

YET-MF2 UART Reader

User Manual

1. Mifare Standard

- 1024 bytes EEPROM, divided into 16 sectors with 64 bytes on each sector
- 100,000 write endurance cycles
- 10 years data retention
- ISO 14443 A
- 13.56MHz transponder frequency
- 106 kbit baud rate
- Bit-wise anti-collision
- Up to 10 cm operating distance
- 4 byte unique serial number
- Random number generator
- 2 bytes access key per sector
- Individual access condition for each sector
- Purse functionality

2. YET-MF2 Technical Specification

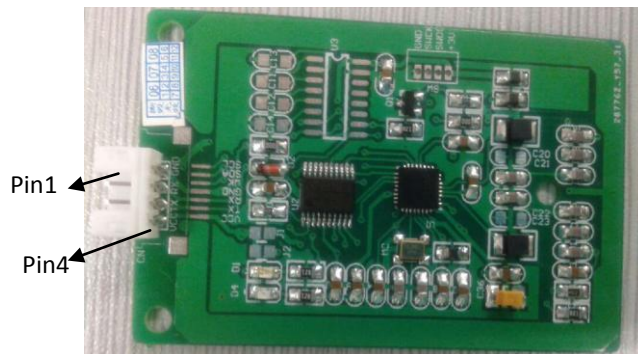
- Power supply: 4.5V~5.5V , 40-100mA
Core: ARM® 32-bit Cortex™-M0 CPU,
frequency up to 48 MHz
Memories 16 Kbytes of Flash memory
4 Kbytes of SRAM with HW parity checking
- Interface: UART
- Transmission speed: 19200 bps (2400 ~115200 setting)
- R/W distance of up to 60mm (up to 100mm with bigger antenna size), depending on TAG
- Storage temperature: -60 °C ~ +150°C
- Operating temperature: -40 °C ~ +85 °C

3. Communication setting

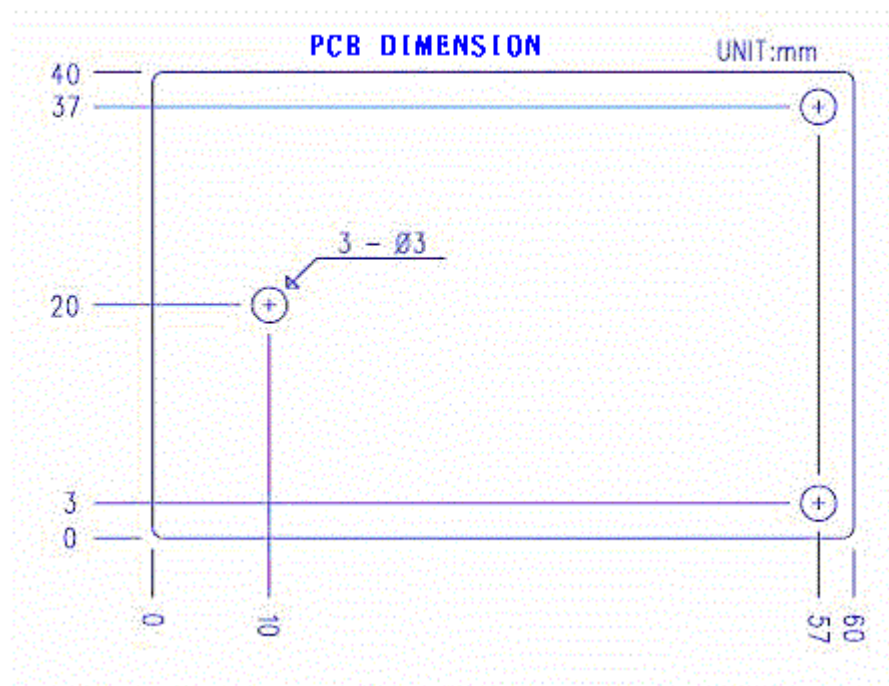
The communication protocol is byte oriented. Both sending and receiving bytes are in hexadecimal format. The communication parameters are as follows,

Baud rate: 19200 bps
 Data: 8 bits
 Stop: 1 bit
 Parity: None
 Flow control: None

4. Dimension & Pin Description



PIN



Pin	Name	description
1	GND	GND
2	RxD	UART RXD
3	TxD	UART TXD

4	VC	4.5~5.5V
---	----	----------

YET-MF2: 4.5~5.5V

Character

Parameter	Min	Type	Max	Units
YET-MF2 voltage	4.5	5.0	5.5	V
Current(Working)	40		60	ma
Current(Sleep)		<10		microamp
Initialization time	50		200	MS
Operatating temperature	- 25		+ 85	℃
Storage temperature	- 40		+ 125	℃

5. UART Setting & Command Protocol

UART Setting :

Transmission rate		Default 19200 , N , 8 , 1				
Data format		Binary HEX “hexadecimal”				
Data package						
Head	Length	Node ID	Function Code	Data ...	XOR	

COMMAND Format :

	Data length (Byte)			
Head	02	Fixed: 0xAA , 0xBB		
Length	02	There are several effective bytes that including XOR follows this column.		
Node ID	02	Destination Node Address Number. xx xx: Low byte first 00 00: Broadcast to each reader.		
Function	02	It will be transmission ability of each different		

code		command . Low byte frist		
Data	00~D0	Data length is not fixed, according to its purpose.		
XOR	01	XOR each byte from Node ID to Last Data byte with 0xFF.		

REPLY DATA FORMAT :

	Data length (Byte)			
Head	02	Fixed: 0xAA , 0xBB		
Length	02	There are several effective bytes that including XOR follows this column.		
Node ID	02	Destination Node Address Number. xx xx: Low byte first 00 00: Broadcast to each reader.		
Function code	02	It will be transmission ability of each different command . Low byte frist		
Status	1	Reply result , if succeed is 0 ,other fail .		
Data	00~D0	Data length is not fixed, according to its purpose.		
XOR	01	XOR each byte from Node ID to Last Data byte		

NOTE: if from “Length” to “XOR ” have a data is “AA” then should follow a data “0x00” ,but length don’t changed.

While a command send and after 100ms no reply then consider this command failed .

Command List :

- 1、Initialize port : 0x0101
- 2、Set device node number : 0x0102
- 3、Read device node number : 0x0103
- 4、Read device Mode : 0x0104
- 5、Set buzzer beep: 0x0106 (Ignore)
- 6、Set Led color : 0x0107
- 7、Set reader working status : 0x0108 (Ignore)
- 8、Reset Reader : 0x010B

- 9、 Set antenna status 。 0x010c
- 10、 Get Reader Reset Status : 0x010D
- 11、 Set Reader to Sleep Mode : 0x0112
- 12、 Mifare Request , 0x0201
- 13、 Mifare anticollision , 0x0202:
- 14、 Mifare Select 0x0203:
- 15、 Mifare Hlta , 0x0204:
- 16、 Mifare Select Card 0x0205:
- 17、 Mifare Authentication1 0x0206 :
- 14、 Mifare Authentication2 0x0207:
- 15、 Mifare Read 0x0208:
- 16、 Mifare Write 0x0209:
- 17、 Mifare Initval 0x020A:
- 18、 Mifare Read Balance 0x020B:
- 19、 Mifare Decrement 0x020C:
- 20、 Mifare Increment 0x020D:
- 21、 Mifare Restore 0x020E:
- 22、 Mifare Backup 0x020F:
- 23、 Mifare UltraLight Anticoll 0x0212:
- 24、 Mifare UltraLight Write 0x0213:
- 25、 Write key store in RC500 EEPROM

1. Initialize port : 0x0101

Function : set baud rate

Format : aa bb 06 00 00 01 01 “Baud_para” “xor Chk”

Baud_parameter :

- 0 = 4800;
- 1 = 9600;
- 3 = 19200;
- 4 = 28800;
- 5 = 38400;
- 6 = 57600;
- 7 = 115200;

sample : Host To Reader;

aa bb 06 00 00 00 01 01 03 03 Set Baud Rate as 19200

Respond: aa bb 06 00 bf ff 01 01 00 40

2. Set device node number : 0x0102 (Ingore)

Aa bb 07 00 00 00 02 01 00 00 03 set device node number = 0x00 00

3. Read device node number : 0x0103 (Ingore)

Aa bb 05 00 00 00 03 01 02 Read device node number.

4. Read device Mode : 0x0104

function : read device mode and version

Host to Reader : aa bb 05 00 00 00 04 01 05

Respond: aa bb 12 00 00 00 04 01 00 43 52 30 33 30 4c 52 2d 31 32 30 33 14

5. Set buzzer beep: 0x0106

Function : beep

format : aa bb 06 00 00 00 06 01 Delay XOR

Delay*10ms beep time , XOR is xor check。

Sample : Host to Reader : aa bb 06 00 00 00 06 01 64 63

Respond : aa bb 06 00 52 51 06 01 00 04

6. Set Led color : 0x0107

Host To Reader;

aa bb 06 00 00 00 07 01 03 05 // set Red&green LED on 。

Respond: aa bb 06 00 bf bf 07 01 00 06

Tenth data is LED parameter ,function as below :

0 = LED_RED Off , LED_GREEN Off

1 = LED_RED On , LED_GREEN = Off

2 = LED_GREEN Off , LED_RED On

3 = LED_GREEN On , LED_RED ON

7. Set reader working status : 0x0108 not use in this device。

8. Reset Reader : 0x010B

Reader should Reset by Watchdog after receive this command, no respond from Reader

Host To Reader : aa bb 05 00 00 00 0b 01 0a

9. Set antenna status 。 0x010c

Host To Reader;

aa bb 06 00 00 00 0c 01 00 0D //set antenna off。

Tenth data is Antenna status parameter ; 0 = Close Filed 1= Open Filed.

9. Request Reader Status , 0x010D

Request Reader RFID CHIP Reset status ,

HOST to Reader : aa bb 05 00 00 00 0d 01 0c

Respond: aa bb 06 00 00 00 0d 01 00 0c Reset RFID ok

Aa bb 06 00 00 00 0d 01 01 0d Reset Fail

10. Set Reader to Sleep MODE , 0x0112

Function : After receive this command Reader should turn to SLEEP mode , no respond ;

Reader & RFID chip are not working after Wake pin Low active or Power On .

Host To Reader : aa bb 05 00 00 00 12 01 13

11. Mifare Request , 0x0201

Function : Request Type a Card
 Format : aa bb 06 00 00 00 01 02 req_code XOR
 req_code=Request mode
 req_code=0x52: request all Type A card In filed
 req_code=0x26: request idle card

sample : Host To Reader:

aa bb 06 00 00 00 01 02 52 51

Respond : aa bb 08 00 52 51 01 02 00 04 00 04

TagType : 0x4400 = ultra_light
 0x0400 = Mifare_One(S50)
 0x0200 = Mifare_One(S70)
 0x4403 = Mifare_DESFire
 0x0800 = Mifare_Pro
 0x0403 = Mifare_ProX

12. Mifare anticollision , 0x0202:

Function : card anticollision
 Format : aa bb 05 00 00 00 02 02 00
 Respond : aa bb 0a 00 52 51 02 02 00 46 ff a6 b8 a4
 “46 ff a6 b8” is card serial number.

13. Mifare Select 0x0203:

Function : Select card
 Format : aa bb 09 00 00 00 03 02 xx xx xx xx XOR
 Ninth to twelfth is card serial number .
 Sample : Host to Reader : aa bb 09 00 00 00 03 02 46 ff a6 b8 a6
 Respond : aa bb 07 00 52 51 03 02 00 08 0a

14. Mifare Hlta , 0x0204:

Function : Hlta card
 Host to reader : aa bb 05 00 00 00 04 02 06
 Respond : aa bb 06 00 52 51 04 02 00 05

15. Mifare Select Card 0x0205 :

Function : Mifare Request + Mifare Anticollision + Select
 If ok respond SNR of cards
 Host to Reader : aa bb 05 00 00 00 05 02 07
 Respond : aa bb 0a 00 00 00 05 02 00 46 ff a6 b8 a0
 46 ff a6 b8 Mifare SNR

16. Mifare Authentication1 0x0206 :

Function : authenticate Card (Key Stroe in RC500)

Format : aa bb xx 00 00 00 06 02 Auth_mode Block KeyEE CHK
Auth_mode=**Authenticate mode**,0x60: Key A ,0x61: Key B
Block= Authenticate block
KeyEE = Key store in RC500 EEPROM group, from 0 to 31 total 32 。
Sample : Host to Reader : aa bb 08 00 00 00 06 02 60 04 01 61
Authenticate Block 4 Key = “group 01 ”
Respond : aa bb 06 00 11 12 06 02 00 07

17. Mifare Authentication2 0x0207:

Function : authenticate Card
Format : aa bb xx 00 00 00 07 02 Auth_mode Block xx xx xx xx xx xx XOR
Auth_mode= **Authenticate mode** ,0x60: KEY A ,0x61: KEY B
Block = Authenticate block
Sample : Host to Reader : aa bb 0d 00 00 00 07 02 60 04 ff ff ff ff ff ff 61
Authenticate Block 4 Key A = “FF FF FF FF FF FF”
Respond : aa bb 06 00 52 51 07 02 00 06

18. Mifare Read 0x0208:

Function : Read block
Format : aa bb 06 00 00 00 08 02 Block XOR
Block = which block want read
Sample : Host to Reader : aa bb 06 00 00 00 08 02 04 0e
Respond : aa bb 16 00 52 51 08 02 00 00 00 00 00 00 00 00 00 00 00 00 12 34 56 78 01
Tenth to sixteenth byte is Data.

19. Mifare Write 0x0209:

Function : write block
Format : aa bb 16 00 00 00 09 02 Block
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 Da Db Dc Dd De Df XOR
Sample : write data to Block4
Host to Reader
aa bb 16 00 00 00 09 02 04 00 00 00 00 00 00 00 00 00 00 00 12 34 78 56 07
Respond: aa bb 06 00 52 51 09 02 00 08

20 Mifare Initval 0x020A:

Function : initialize purse
Format : aa bb 0a 00 00 00 0a 02 Block V0 V1 V2 V3 XOR

21. Mifare Read Balance 0x020B:

Function : read balance
Format : aa bb 06 00 00 00 0B 02 Block XOR
Return four byte balance .

22 Mifare Decrement 0x020C:

Function : decrease balance
Format : aa bb 0a 00 00 00 0c 02 Block V0 V1 V2 V3 XOR

23. Mifare Increment 0x020D:

- Function : increase balance
Format : aa bb 0a 00 00 00 0D 02 Block V0 V1 V2 V3 XOR
- 24. Mifare Restore 0x020E:**
Function : Restore a mifare_one block data to buffer
Format : aa bb 06 00 00 00 0E 02 Block XOR
- 25. Mifare Transfer 0x020F** Function :
Transfer buffer data to a block Format : aa
bb 06 00 00 00 0F 02 Block XOR
- 26. Mifare Backup 0x020F**
Function : Backup a block to another block in same sector .
Format : aa bb 07 00 00 00 0F 02 BlockSource BlockTarget XOR
- 27. Mifare UltraLight Anticoll 0x0212:**
Function : UltraLight anticollision ,respond ultralight ID。
Format : aa bb 05 00 00 00 12 02 CHK
Sample : Host to Reader: aa bb 05 00 00 00 12 02 10
Respond : aa bb 0d 00 52 51 12 02 00 04 1f ae 11 14 7a 00 d9
'04 1f ae 11 14 7a 00' is card serial number .
- 28. Mifare UltraLight Write 0x0213:**
Function : write mifare Ultralight
Format : aa bb 0a 00 00 00 13 02 Page D0 D1 D2 D3 XOR
Page which page want write data ;
D0...D3 data ;
XOR xor check。
Sample : Host to Reader : aa bb 0a 00 00 00 13 02 04 88 88 88 88 15
Respond: aa bb 06 00 52 51 13 02 00 12
- 29. Write key store in RC500 EEPROM .**
Format : aa bb xx 00 00 00 16 02 Auth_mode group xx xx xx xx xx xx XOR
Auth_mode= 0x60: KEY A ,0x61: KEY B (ignore in this command)
Group = 0—31 , write RC500 Eeprom Address from 0x80 to 0x1FF , total 32
group。
“xx xx xx xx xx xx ” = KEY should be writed to EEPROM。
Sample : Host to Reader : aa bb 0d 00 00 00 16 02 60 01 ff ff ff ff ff ff 75
Write group 01 Key = “FF FF FF FF FF FF”
Respond : aa bb 06 00 11 12 16 02 00 17

FCC Warning Statement

Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

This device complies with part 15 of the FCC Rules. Operation is Subject to the following two conditions: (1) This device may not Cause harmful interference, and (2) This device must accept any Interference received, including interference that may cause Undesired operation