App Communication Protocol 3.3

App Connection Algorithm

Pill box will advertise in following two scenarios.

1->General Notification -> In this case PB will advertise every 5 minutes (ping interval) to transfer the parameters like humidity, temperature, any alarm etc. to mobile app.

2-> On pressing the panic button for 3 seconds will trigger PB to advertise every 5 seconds.

Using advertisement from PB app can initiate connection and get connected with PB.

Configuration Packet

HEADER, BL\_NAME, PW, PING INTERVAL, RTC, XXX, LP-TIMER1, LP-TIMER2, LP-TIMER3.... LP-TIMER(XXX), ALARM-VOLUME, TIME-MODE, ALARM-GAP, ALARMDURATION, PANICINTERVAL, BLE CONNECTION TIMEOUT, BACKLIGHT TIMEOUT, DISPLAY TIMEOUT, BATTRYE ALERT VALUE, FR, \*CHECKSUM, &

|  |  |  |
| --- | --- | --- |
| **SL.NO CONTENT** | **SIZE** | **DESCRIPTION** |
| **1.**HEADER | 3(F) | HEADER OF PACKET |
| **2.** BL\_NAME | 15 | TO SET BLUETOOTH NAME |
| **3.** PW | 6(F) | PASSCODE FOR BLUETOOTH |
| **4.** PING INTERVAL 4(F) | | PING INTERVAL OF DEVICE |
| **5.** RTC | 10(F) | SET RTC EPOCH OF DEVICE |
| **6.** XXX | 3(F) | This field tells total number of timers configured, XXX will be replaced by timer count i.e. 001 or 005 or 007 or 009 or 029 or 032 |
| **7.** LP-TIMER n (n =0..120) | 13(F) EACH | SET TIMERS EPOCH FOR ALARMs, LP is led position out of 29 leds. LP (2 bytes) = 00-28. See the Note. |
| **8.** ALARM VOLUME | 1(F) | 0-4 WHERE 0 IS MUTE,4 IS MAXIMUM WITH 25% STEPS. |
| **9.** TIME-MODE | 1(F) | TIME MODE (24 HOUR/12HOUR) |
| **10.** ALARM GAP | 4(F) | GAP BETWEEN TWO ALARM ALERTS |
| **11.** ALARM DURATION | 4(F) | TOTAL ALARM ALERT DURATION |
| **12.** PANIC INTERVAL | 4(F) | DURATION FOR WHICH PANIC STATE SHOULD GIVE ALERT |
| 13. BLE CONNECTION INTERVAL | 4(F) | DEVICE DISCONNECTION TIMEOUT |
| 14. BACKLIGHT\_TIMEOUT | 4(F) | BACKLIGHT ON OFF TIMEOUT |
| 15. DISPLAY TIMEOUT | 4(F) | DISPLAY ON TIME OUT |
| 16. BATTERY\_ALERT\_VALUE | 2(F) | BATTERY ALERT VALUE IN % |
| **17.** FR | 1(F) | FACTORY RESET |
| **18.** \*CHECKSUM | 3(F) | PACKET CHECKSUM |
| **19.** & | 1 | END CHARACTER |

**NOTE:**

1. Setthe field as “-1” which you don’t want to edit.
2. Set the field as “-2” if you want to disable any Timer.
3. Password cannot be less than “0”
4. Timers are 121 but positions are depending on their Model.

Like: PB28 has 28(00-27) compartments, PB29 has 29(00-28) and PB07 has 7(00-06) Compartment. So, some positions can be duplicate in multiple timers.

1. Don’t display HEADER, FR and CHECKSUM fields on editable form.
2. Bluetooth name can’t contain ‘&’.
3. PING INTERVAL, ALARMGAP, ALARM DURATION, PANIC INTERVAL can be in seconds, minutes, hours e.g. 120S or 120M or 120H. Give option to user for selecting interval value (3 DIGIT )
4. If the Ble Interval, display timeout and Backlight timeout is set ‘0’ then device neither disconnect from the app nor turn off the Backlight.
5. Device will choose the maximum timeout for the sleep from backlight, BLE connection and display Timeout.
6. If anyone from the Backlight, BLE connection and Display Timeout is configured “0”, then device will not go into the sleep.

**Sample Configuration Packet:**

CFG, DEVICE\_TEST,661900,VERSION, 300S, 2905142532,032,01-0205182632, 02-0305082532, 03-0405082532 , 04-0505082532 , 06-0523082532 , 07-0523082532 , 08-0523082532 , 09-0523082532 , 10-0523082532 , 11-0523082532 , 12-0523082532 , 13-0523082532 , 14-0523082532 , 15-0523082532 , 16-0523082532 , 17-0523082532 , 18-0523082532 , 19-0523082532 , 20-0523082532 , 21-0523082532 , 22-0523082532 , 23-0523082532 , 24-0523082532 , 25-0523082532 , 26-0523082532,27-0523082532,00-0523082532,05-0523082532,28-0523083532,28-0523083630,28-0523083732,28-0523083932,0,0,005M,030M,010M,120S,060S,040S,20,0, \*43,&

|  |  |  |
| --- | --- | --- |
| **S.NO CONTENT** | **SIZE** | **DESCRIPTION** |
| **1.** CFG | 3(F) | HEADER OF PACKET |
| **2.** DEVICE\_TEST | 15 | BLUETOOTH NAME |
| **3.** 661900 | 6 | PASSCODE FOR BLUETOOTH |
| **4.** VERSION | 15 | Device version |
| **5.** 300S | 4 | PING INTERVAL OF DEVICE |
| **6.**  2905142532 | 10(F) | SET RTC OF DEVICE |
| **7.**032 | 3(F) | It indicates 32 timers are being configured in this command |
| **8.**02-1702318869...  28-1702338869 | 13(F) EACH-ALARM | 1. means third led and 1702318869 Is epoch. These are 32 timers with 29 ids. |
| **9.** 0 | 1(F) | ALARM VOLUME |
| 10. 0 | 1(F) | TIME MODE (0 FOR 24 HOUR MODE, 1 FOR AM/PM MODE) |
| **11.** 300S | 4 (F) | ALARM GAP |
| **12.** 030M | 4(F) | ALARM DURATION |
| **13.** 010M | 4(F) | PANIC INTERVAL |
| **14. 120S** | **4(F)** | **BLE CONNECTION TIMEOUT** |
| 15. 060S | 4(F) | BACKLIGHT TIMEOUT |
| 16. 040S | 4(F) | DISPLAY ON TIMEOUT |
| 17. 20 | 2(F) | BATTERY ALERT |
| **18.** 0 | 1(F) | FACTORY RESET |
| **19.\***43 | 2(F) | PACKET CHECKSUM is 43 |
| **20.** & | 1 | END CHARACTER |

WORKING: If device is connected then upon pressing device image on app, app should query the configuration packet and display all received fields in editable form. This form should have saved button, which when pressed would send configuration packet to device. If device gives acknowledgement, then app should display “configuration saved” else app should reattempt one more time after which it will display “configuration failed”.

**QUERY COMMAND 🡺**

*CFG::?*

**REPLY=>**

HEADER, BL\_NAME, PW, VERSION, PING INTERVAL, RTC, XXX, LP-TIMER1, LP-TIMER2, LP-TIMER3....LP-TIMER(XXX), ALARM-VOLUME, TIME-MODE, ALARM GAP, ALARM DURATION, PANIC INTERVAL, BLE CONNECTION TIMEOUT, BACKLIGHT TIMEOUT, DISPLAY TIMEOUT, BATTRYE ALERT VALUE, FR, \*CHECKSUM, &

Working: User should get one button in app to read configuration from device. On receiving this command, device will respond with current configuration of device.

Protocol will be exactly as that of configuration saving protocol as explained above.

**Note: If any packet’s size increases beyond 480 bytes then it should be broken into multiple 480 byte packets as follows**:

$$0,480(bytes)

$$1,480(bytes)

$$2,480(bytes)

.

.

.

.

$$N,480(bytes), %%

When %% is received then we will process the received 480\*(N+1) bytes of data received.

\*At present N can be 0-3.

\*Last packet can be less than 480 bytes, depending on the available data.

Example:

**$$0,***CFG,DEVICE\_TEST,661900,3.31,300S,2905142532,062,01-0205182632,02-0305082532,03-0405082532,04-0505082532,06-0523082532,07-0523082532,08-0523082532,09-0523082532,10-0523082532,11-0523082532,12-0523082532,13-0523082532,14-0523082532,15-0523082532,16-0523082532,17-0523082532,18-0523082532,19-0523082532,20-0523082532,21-0523082532,22-0523082532,23-0523082532,24-0523082532,25-0523082532,26-0523082532,27-0523082532,00-0523082532,05-0523082532,28-0523083532,28-0523083630,28-05230837*

**$$1,***32,20-0523083942,21-0523083832,22-0523084932,19-0523082932,28-0523083933,25-0523086932,18-0523073932,12-0523083232,11-0523083982,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,0,0,005M,030M,010M,120S,060S,040S,20,0,\*43,&***,%%**

On receiving above two packets PB will club both packets and process as follows:

*CFG,DEVICE\_TEST,661900,3.31,300S,2905142532,062,01-0205182632,02-0305082532,03-0405082532,04-0505082532,06-0523082532,07-0523082532,08-0523082532,09-0523082532,10-0523082532,11-0523082532,12-0523082532,13-0523082532,14-0523082532,15-0523082532,16-0523082532,17-0523082532,18-0523082532,19-0523082532,20-0523082532,21-0523082532,22-0523082532,23-0523082532,24-0523082532,25-0523082532,26-0523082532,27-0523082532,00-0523082532,05-0523082532,28-0523083532,28-0523083630,28-0523083732,20-0523083942,21-0523083832,22-0523084932,19-0523082932,28-0523083933,25-0523086932,18-0523073932,12-0523083232,11-0523083982,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,28-0523083932,0,0,005M,030M,010M,120S,060S,040S,20,0,\*43,&*

Sample Preset Configuration Packet:

HEADER,TEMPERATURE\_UNIT,PROLONG\_TH,WARNINH\_TH, NO\_OF\_TIME\_VOICEPLAY,&

Example:

**PRECFG,0,002M,015S,5,&**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO CONTENT** |  | **SIZE** |  | **DESCRIPTION** |
| **1.**  PRECFG |  | 6(F) |  | HEADER OF PACKET |
| **2.**  0 |  | 1 |  | TEMPERATURE\_UNIT |
| **3. 002M** |  | 4 |  | PROLONG\_THRESHOLD |
| **4. 015S** |  | 4 |  | WARNING\_THRESHOLD |
| **5.**  1 |  | 1 |  | NO\_OF\_TIME\_VOICEPLAY |
| **6. &** |  | 1 |  | FOOTER OF PACKET |

**QUERY COMMAND** 🡺 PRESET CONFIGURATION CHECK

**PRECFG::?**

**RESPONSE**🡺PRECFG,TEMPERATURE\_UNIT,PROLONG\_THRESHOLD,WARNING\_THRESHOLD,NO\_OF\_TIME\_VOICEPLAY,&

**TEMPERATURE UNIT**

1. 🡺 Degree Celsius
2. 🡺 Fahrenheit

QUERY COMMAND

QUERY COMMAND 🡺 Model Information

**MODEL::?**

**RESPONSE** 🡺

PB29 // For the Pill-Box 29

PB28 // For the Pill-Box 28

PB07 // For the Pill-Box 7

**QUERY COMMAND** 🡺 Factory Erase

**FR**

**RESPONSE** 🡺

Flash Erase Complete

**QUERY COMMAND** 🡺 PCBA Polarity

**PCBA\_POLARITY**

**RESPONSE** 🡺

Device configured for PCBA Polarity

**QUERY COMMAND** 🡺 CABINET Polarity

**CAB\_POLARITY**

**RESPONSE** 🡺

Device configured for CABINET Polarity

REPORT FETCHING PROTOCOL

**QUERY COMMAND=>**

PB\_REPORT\_ERASE

**REPLY=>**

ERASED SUCCESSFULLY

**QUERY COMMAND=>**

PB\_REPORT\_FETCH

**REPLY=>**

T000,BoxId,alarm\_status,epoch,epoch\_result,alarm\_repeatition\_cnt

T001,BoxId ,alarm\_status,epoch,epoch\_result,alarm\_repeatition\_cnt

.

.

.

T028,BoxId,alarm\_status,epoch,epoch\_result,alarm\_repeatition\_cnt

T028,BoxId,alarm\_status,epoch,epoch\_result,alarm\_repeatition\_cnt

T118,BoxId,alarm\_status,epoch,epoch\_result,alarm\_repeatition\_cnt

T120,BoxId,alarm\_status,epoch,epoch\_result,alarm\_repeatition\_cnt

$BoxId00,Wrong\_cnt,epoch1,epoch2,epoch3,...,epoch10,&

$BoxId01,Wrong\_cnt,epoch1,epoch2,epoch3,...,epoch10,&

$BoxId02,Wrong\_cnt,epoch1,epoch2,epoch3,...,epoch10,&

.

.

.

$BoxId28,Wrong\_cnt,epoch1,epoch2,epoch3,...,epoch10,&

$P00, prolonged\_cnt, epoch1, epoch2, epoch3, ..., epoch10, &

$P01, prolonged\_cnt, epoch1, epoch2, epoch3, ..., epoch10, &

$P02, prolonged\_cnt, epoch1, epoch2, epoch3, ..., epoch10, &

.

.

.

$P28, prolonged\_cnt, epoch1, epoch2, epoch3, ..., epoch10, &

**NOTE:**

* In this report we maintain two types of data for all epoch timer, first one is resulting time for pill consumption in which we store the pill consumption status, time and repetition count and separated by ‘,’.
* The second type data, we store the wrong activity (untimely/RANDOM opening) done on this compartment and separated by ‘$’.
* The Third type data, we store the prolonged activity(Lid open more than 120 sec) done on this compartment and separated by ‘$’.
* If the application wishes to halt the reception of reports while the device is in the process of sending data, it will transmit an 'Error' signal to the device.

**EXAMPLE:**

**COMMAND:**

PB\_REPORT\_FETCH

**REPLY:**

T000,00,1,0523082532,0523083520,3

T001,01,0,0523092532,0523093520,2

T002,05,2,0623092532,0623093520,2.

.

.

T028,06,3,0523052532,0523053520,1

T028,09,3,0523053532,0523054542,1

T028,10,3,0523053630,0523054632,1

T028,28,3,0523053730,0523054752,1

**Description**:

T001,05,1, 0523082532, 0523083520,3

|  |  |  |
| --- | --- | --- |
| **S.NO CONTENT** | **SIZE** | **DESCRIPTION** |
| **1.** T001 | 4 | HEADER OF PACKET,001 AFTER HEADER IS TIMER ID, IT CAN BE (0-120) |
| **2.** 05 | 2(F) | BoxId(00-28) |
| **3.**  1 | 1(F) | ALRAM STATUS   1. NOT CONFIGURED 2. PASS 3. FAIL   3ABNORMAL PASS |
| **4.** 0523082532 | 10(F) | CONFIGURED TIME FOR ALARM |
| **5.** 0523083520 | 10(F) | PILL TAKING TIME |
| **6.** 3 | 1(F) | PILL WAS TAKEN AFTER 3 ALERTS |

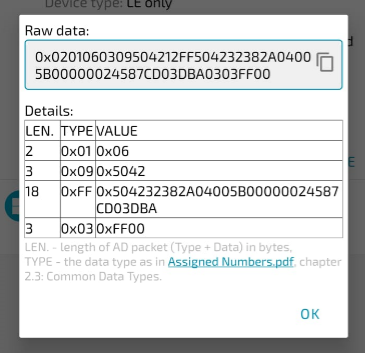
Description:

$01,15, 0523082532, 0523082932, 0523085597,..., 0523083598,&

|  |  |  |
| --- | --- | --- |
| **S.NO CONTENT** | **SIZE** | **DESCRIPTION** |
| **1.** $01 | 3 | HEADER OF PACKET,01 AFTER HEADER IS BOX ID, IT CAN BE (0-28) |
| **2.**15 | 3 | Wrong Opening Event Count |
| **3.** 0523082532 | 10(F) | Epoch of wrong event, there can be 10 such epoch for a single boxid. |
| **4** & | 1(F) | Termination |

BLE ADVERTISING PROTOCOL

We will share some device information in the advertising packet. We put the Device information (Model Info, Temperature, Humidity, Battery-Percentage, Alert-Type, Alert-info1, Alert-info2, BLE MAC Address) on the Manufacturing data. Example: 0x504232382A040005050605F412FAA093F6



Model Info = First Byte Second Byte Third Byte Fourth Byte

Temperature =fifthByte.SixthByte

Humidity = SeventhByte

Battery% = Eighth Byte

Alert Type = Ninth Byte

Alert-Info1 = Tenth Byte

Alert-Info2 = Eleventh Byte

BLE MAC = Twelfth to Seventeenth Byte

**Model Info**

1. PB28 For 28 Compartment
2. PB29 For 29 Compartment
3. PB07 For 7 Compartment

NOTE:

If you want to fetch advertisement packet then you will send**“Fetch\_Ble\_Adv?”** Command**.**

It is required to be fetch advertisement regularly while Device is connected with application.

Alert Type

1. NONE
2. PANIC
3. LOW BATTERY

**CHECKSUM CALCULATING METHOD:**

char CALC\_CHECKSUM(char []data, int data\_length)

{

Unsigned int i = 0;

Char chksum=0;

for(i=0;i<data\_length;i++)

{

chksum += data[i];

}

return(chksum);

}

**CHECKSUM CALCULATION EXAMPLE:**

“CFG,PB,6906,100M,1702432789,032,01-1702432799,02-1702432796,03-1702432797,04-1702432798,05-1702432799,06-1702432760,07-1702432761,08-1702432762,09-1702432763,10-1702432764,11-1702432769,12-1702432766,13-1702432767,14-1702432768,19-1702432769,16-1702432770,17-1702432771,18-1702432772,19-1702432773,20-1702432774,21-1702432779,22-1702432776,23-1702432777,24-1702432778,00-1702432779,26-1702432780,27-1702432781,28-1702432782,28-1702432782,28-1702432782,28-1702432782,0, 0,005M,030M,010M, 0,\*43,&”

NOTE: ALL THE CHARACTERS IN YELLOW COLOUR SHOULD BE ADDED AS PER ABOVE GIVEN ALGORITHM TO GET THE CHAECKSUM SHOWN IN RED.

FILE TRANSFER PROTOCOL

**FIRST PACKET WILL CONTAIN BELOW INFORMATION:**

FTP,FILE\_NAME,FILE\_SIZE,TOTAL\_PACKET\_IN\_FILE,FILE\_CHECKSUM,FILE\_CHECKXOR,PACKET\_SIZE(CHUNK),PACKET\_CHECKSUMPACKET\_CHECKXOR&

Example: FTP,1.wav,20000,40,38,25,500,0x520x130x24

**NOTE**: Last three bytes are in HEX format.

If the calculated CheckSum or CheckXor does not match with the packet checksum and checkxor then you will get Error message “ERROR: FPCHKSUMFAILED” and if match then you will get “OK”

**SECOND PACKET WILL CONTAIN BELOW INFORMATION:**

FTDATA(500 HEX-Bytes)CHECKSUMCHECKXOR

Example: FT10214577852…………………………………221452

Yellow colour data is 500-byte hex data

Red colour data is packet checksum

Green colour data is packet check-xor

Sky Blue colour is ‘&’

**NOTE**: For every Packet response device will “OK” if everything is ok in the packet.

NOTE: BEFORE ALL THE ABOVE COMMUNICATION, PLEASE SET THE MTU TO 517.