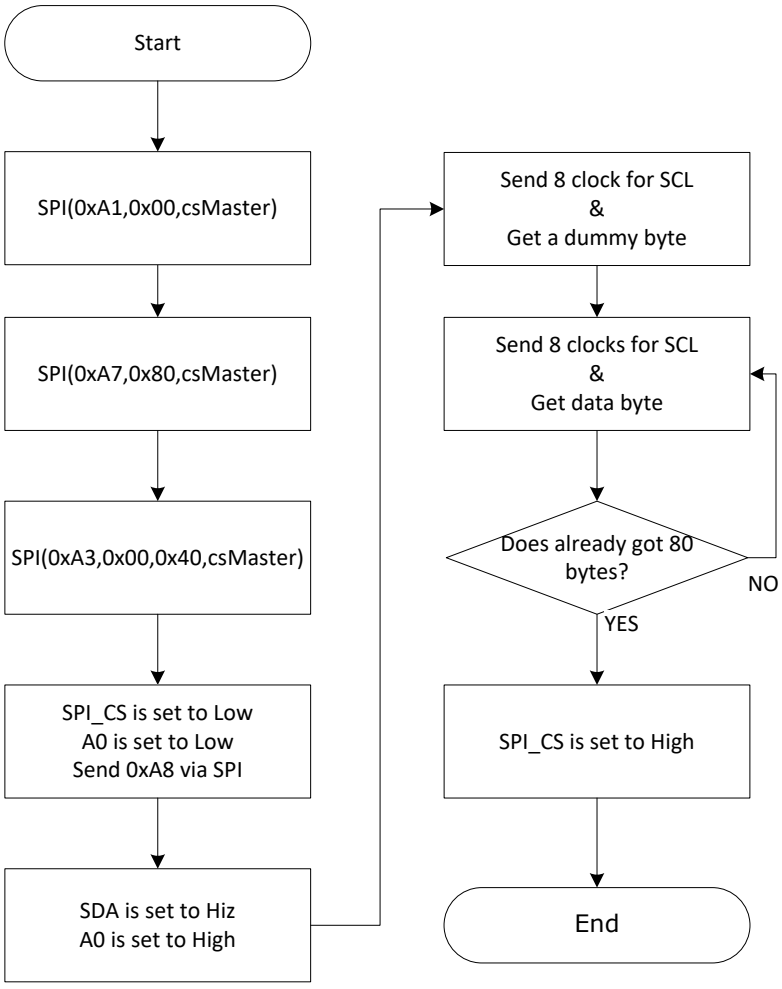
1.6 Read MTP data

The first 80-bytes section of MTP have saved the user-defined data that includes the information of the display and soft-start parameters. The section will introduce how to read out the data through the SPI.



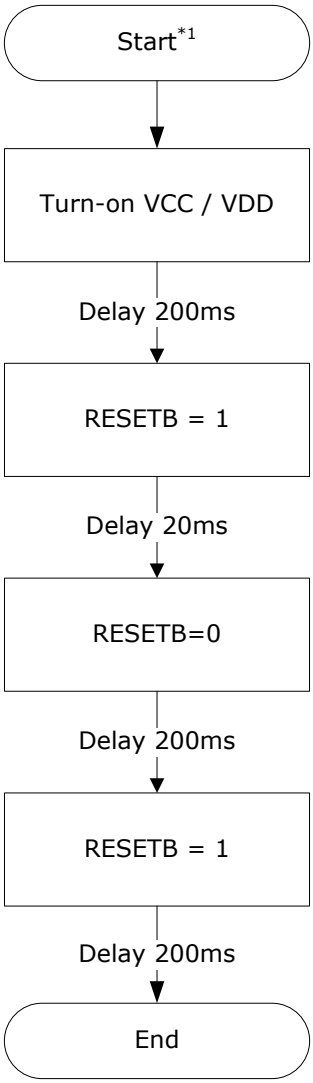
Note: 1. After read enable command is set, SDA must set Hiz, and A0 set high to active read operation
2. When read operation is done, CSB must set high once to quit read operation.

Read operation of 4-Line SPI



2 Power on COG driver

This flowchart describes power sequence for driver chip.

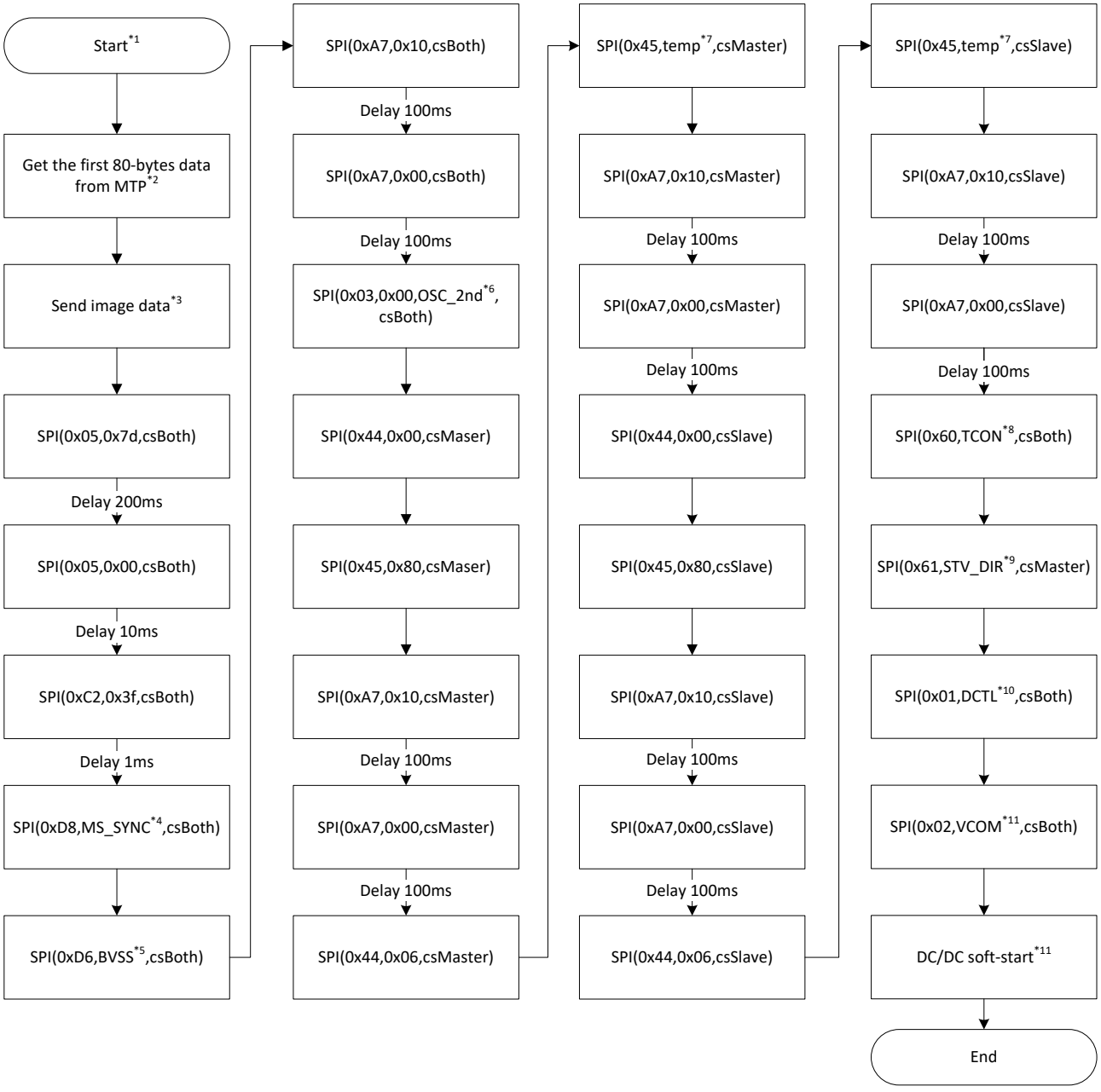


Note:

1. Start
   Initial State:
   VCC/VDD, RESETB, M_CSB, S_CSB, SDA,

3 Initialize COG Driver

3.1 Initial flow chart



Note:

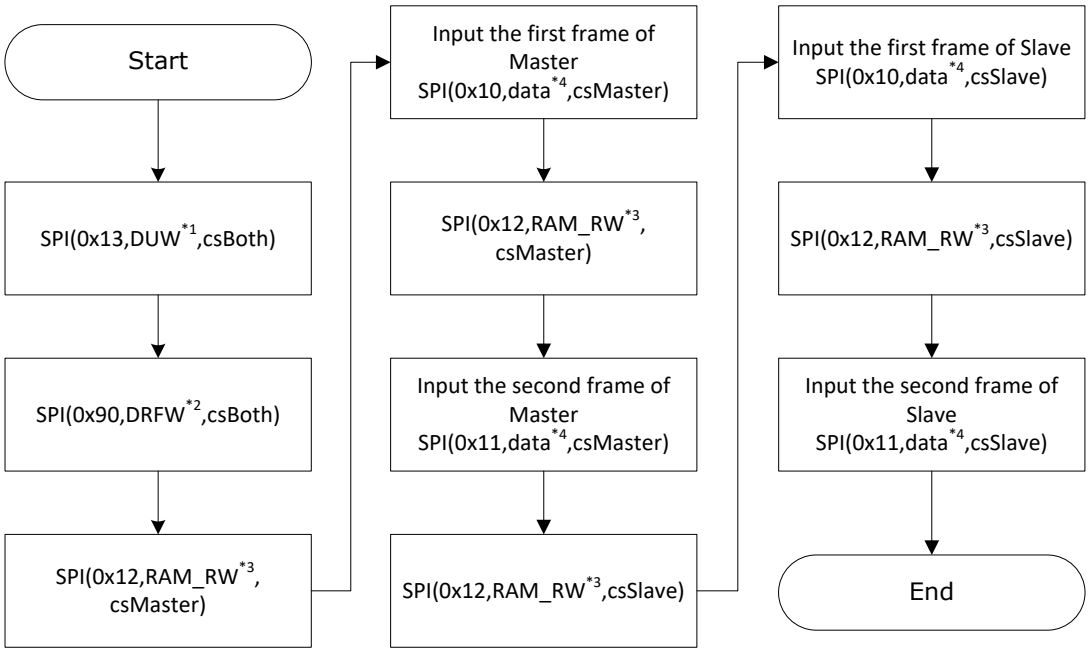
1, Start: Follow the end of the power on sequence
2. Please refer to section 1.6 to get the 80-bytes of the user-defined.
3. Please refer to section 3.2
4. MS_SYNC is read from 0x1D of MTP memory
5. BVSS is read from 0x1E of MTP memory
6. OSC is read from 0x12 of MTP memory
7. The data represents the temperature value. The acceptable range of temperature is -40 ~ 87°C and 0.5°C per step. Such as,

   -40°C = 0x00,
   0°C = 0x50,
   25°C = 0x82,
   87°C = 0xFE
8. TCON is read from 0x0B of MTP memory.

9. STV_DIR is read from 0x1C of MTP memory
10. DCTL is read from 0x10 of MTP memory
11. VCOM is read from 0x11 of MTP memory
12. Please refer to section 3.3

3.2 Send image to the EPD

This section describes how to send image data into COG which will be displayed on the display.



Note:
1, **DUW**: there is 6 bytes' data that are read from 0x16 ~ 0x1B of MTP memory.
2. **DRFW**: there is 4 bytes' data that are read from 0x0C ~ 0x0F of MTP memory.
3. **RAM_RW**: there is 3 bytes' data that are read from 0x13 ~ 0x15 of MTP memory.
4. The data of totally have 46,080 bytes, please refer to next page to send the data.

Both First and Second frame data need to be send into Master and Slave driver respectively each updating. The index of the First frame is **0x10** and the Second frame is **0x11.**

- Image format

The data of image frame, one bit represents 1 pixel. (e.g. the first byte represents the 1st~ 8thpixels of the first line, the second byte represents the 9th~ 16thpixels of the first line, …… and so on).

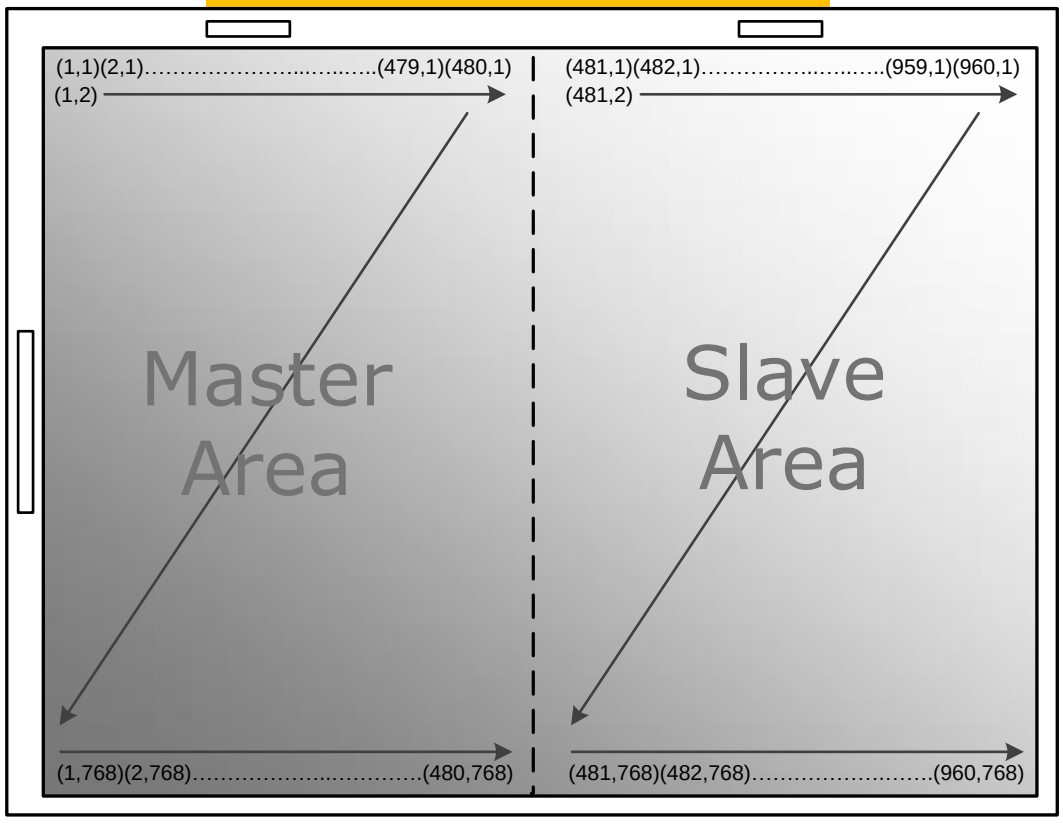
Master Image data input sequence :
Line1:(1,1)>(2,1)>…>(480,1)>
Line2:(1,2)>(2,2)>…>(480,2)>
:
:
Line768: ……>(480,768)

Total : 1 x 480 x 768
= 368,640 bits
= 46,080 Bytes

Slave Image data input sequence :
Line1:(481,1)>(482,1)>…>(960,1)>
Line2:(481,2)>(482,2)>…>(960,2)>
:
:
Line768: ……>(960,768)

Total : 1 x 480 x 768
= 368,640 bits
= 46,080 Bytes

Data Byte	D[7]	D[6]	D[5]	D[4]	D[3]	D[2]	D[1]	D[0]
Pixel	P[n]	P[n+1]	P[n+2]	P[n+3]	P[n+4]	P[n+5]	P[n+6]	P[n+7]



- First frame

  In this frame, the data "1" represents both black and red color pixel and the data "0" represents white color pixel.

Data	Pixel Color
1	Black
0	White/Red

- Second frame

  In this frame, the data "1" represents black color pixel and the data "0" represents both red and white color pixel.

Data	Pixel Color
1	Red
0	Black/White

3.3 DC/DC soft-start

There are 32-bytes data for describing the sequence of soft-start.

	0/8	1/9	2/10	3/11	4/12	5/13	6/14	7/15
…	…….							
0x20	1st stage							
0x28	2nd stage							
0x30	3rd stage							
0x38	4th stage							
…	……							

The sequence totally has 4 stages. Each stage has 8 byte parameters. The bytes of each stage can be interpreted in 2 ways.

Data structure and definition:

	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
format1	REPEAT/FORMAT	PHL_INI	PHH_INI	PHL_VAR	PHH_VAR	BST_SW_a	BST_SW_b	DELAY
format2	REPEAT/FORMAT	BST_SW_a	BST_SW_b	DELAY_a	DELAY_b	?	?	?

REPEAT/FORMAT:

The times to repeat and the data format used in this stage

The MSB defines the format used in this stage

bit	7	6	5	4	3	2	1	0
REPEAT/FORMAT	Format	Times to repeat						

Format: 1-> bytes are defined as “format1”(see above)

0-> bytes are defined as “format2”(see above)

Example: 0x87 -> format1, repeat 7 times

0x64 -> format2, repeat 100 times

PHL_INI:

Define the initial value of PHL(the first data of the reg.0x51)

PHH_INI:

Define the initial value of PHH(the second data of the reg.0x51)

PHL_VAR:

The byte represents the changing value of PHL with each iteration(REPEAT)

PHH_VAR:

The byte represents the changing value of PHH with each iteration(REPEAT)

Both PHL_VAR_n and PHH_VAR_n could be a negative number. The negative number is represented by 2's complement.

Example: -5 equals 0xFB

BST_SW_a:

BST_SW setting is the power on/off manager(reg.0x09) at the start of the phase.

BST_SW_b:

BST_SW setting is the power on/off manager(reg.0x09) at the end of the phase.

DELAY:

The delay time at the end of the stage.

bit	7	6	5	4	3	2	1	0
DELAY_n	Scale	Delay time						

Scale: 1 -> the scale of the delay time is msec.

0 -> the scale of the delay time is 10usec.

Example: 0x82 -> delay 2ms

0x02 -> delay 20us

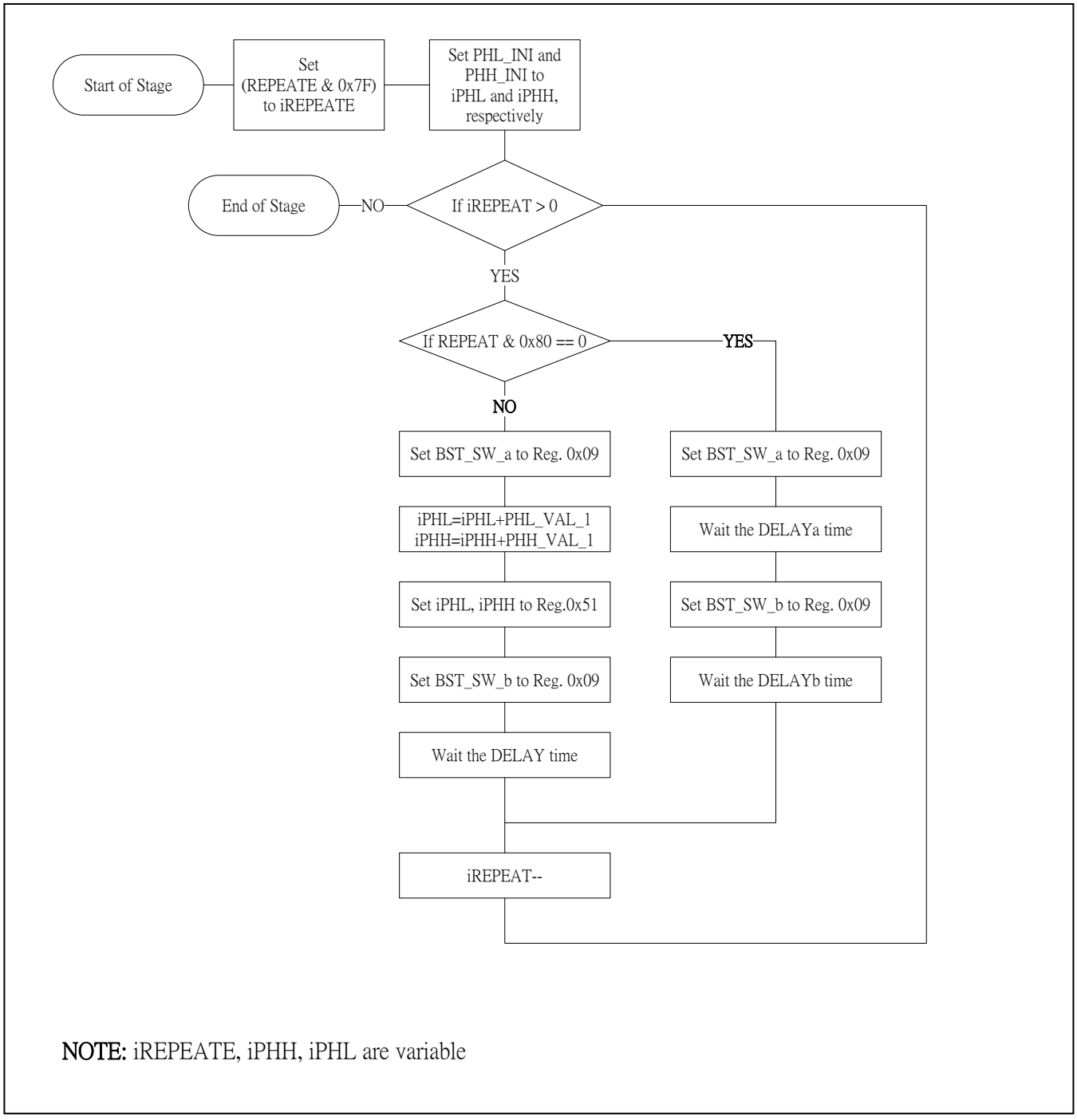
DELAY_a:

Same as "DELAY" but inserted after BST_SW_a

DELAY_b:

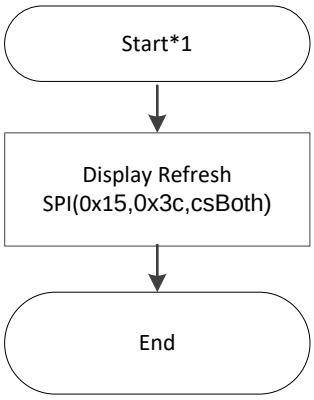
Same as "DELAY" but inserted after BST_SW_b

Following is the flowchart for each “stage”,



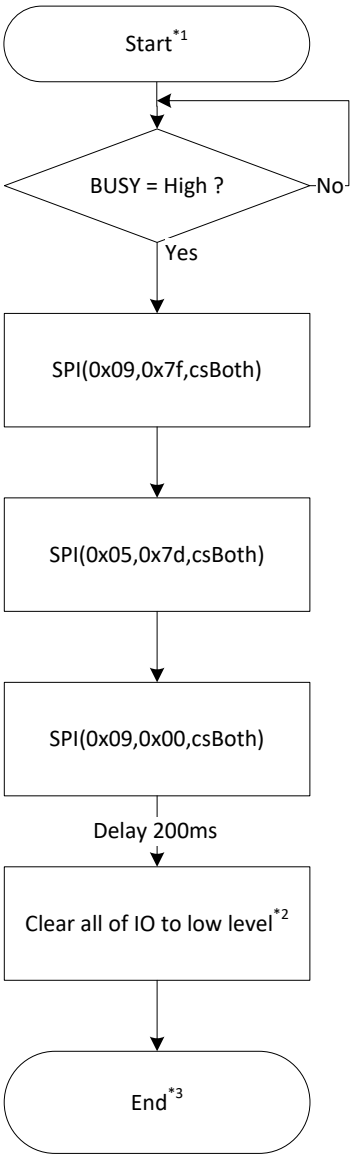
NOTE: iREPEATE, iPHH, iPHL are variable

4 Send updating command



Note:
1. Start
   Follow the end of the COG initial flow

5 Turn-off DC/DC



Note:

1. Start
   Follow the end of the send updating command
2. VCC/VDD, RESETB, A0, M_CSB, S_CSB, SCL and SDA
3. Finished the all of the steps for update the 12" EPD

Copyright

龍亭新技股份有限公司 Pervasive Displays Inc.

4F, No. 28, Chuangye Rd., Tainan Science Park, Tainan City 74144, Taiwan (R.O.C.)

Tel: +886-6-279-5399

http://www.pervasivedisplays.com

Revision History

Version	Date	Page (New)	Section	Description
01	2017/11/6			First official edition

Glossary of Acronyms

EPD	Electrophoretic Display (e-Paper Display)
EPD Panel	EPD
TCon	Timing Controller
FPL	Front Plane Laminate (e-Paper Film)
SPI	Serial Peripheral Interface
COG	Chip on Glass
PDI, PDi	Pervasive Displays Incorporated