****

********** Communication protocol between PC and Data Terminal **********

****

The data communication between PC and data terminal is implemented by the request command and response packet. The master device send the prefix data packet to the slave device to request a service and slave device will return a response packet to the master to confirm the request. The overall communication sequence are described as following,

1. Prefix Packet

■ Prefix Packet

Field Name	Header 1	Header 2	Master ID	Slave ID	Extension	Parameter
Numbers of Byte	1bh	02h	XXh	XXh	XXh	XXh

Length	END 1	END 2	Checksum
Size of Data Buffer in Data Packet (4 Byte)	02h	1bh	Sum of Header1 ~ END2 in Prefix Packet (2 Byte)

14 bytes

♦ Master ID: (Send Packet)

Value	Description
0Ah	PC Side
0Bh~FEh	PDT1 ~ PDT244

♦ Slave ID: (Receive Packet Client)

Value	Description
0Ah	PC Side
0Bh~FEh	PDT1 ~ PDT244

♦ Extension:

Value	Description
00h	Normal
01h~FFh (Cradle Version)	If “parameter” equals 0Ah, this value is for Cradle version. **Note: Value / 10 = Cradle Version**

♦ Parameter:

Value	Description	Remark
01h	Connection Request	
03h	File Transfer	
04h	Terminal Function Call	
05h	Product Test	
07h	Terminal is busy(※No Response)	
08h	File Transfer with Full Path	ArgoLink 2.10
0Ah	Return Cradle Version.	

2. Data Packet

■ Data Packet

Field Name	Header	Master ID	Slave ID	File ID	Block ID
Numbers of Byte	02h	XXh	XXh	00h	XXXXh (2 Bytes)

Data Buffer	END	Checksum
Size of data buffer is according to the “Length” field of prefix packet.(*1)	02h	Sum of Header ~ END in Data Packet (2 Byte)

♦ Master ID:

Value	Description
0Ah	PC
0Bh~FEh	PDT1 ~ PDT244

♦ Slave ID:

Value	Description
0Ah	PC
0Bh~FEh	PDT1 ~ PDT244

♦ Block ID:

Value	Description
00h~FFFFh	The number of this Data Packet.

♦ Data Buffer:

Value	Description
Any value	Read the next page. **Max. length equal 512 bytes**

2.1. Amount of Data Buffer for File Transfer

2.1.1. Data Buffer structure for file transfer

Field Name	Load Type	File Type	File Name	File Length	Date	Time
Numbers of Byte	XXh	XXh	XXXXXXXX.OOO (12 Bytes)	XXXXXXXXh	XXXXh	XXXXh

Create Flag	Reserve	File Data
XXh	9 Bytes	According to the file length

♦ Load Type:

Value	Description
01h	Upload (PC->PDT)
02h	Download (PDT--> PC)

♦ File Type:

Value	Description
00h	Project
01h	Lookup file
02h	font file
04h	BIOS(*2)
05h	Data File(Download)

♦ File Name:

Value	Description
Any value	A string for file name. For example, it will be “ABC.dat”.

♦ File Length:

Value	Description
00h~FFFFFFFFh	The file Total length.

♦ Date:

Value	Description
00h~FFFFh	2 bytes

	Year – 1980							Month				Date				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

♦ Time:

Value	Description
00h~FFFFh	2 bytes

	Hour					Minute						Second/2				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

♦ Create Flag:

Value	Description
00h	If file exist no write.
01h	If file exist overwrite.

♦ File Data:

Value	Description
Any value.	File Data, Max length is 512 bytes.

2.2. Amount of Data Buffer for Terminal Function Call

2.2.1. Data Buffer structure for Terminal Function Call

Field Name	Parameter	Data
Numbers of Byte	XXh	According to the type of Parameter

♦ Parameter:

Value	Description	Data Type
00h	Dir direct	directory + 0x0d + 0x0a
		directory: PDT, C:\, C:\DATA, D:\, D:\FONTS, D:\LOOKUP or D:\PROGRAM Ex: C:\+ 0x0d + 0x0a
01h	Delete File.	File + 0x0d + 0x0a Ex: D:\fonts\font16.cft + 0x0d + 0x0a
03h	Copy File from Terminal to PC.	File + 0x0d + 0x0a Ex: D:\fonts\font16.cft + 0x0d + 0x0a
04h	Format Disk.	Disk + 0x0d + 0x0a Disk: C:\ or D:\ Ex: D:\ + 0x0d + 0x0a
05h	Ask Disk Information.	0x0d + 0x0a
10h	Get ROM ID (*3).	0x0d + 0x0a
11h	Start to upgrade new firmware.	0x11 + 0x0d + 0x0a
12h	Upgrading.	0x12 + 0x0d + 0x0a
13h	Get cradle version.	0x13 + 0x0d + 0x0a
14h	Set Date/Time	Date + 0x0d + 0x0a Ex: 20060727153025 + 0x0d + 0x0a Date/Time Format： YYYYMMDDhhmmss YYYY => Year MM => Month DD =>Day hh => Hours mm =>Minutes ss => Second
64h	Delete File FALSE	0x64
65h	Delete File OK	0x65

2.3. Amount of Data Buffer for Terminal Function Call Response

2.3.1. Data Buffer structure for Terminal Function Call Response in Dir Structure(Per file, folder or disc)

Field Name	File name	File Type	File Time	File Date	Sector	File Size	Divide Char 1	Divide Char 2
Numbers of Byte	11Bytes	XXh	2Bytes	2Bytes	2Bytes	4Bytes	0dh	0ah

♦ Example:

File1	Divide Char 1	Divide Char 2	File2	Divide Char 1	Divide Char 2	File3	Divide Char 1	Divide Char 2

♦ File Name:

Value	Description
Any value	A string for file name. For example, if a file name is 12345678.dat, it will be “12345678dat”.

♦ File Type:

Value	Description
00h	Disc
10h	Folder
20h	File

♦ File Time:

Value	Description
00h~FFFFh	2 bytes(F/W 2.00 or new)

	Hour					Minute						Second/2				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

♦ File Date:

Value	Description
00h~FFFFh	2 bytes(F/W 2.00 or new)

	Year – 1980							Month				Date				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

♦ Sector:

Value	Description
00h~FFFFh	The sector of the file.

♦ File size:

Value	Description
00h~FFFFFFFFh	The file total size.

2.3.2. Data Buffer structure for Terminal Function Call Response in Delete File

Field Name	Data
Numbers of Byte	XXh

♦ **Data:**

Value	Description
64h	False.
65h	OK.

2.3.3. Data Buffer structure for Terminal Function Call Response in Format Disc

Field Name	Data
Numbers of Byte	XXh

♦ **Data:**

Value	Description
64h	False.
65h	OK.

2.3.4. Data Buffer structure for Terminal Function Call Response in Ask Disk Information

Field Name	Response	Firmware Ver.	Disk C Size	Disk C Free Size	Disk D Size	Disk D Free Size
Numbers of Byte	XXh	4Bytes	4Bytes	4Bytes	4Bytes	4Bytes

♦ Response:

Value	Description
64h	False.
65h	OK.

♦ Firmware Version:

Value	Description
00h~FFFFFFFFh	Return F/W Ver.

♦ Disk C Size:

Value	Description
00h~FFFFFFFFh	Return C Disk size.

♦ Disk C Free Size:

Value	Description
00h~FFFFFFFFh	Return C Disk free size.

♦ Disk D Size:

Value	Description
00h~FFFFFFFFh	Return D Disk size.

♦ Disk D Free Size:

Value	Description
00h~FFFFFFFFh	Return D Disk free size.

2.3.5. Data Buffer structure for Terminal Function Call Response in Get ROM ID

Field Name	Response	Main ID	Device ID
Numbers of Byte	XXh	4Bytes	4Bytes

♦ Response:

Value	Description
64h	False.
65h	OK.

♦ Main ID(*3):

Value	Description
00h~FFFFFFFFh	Return Main ID.

♦ Device ID(*3):

Value	Description
00h~FFFFFFFFh	Return Device ID.

2.4. Amount of Data Buffer for File transfer with Full Path

2.4.1. Data Buffer structure for file transfer in full path

Field Name	Load Type	File Length	Date	Time	Create Flag	Reserve	Name Length	File Name
Numbers of Byte	1 Byte	4 Bytes	2 Bytes	2 Bytes	1 Byte	20Bytes	2 Bytes	According to the Name Length

♦ Load Type:

Value	Description
01h	Upload (PC->PDT)
02h	Download (PDT--> PC)

♦ File Length:

Value	Description
00h~FFFFFFFFh	For return file total length.

♦ File Date:

Value	Description
00h~FFFFh	2 bytes(F/W 2.00 or new)

	Year – 1980							Month				Date				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

♦ File Time:

Value	Description
00h~FFFFh	2 bytes(F/W 2.00 or new)

	Hour					Minute						Second/2				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

♦ File Length:

Value	Description
00h	If file exist no write.
01h	If file exist overwrite

♦ Name Length:

Value	Description
00h~FFFFh	File name length.

3. Response Packet

■ Response Packet

Field Name	Header	Master ID	Slave ID	File ID	Block ID	Response
Numbers of Byte	02h	XXh	XXh	00h	XXXXh (2 Bytes)	XXh

END 1	END 2	END 3	Checksum
0dh	0ah	02h	Sum of Header ~ END3 in Data Packet (2 Byte)

♦ Master ID:

Value	Description
0Ah	PC Side
0Bh~FEh	PDT1 ~ PDT244

♦ Slave ID:

Value	Description
0Ah	PC Side
0Bh~FEh	PDT1 ~ PDT244

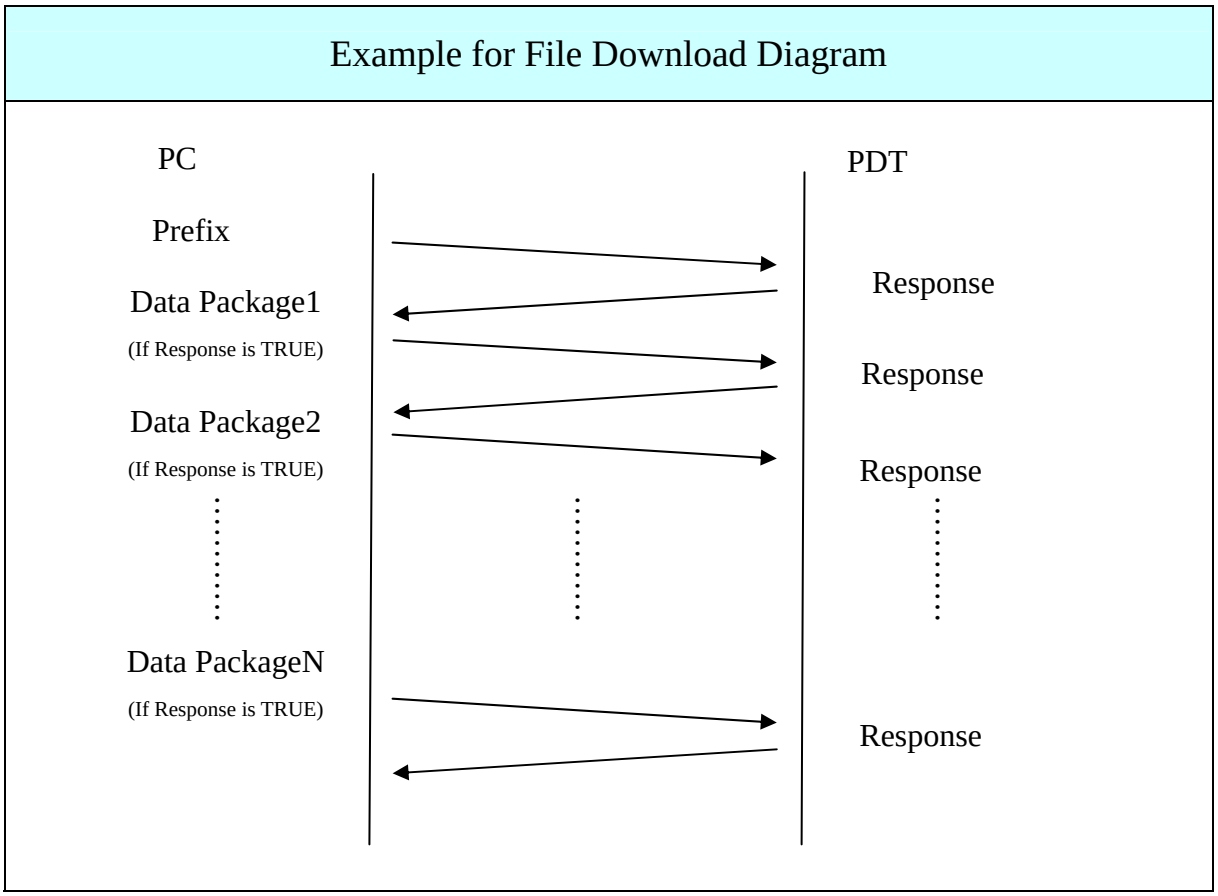
♦ Block ID:

Value	Description
00h~FFFFh	The number of this Data Packet.

♦ Response ID:

Value	Description
00h	Fail
01h	OK
03h	Download ROM Write File(ROMWrite.abp) ※ Block ID is replace by Main ID ※ 0x0d 0x0a is replace by Device ID
04h	Disk is unformat.
05h	Disk space is not enough.
06h	File has already existed.
07h	Terminal is busy. (※No Response)
08h	File Open Fail in Terminal
09h	Power lower
0Ah	Folder not exists.
0Bh	Disk D is needed to rearrange by manual
0Ch	File in using(R/W) can’t be deleted.
0Dh	File in using(R/W) can’t be replaced.
0Eh	There are files that are used in the disk, it is unable to format

4. Example of Communication between PC and Terminal



Data Package 2 ~ N Data Buffer

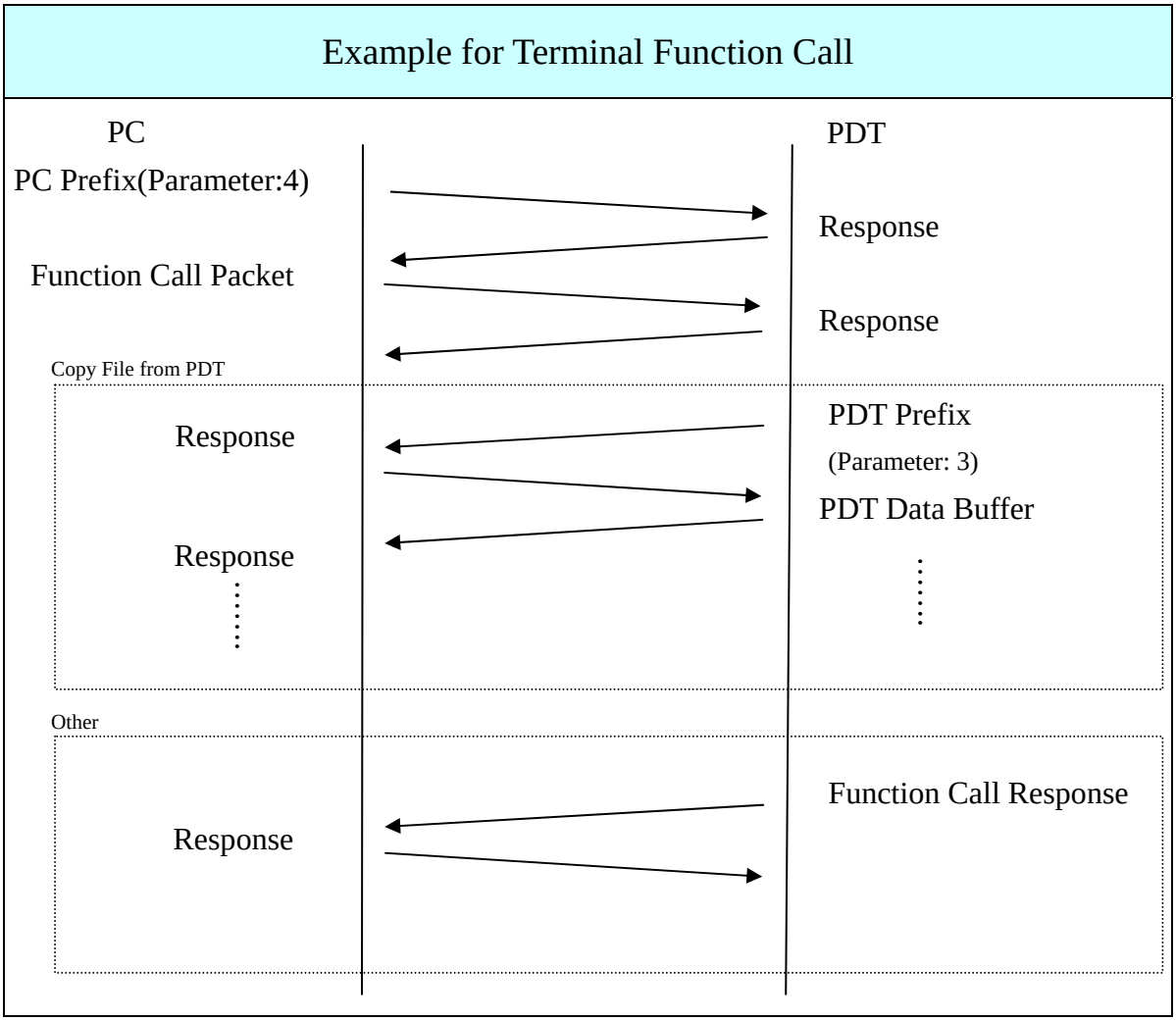
<02h><00h><00h><00h><00h><00h><File Data>……<02h><70h><F3h>

Check Sum: F370h

※Copy File From PC to PDT(Full Path, copy to c:\Data)

Prefix Data
<1Bh><02h><00h><00h><00h><08h><C7h>
<02h><00h><00h><02h><1Bh><0Eh><01h>
Parameter: 0x08
Length: 000002C7h
Checksum: 010Eh

Data Package1 Data
<02h><00h><00h><00h><00h><00h><01h><94h>
<02h><00h><00h><09h><35h><51h><5Fh><00h>
<00h><00h><00h><00h><00h><00h><00h><00h>
<00h><00h><00h><00h><00h><00h><00h><00h>
<00h><00h><00h><00h><13h><00h><63h><3Ah>
<5Ch><64h><61h><74h><61h><5Ch><43h><68h>
<65h><63h><6Bh><53h><4Eh><2Eh><64h><61h>
<74h> <File Data>…….
<02h><1Bh><5Dh>
Load Type: 1
File Length: 0294h
File Date: 3509h
File Time: 5F51h
File Length: 0294h (660 byte)
Create Flag: 0
File Name Length: 0013h
File Name: c:\data\CheckSN.dat
Check sum: 5D1Bh



Copy File from PDT to PC

PC Data Packet Load Type: 03 Data Buffer c:\data\CheckSN.dat\r\n

PDT Prefix Parameter: 3

PDT Data Buffer Load Type: 2 File Type: 5 File Name: CheckSN.dat□ File Length: 660(byte) File Date: 3509h File Time: 5F51h Flag: 0 File Data

5. Remark:

♦ **Checksum**

2 bytes accumulation

♦ **Data Buffer maximum is 512 bytes, if the transmitted data is over 512, it must be divided to several time to transmit**

♦ **(*1)If the Parameter of Terminal function Call is '\x00'(Dir)**
The Parameter of Prefix is '\x05' (Terminal Function Call Response)
And Data Buffer is Command Response.
If the Parameter of Terminal function Call is '\x03' (Copy File)
The Parameter of Prefix is '\x03'
And Data Buffer is File Transfer.

♦ **(*2)In BIOS File Transmission**
File Length must add 6 bytes and File content must add File content Length (4 bytes) + Bios File Content + checksum (2 bytes)

♦ **(*3)Table of Romwrite File**

Main ID	Device ID	File
0x00BF	0x2781	SST_RW.abp
	0x2789	

♦ **If old version don't support, show message to warning user.**

Protocol Ver.	Terminal Ver.	Software Ver.	
Version 1.1	V 2.10	V 2.10	1.Add transfer with full path

♦ **Master(Slave) ID:**

0x0A: PC
0x0B: PDT1
.
.
.
0xFE: PDT244
0xFF: ALL PDT
(0x00 ~ 0x09 is present for Firmware)