|  |
| --- |
| hkpu |
| AVR-CC2530 Interface Protocol |
| [Type the document subtitle] |
|  |
| **hkpu** |
| **7/5/2010** |

|  |
| --- |
| [Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.] |

Contents

[Revision History 3](#_Toc266097957)

[Introduction 3](#_Toc266097958)

[Description 3](#_Toc266097959)

[1. Hardware Setup 3](#_Toc266097960)

[Connection Diagram 4](#_Toc266097961)

[2. Writing Operation from AVR to CC2530 4](#_Toc266097962)

[3. Writing Operation from CC2530 to AVR 4](#_Toc266097963)

[4. Notes 5](#_Toc266097964)

[5. Routines 6](#_Toc266097965)

[A. u08 HAL\_PP\_Data\_Port\_Init(void) 6](#_Toc266097966)

[B. u08 Check\_AVR\_Idle(void) 6](#_Toc266097967)

[C. u08 HAL\_PP\_Data\_Port\_Send(u08 buffer) 6](#_Toc266097968)

[Application Layer 8](#_Toc266097969)

[Description 8](#_Toc266097970)

[1. Length : The Total Length of the Packet 8](#_Toc266097971)

[2. Type : Message Type 8](#_Toc266097972)

[3. PayLoad : The Length of the Payload 8](#_Toc266097973)

[4. Checksum : Checksum 8](#_Toc266097974)

[PayLoad Specification 10](#_Toc266097975)

[Unspecified : variable, value can range from 0-255 10](#_Toc266097976)

[Metering Reading : 10](#_Toc266097977)

[LCD Display : 10](#_Toc266097978)

[Key Press : 10](#_Toc266097979)

[RFID 11](#_Toc266097980)

# Revision History

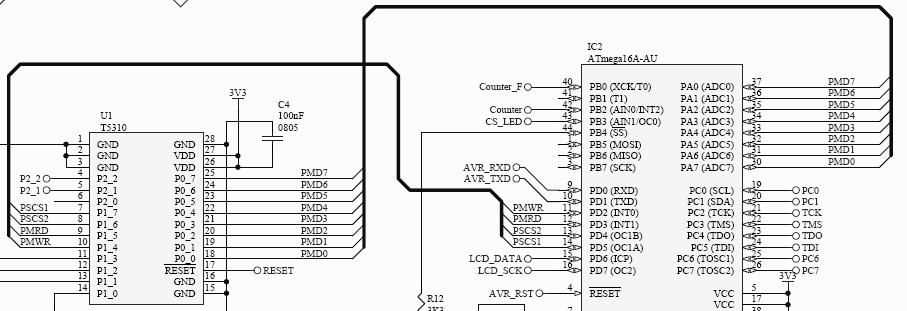
|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Revision** | **Name** |
| **0.2** | 30/6/2010 | 0.2 | Rimsky |
| **0.3** | 21/6/2010 | 0.3 | Gary |

# Introduction

This document is written to explain the physical layer of the AVR-CC2530 parallel interface. As well as the additional layer built on top of the physical interface.

# Description

1. Hardware Setup



## Connection Diagram

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CC2530** | **Function** | **ATmega16L** | | **Description** |
| **Main Board** | **Panel** |
| **P0\_7** | PMD7 | PA7 | PA0 | Parallel Data Port |
| **P0\_6** | PMD6 | PA6 | PA1 |
| **P0\_5** | PMD5 | PA5 | PA2 |
| **P0\_4** | PMD4 | PA4 | PA3 |
| **P0\_3** | PMD3 | PA3 | PA4 |
| **P0\_2** | PMD2 | PA2 | PA5 |
| **P0\_1** | PMD1 | PA1 | PA6 |
| **P0\_0** | PMD0 | PA0 | PA7 |

|  |  |  |  |
| --- | --- | --- | --- |
| **CC2530** | **Function** | **ATmega16L** | **Description** |
| **P1\_7** | BOZ | PD5 | Zigbee request Bus Occupy, 0=Active |
| **P1\_6** | BOA | PD4 | AVR request Bus Occupy, 0=Active |
| **P1\_5** | ACKZ | PD3 | Zigbee Acknowledge, 0=Active |
| **P1\_4** | ACKA | PD2 | AVR Acknowledge, 0=Active |

1. Writing Operation from AVR to CC2530
   1. Check if the Bus is idle
   2. AVR pull PSCS2 to low
   3. Nop();
   4. Check if the Bus is idle again
   5. AVR put data on to the 8bit parallel bus PMD
   6. AVR pull PMWR to low to indicate the placement of the data
   7. CC2530 read the data from databus after observing PMWR pull to low
   8. CC2530 acknowledge the read by pull PMRD to low
   9. AVR restore to initial condition after observing PMRD pulled to low
   10. CC2530 restore to initial condition after observing PMWR pull back to high as part of the initial condition
2. Writing Operation from CC2530 to AVR
   1. Check if the Bus is idle
   2. CC2530 pull PSCS1 to low
   3. CC2530 put data on to the 8bit parallel bus PMD
   4. CC2530 pull PMWR to low to indicate the placement of the data
   5. AVR read the data from databus after observing PMWR pull to low
   6. AVR acknowledge the read by pull PMRD to low
   7. CC2530 restore to initial condition after observing PMRD pulled to low
   8. AVR restore to initial condition after observing PMWR pull back to high as part of the initial condition
3. Notes
   1. CC2530 will not wait in a while loop when observing the PMRD or PMWR to go high or low. If the conditions are not met, CC2530 would break off the writing/reading sequence and try again another time. This is done to prevent any potential mishaps done on the RF routine, which are more important.
   2. AVR on the other hands, will wait for a longer period, but would eventually time out, if CC2530 hadn’t followed the procedures.
4. Routines
   1. u08 HAL\_PP\_Data\_Port\_Init(void)

Initialization Routines set CC2530 or AVR all pins defaults as input and pullup

* 1. u08 Check\_AVR\_Idle(void)

Check Routines for the processors to check if the other processors are busy. Example : Check Operation for AVR

Return 1

CC2530 busy

PSCS1 = 1 AND

PSCS2 = 1?

Return 0

PSCS1 = 0?

PSCS2 = 0?

Return 2

Abnormal Situation

Y

N

Y

Y

N

* 1. u08 HAL\_PP\_Data\_Port\_Send(u08 buffer)

Data Send Routine

The Routines first execute sub-routine Check\_AVR\_Idle() to ensure that the bus is not preoccupied with other data. After the assertion, the routine would pull the CS pin low and then switch the data port into output, put the data on the port, and set the WR pin into low.

Then the Routine would wait for the RD pin to set low, then it would restore all the associated pins into initial condition.

The Routines are almost identical for CC2530 and AVR except that the chip select are different.

# Application Layer

Packet Structure

|  |  |  |  |
| --- | --- | --- | --- |
| Length[1] | Type[1] | Time Out[1] | Payload [Length – 3] |

## Description

1. Length : The Total Length of the Packet

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit** | 0-255 | Number of bytes |

1. Type : Message Type

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit** | 0-255 | Message Type |

* 1. 0 : Unspecified
  2. 1 : Metering Reading (Integer)
  3. 2 : Metering Reading (Character)
  4. 3 : Metering Reading (Hex)
  5. 4 : RFID Reading
  6. 5 : Keypad Reading
  7. 6 : LCD Display
  8. 7 : Penal Reading General Process
  9. 8 : Penal Reading Preheat Process
  10. 9 : Penal Reading Rework Process
  11. 10: Penal Reading InputQTY Process
  12. 11: Metering Reading (Long Integer)
  13. 12: Penal Reading Combine (Card) Process
  14. 13: Penal Reading Separate (Card) Process

1. Time Out : A flag will be set to indicate the message is sent out due to time out

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x00 | Message sent due to user complete input. |
|  | 0x01 | Message sent due to Time Out. |

1. Payload : The Length of the Payload

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0-255 | Data Load |

This is total length of the packet minus 3, which those three bytes are used for length, type and time out. For unspecified message type packet, the additional identified information should be stored in the payload, which would be decoded for further processing.

Payload Specification

Payload is different for different kind of message type

**Penal Reading General Process:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Staff Card[10] | Input Method[1] | Flow Direction[1] | Total Order Card[1] | Order Card[10] | Input Method[1] | QTY [2] | Next Order Card[10]… |

4.1 **Staff Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Staff Card Number |

4.2 **Input Method**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x55 | UART Input |
|  | 0x4B | Keypad Input |

4.3 **Flow Direction**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x30 | Output Device |
|  | 0x31 | Input Device |

4.4 **Total Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **Unsigned char 8 bit** | 0x01~0x05 | Number of Order Card contain in the packet |

4.5 **Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Order Card Number |

4.6 **QTY**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **2 x Unsigned char (2 bytes)** | 0~65535 | QTY respected to associate Order Card |

Example of Payload in Penal Reading General Process:

24 53 54 00 12 4B 00 01 0B D3 D6 01

4F 07 00

30 30 30 33 31 33 32 30 39 33 55

31

05

30 30 31 33 34 37 30 33 30 36 4B 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

2B

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B D3 D6 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 4F | Length, Hex, Range 0x1B to 0x50 |
| 07 | Type, Hex, Range 0x00 to 0x0D |
| 00 | TimeOut, Hex |
| 30 30 30 33 31 33 32 30 39 33 | Staff Card, ASCII (0003132093) |
| 55 | Input Method, ASCII (U) |
| 31 | Flow Direction, ASCII (1) |
| 05 | Total Order Card, Hex, Range 0x01 to 0x05 |
| 30 30 31 33 34 37 30 33 30 36 | Order Card, ASCII (0013470306) |
| 4B | Input Method, ASCII (K) |
| 00 00 | QTY, Hex |
| 2B | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

**Penal Reading Preheat Process:**

|  |  |  |
| --- | --- | --- |
| Staff Card[10] | Input Method[1] | Preheat Status[1] |

3.1 **Staff Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Staff Card Number |

3.2 **Input Method**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x55 | UART Input |
|  | 0x4B | Keypad Input |

3.3 **Preheat Status**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x30 | Preheat Start |

Example of Payload in Penal Reading Preheat Process:

24 53 54 00 12 4B 00 01 0B D3 D6 01

0D 08 00

30 30 30 33 31 33 32 30 39 33 55

30

6D

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B D3 D6 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 0D | Fixed Length, Hex, 0x0D |
| 08 | Type, Hex, Range 0x00 to 0x0D |
| 00 | TimeOut, Hex |
| 30 30 30 33 31 33 32 30 39 33 | Staff Card, ASCII (0003132093) |
| 55 | Input Method, ASCII (U) |
| 30 | Preheat Status, ASCII (1) |
| 6D | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

**Penal Reading Rework Process:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Staff Card[10] | Input Method[1] | Dummy Value[1] | Total Order Card[1] | Order Card[10] | Input Method[1] | QTY [2] | Next Order Card[10]… |

3.1 **Staff Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Staff Card Number |

3.2 **Input Method**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x55 | UART Input |
|  | 0x4B | Keypad Input |

3.3 **Dummy Value**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x30 | Nil |

3.4 **Total Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **Unsigned char 8 bit** | 0x01~0x05 | Number of Order Card contain in the packet |

3.5 **Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Order Card Number |

3.6 **QTY**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **2 x Unsigned char (2 bytes)** | 0~65535 | QTY respected to associate Order Card |

Example of Payload in Penal Reading Rework Process:

24 53 54 00 12 4B 00 01 0B D3 D6 01

28 09 00

30 30 30 33 31 33 32 30 39 33 55

30

02

30 30 31 33 34 37 30 33 30 36 4B 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

1B

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B D3 D6 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 28 | Length, Hex, Range 0x28 to 0x4F |
| 09 | Type, Hex, Range 0x00 to 0x0D |
| 00 | TimeOut, Hex |
| 30 30 30 33 31 33 32 30 39 33 | Staff Card, ASCII (0003132093) |
| 55 | Input Method, ASCII (U) |
| 30 | Dummy value |
| 02 | Total Order Card, Hex, Range 0x02 to 0x05 |
| 30 30 31 33 34 37 30 33 30 36 | Order Card, ASCII (0013470306) |
| 4B | Input Method, ASCII (K) |
| 00 00 | QTY, Hex |
| 1B | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

**Penal Reading InputQTY Process:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Staff Card[10] | Input Method[1] | Dummy Value[1] | Total Order Card[1] | Order Card[10] | Input Method[1] | QTY [2] | Next Order Card[10]… |

3.1 **Staff Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Staff Card Number |

3.2 **Input Method**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x55 | UART Input |
|  | 0x4B | Keypad Input |

3.3 **Dummy Value**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x30 | Nil |

3.4 **Total Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **Unsigned char 8 bit** | 0x01~0x05 | Number of Order Card contain in the packet |

3.5 **Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Order Card Number |

3.6 **QTY**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **2 x Unsigned char (2 bytes)** | 0~65535 | QTY respected to associate Order Card |

Example of Payload in Penal Reading InputQTY Process:

24 53 54 00 12 4B 00 01 0B D3 D6 01

28 0A 00

30 30 30 33 31 33 32 30 39 33 55

30

02

30 30 31 33 34 37 30 33 30 36 4B 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

1C

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B D3 D6 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 28 | Length, Hex, Range 0x28 to 0x4F |
| 0A | Type, Hex, Range 0x00 to 0x0D |
| 00 | TimeOut, Hex |
| 30 30 30 33 31 33 32 30 39 33 | Staff Card, ASCII (0003132093) |
| 55 | Input Method, ASCII (U) |
| 30 | Dummy value |
| 02 | Total Order Card, Hex, Range 0x01 to 0x05 |
| 30 30 31 33 34 37 30 33 30 36 | Order Card, ASCII (0013470306) |
| 4B | Input Method, ASCII (K) |
| 00 00 | QTY, Hex |
| 1C | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

**Penal Reading Combine (Card) Process:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Staff Card[10] | Input Method[1] | Dummy Value[1] | Total Order Card[1] | Order Card[10] | Input Method[1] | QTY [2] | Next Order Card[10]… |

3.1 **Staff Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Staff Card Number |

3.2 **Input Method**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x55 | UART Input |
|  | 0x4B | Keypad Input |

3.3 **Dummy Value**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x30 | Nil |

3.4 **Total Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **Unsigned char 8 bit** | 0x01~0x05 | Number of Order Card contain in the packet |

3.5 **Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Order Card Number |

3.6 **QTY**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **2 x Unsigned char (2 bytes)** | 0~65535 | QTY respected to associate Order Card |

Example of Payload in Penal Reading Combine Process:

24 53 54 00 12 4B 00 01 0B D3 D6 01

28 0C 00

30 30 30 33 31 33 32 30 39 33 55

30

02

30 30 31 33 34 37 30 33 30 36 4B 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

1B

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B D3 D6 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 28 | Length, Hex, Range 0x28 to 0x4F |
| 0C | Type, Hex, Range 0x00 to 0x0D |
| 00 | TimeOut, Hex |
| 30 30 30 33 31 33 32 30 39 33 | Staff Card, ASCII (0003132093) |
| 55 | Input Method, ASCII (U) |
| 30 | Dummy value |
| 02 | Total Order Card, Hex, Range 0x02 to 0x05 |
| 30 30 31 33 34 37 30 33 30 36 | Order Card, ASCII (0013470306) |
| 4B | Input Method, ASCII (K) |
| 00 00 | QTY, Hex |
| 1B | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

**Penal Reading Separate (Card) Process:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Staff Card[10] | Input Method[1] | Dummy Value[1] | Total Order Card[1] | Order Card[10] | Input Method[1] | QTY [2] | Next Order Card[10]… |

3.1 **Staff Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Staff Card Number |

3.2 **Input Method**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x55 | UART Input |
|  | 0x4B | Keypad Input |

3.3 **Dummy Value**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value** | **Representation** |
| **Unsigned char 8 bit** | 0x30 | Nil |

3.4 **Total Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **Unsigned char 8 bit** | 0x01~0x05 | Number of Order Card contain in the packet |

3.5 **Order Card**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Range** | **Representation** |
| **Unsigned char 8 bit Array** | 0x30-0x39  (fixed length 10) | Order Card Number |

3.6 **QTY**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Value (range)** | **Representation** |
| **2 x Unsigned char (2 bytes)** | 0~65535 | QTY respected to associate Order Card |

Example of Payload in Penal Reading Separate (Card) Process:

24 53 54 00 12 4B 00 01 0B D3 D6 01

28 0D 00

30 30 30 33 31 33 32 30 39 33 55

30

02

30 30 31 33 34 37 30 33 30 36 4B 00 00

30 30 31 33 34 37 30 33 30 36 55 00 00

1B

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B D3 D6 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 28 | Length, Hex, Range 0x28 to 0x4F |
| 0D | Type, Hex, Range 0x00 to 0x0D |
| 00 | TimeOut, Hex |
| 30 30 30 33 31 33 32 30 39 33 | Staff Card, ASCII (0003132093) |
| 55 | Input Method, ASCII (U) |
| 30 | Dummy value |
| 02 | Total Order Card, Hex, Range 0x02 to 0x05 |
| 30 30 31 33 34 37 30 33 30 36 | Order Card, ASCII (0013470306) |
| 4B | Input Method, ASCII (K) |
| 00 00 | QTY, Hex |
| 1B | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

Example of Payload in Penal Reading General Process (Time Out):

24 53 54 00 12 4B 00 01 0B D3 D6 01

28 07 01

30 30 30 33 31 33 32 30 39 33 55

00

01

30 30 31 33 34 37 30 33 30 36 4B

9C

0D 0A

When the device is idled over a given period of time, it will be reset to initial stage. If one or more Order Cards are stored in the data stack, the stored data will be sent out AUTOMATICALLY. The message will has the TimeOut flag be set to 0x01, and the QTY will be filled with dummy value (0x00) due to null input.

Example of Payload in Penal Reading General Process (FounTain Testing):

24 53 54 00 12 4B 00 01 0B C8 C5 01

1A 07 00

00 00 00 00 00 00 00 00 00 00 00

31

01

00 00 00 00 00 00 31 31 32 35 4B

29

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B C8 C5 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 1A | Fixed Length = 0x1A, Hex |
| 07 | Fixed Type = 0x07, Hex |
| 00 | TimeOut, Hex |
| 00 00 00 00 00 00 00 00 00 00 | Dummy value |
| 00 | Dummy value |
| 31 | Flow Direction, ASCII (1) |
| 01 | Fixed Total Order Card = 0x01, Hex |
| 00 00 00 00 00 00 31 31 32 35 | Order Card, ASCII (1125), Fill Dummy if string length less than 10 |
| 4B | Input Method, ASCII (K) |
| 00 00 | Dummy value |
| 29 | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

Above are the message samples retuned by the testing panel installed in FounTain. The message type = 0x07(Penal Reading General Process), Staff Card Number and InputMethod are filled with Dummy value (0x00). Users are required to input the order number, and select the direction flow of operation. If no order number is entered, or the string length is less than 10 digits, the remains will be filled by Dummy value. Zigbee message will be sent back to server once the direction flow is entered.

Example of Payload in Penal RFID Card Reading (Debug Testing):

24 53 54 00 12 4B 00 01 0B C8 C5 01

0E 08 00

30 30 31 30 32 36 30 31 38 31 55

31

51

0D 0A

|  |  |
| --- | --- |
| 24 53 54 | Zigbee Signature, Start byte, ASCII ($ST) |
| 00 12 4B 00 01 0B C8 C5 | Zigbee Signature, End Device ID, Hex |
| 01 | Zigbee Signature, Meter ID, Hex |
| 0E | Fixed Length = 0x1A, Hex |
| 08 | Fixed Type = 0x08, Hex |
| 00 | TimeOut, Hex |
| 30 30 31 30 32 36 30 31 38 31 | Dummy value |
| 55 | Dummy value |
| 31 | Distinct flag to Preheat Msg, ASCII (1) |
| 51 | Coordinator Packet CheckSum, Hex, Adding all bytes from Zigbee Signature to the last QTY, Range 0x00 to 0xFF |
| 0D 0A | Zigbee Signature, Stop byte |

Above are the message samples retuned by the panel debug mode. The message type = 0x08(Penal Reading Preheat Process), Staff Card Number is stored and sent back to server if user press the ‘#’ key. To distinct the testing message from Preheat message, the Preheat Status is marked as Completed, which only be generated under this debug mode.