智能钥匙低频高频通信帧定义

KeyFob LF-RF Telegram Definition

Version: 1.0

	负责人	签名	时间
编写 (纳恩)：	Eric Zhou		
审核（）：			
批准（）：			

发布: 上海纳恩汽车技术有限公司

Table of Contents

Change History

Table 1. *Change History*

Revision	Date	Change Description	Updated By
V1.0	2013-5-31	Draft	Eric Zhou

Glossary

UID	User ID Device
LF	Low Frequency
UHF	Ultra High Frequency
RSSI	Received Signal Strength Indicator
EOL	End of Line
RKE	Remote Keyless Entry
PEPS	Passive Entry Passive Start
IDE	Identifier
TBD	To Be Define

1 Introduction

1.1 General

1.2 Document Scope

2 RKE Function

2.1 Long RKE Telegram

Preamble ('111111…')	Start bit ('0')	ID
104 bits	1 bit	7 bits

UID number	Function Code	IDE				Battery Status	Button Code
4 bits	4 bits	IDE0	IDE1	IDE2	IDE3		
Byte1		Byte2	Byte3	Byte4	Byte5	Byte6	

SI-Counter				
SI[27…20]	SI[19…12]	SI[11…4]	SI[3…0]	Reserved
Byte7	Byte8	Byte9	Byte10	

Sequence Response				CRC8
SR1	SR2	SR3	SR4	
Byte11	Byte12	Byte13	Byte14	Byte15

Extra Byte
8 bits

✧ Frame Transmit time: (104 + 1 + 7 + 15*8 + 8) bits / (2000) bps = 240 /2000 = 120 ms

✧ Function Code (4 bits) : 0001b (Long RKE Telgram)

✧ UID Number (4 bits) : UID 1 = 0001b
UID 2 = 0010b
UID 3 = 0011b
UID 4 = 0100b

(Remark: Max number 4 UIDs are supported.)

✧ IDE: Serial number of UID. (Factory programmed in PCF7952 EEPROM Page0.)

✧ Battery Status (4 bits) :

VST3	VST2	VST1	VST0	V_{BAT} [V] (typ)
0	0	0	0	1.90
0	0	0	1	1.99
0	0	1	0	2.08
0	0	1	1	2.17
0	1	0	0	2.26
0	1	0	1	2.35
0	1	1	0	2.44
0	1	1	1	2.53
1	0	0	0	2.62
1	0	0	1	2.71
1	0	1	0	2.80
1	0	1	1	2.89
1	1	0	0	2.98
1	1	0	1	3.07
1	1	1	0	3.16
1	1	1	1	3.25

Figure 2.1 Battery Status

✧ Button Code (4 bits) :

0h = free
1h = Lock
2h = Trunk Release
4h = Unlock

✧ SI-Counter: SI-Counter + Button Code 4: ~~0~~0 00 00 02+ 4 => 00 00 00 24

✧ Sequence Response: HitagII encrypted data

✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 toByte14.

2.2 Short RKE Telegram

Preamble ('111…')	Start bit ('0')	ID
10 bits	1 bit	7 bits

UID number	Function Code	IDE				Battery Status	Button Code
4 bits	4 bits	IDE0	IDE1	IDE2	IDE3		
Byte1		Byte2	Byte3	Byte4	Byte5	Byte6	

Extra Byte
8 bits

- Frame Transmit time: (8 + 6 + 6 +6*8 + 8) bits / (2000) bps = 74 /2000 = 37 ms
- Function Code (4 bits) : 2h = 0002b (Short RKE Telgram)
- Button Code (4 bits) :

  0h = free
  1h = Lock
  2h = Trunk Release
  4h = Unlock
  8h = Reserved

- Other data is the same as Long RKE Telegram.

3 PEPS Function

3.1 PEPS Authentication Command

3.1.1 LF Challenge Message

Preamble	SYNC	WUPB			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

Zone and Trigger Info	LF Command	UID Response Sequence	Random Number			
4 bits	4 bits	Max number = 4	Ran1	Ran 2	Ran 3	Ran 4
Byte1		Byte2	Byte3	Byte4	Byte5	Byte6

MAC		CRC8
MAC1	MAC2	
Byte7	Byte8	Byte9

✧ WUP: 32 Bits, Wake Up ID, All UIDs of one vehicle must have the same value.

WUP4_a[7…0] = (WUP4[7…0] << 4) | 0x0F；

(Remark: **Each vehicle must have a UNIQUE Vehicle-ID(VID) data as WUP!** It can be one of the UIDs' IDE or the last 4 bytes VIN code.)

✧ LF Command (4 bits) : 01h = 0001b

✧ Zone and Trigger Info : (TBD)

Passive Entry = 01h;

Passive Start = 02h;

Passive Exit = 03h;

Passive Trunk Close = 04h;

✧ UID Response Sequence : Max 4 UIDs are supported.

UID Response Sequence							
UID4		UID3		UID2		UID 1	
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	1	1	0	0	0	0	1

The UID Response Sequence is: 0->1->2->3 = UID2->UID1->UID3->UID4

- ✧ Random Number: 32 bits Random Number Generated by PEPS Module.
- ✧ MAC: 16 Bits Message Authentication Code, intermediate result of Hitag3 algorithm
- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte8.

3.1.2 UHF Response Message

Preamble ('111…')	Start bit ('0')	ID
10 bits	1 bit	7 bits

UID number	Function Code	Battery Status	UID Info	Response					
4 bits	4 bits			RESP1	RESP2	RESP3	RESP4	RESP5	RESP6
Byte1		Byte2		Byte3	Byte4	Byte5	Byte6	Byte7	Byte8

LF RSSI Value						CRC8
Mant1	Exp1	Mant2	Exp2	Mant3	Exp3	
Byte9	Byte10	Byte11	Byte12	Byte13	Byte14	Byte15

Extra Byte
8 bits

- ✧ Function Code: 4 Bits, 03h = 0011b, PEPS Authentication Response

- ✧ UID Number: 4 Bits, UID1 = 0001b
  UID 2 = 0010b
  UID 3 = 0011b
  UID 4 = 0100b

- ✧ UID Info: 4 Bits, Reserved

- ✧ Battery Status: 4 Bits, See .

- ✧ Response: 48 Bits Message Authentication Code, result of Hitag3 algorithm

- ✧ LF RSSI Value: for 3 antenna, each 2 Byte (exponent and mantissa)

- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte14.

3.2 PEPS Fob Search Command(TBD)

3.2.1 LF Challenge Message

Preamble	SYNC	WUPB			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

Zone and Trigger Info	LF Command	UID Response Sequence
4 bits	4 bits	Max number = 4
Byte1		Byte2

- ✧ LF Command (4 bits) : 02h = 0002b

- ✧ Zone and Trigger Info : (TBD)
  Passive Entry = 01h;

Passive Start = 02h;
Passive Exit = 03h;
Passive Trunk Close = 04h;

✧ UID Response Sequence : Max 4 UIDs are supported.

UID Response Sequence							
UID4		UID3		UID2		UID 1	
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	1	1	0	0	0	0	1

The UID Response Sequence is: 0->1->2->3 = UID2->UID1->UID3->UID4

3.2.2 UHF Response Message

Preamble ('111…')	Start bit ('0')	ID
10 bits	1 bit	7 bits

UID number	Function Code	IDE			
4 bits	4 bits	IDE0	IDE1	IDE2	IDE3
Byte1		Byte2	Byte3	Byte4	Byte5

Extra Byte
8 bits

✧ Function Code: 4 Bits, 0Ah = 1010b

✧ UID Number: 4 Bits, UID1 = 0001b
UID 2 = 0010b
UID 3 = 0011b
UID 4 = 0100b

3.3 PEPS EOL Login

3.3.1 LF Challenge Message

Preamble	SYNC	WUPA			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

Reserved	LF Command	Login ID			
4 bits	4 bits	LID 0	LID1 1	LID1 2	LID1 3
Byte1		Byte2	Byte3	Byte4	Byte5

Reserved		CRC8
1	2	
Byte6	Byte7	Byte8

✧ WUPA: 32 Bits, Wake Up ID for EOL, different with the Authentication command.

Default value = 0x20 0x16 0x12 0xEF, saved in EEPROM;

WUP4_a[7…0] = (WUP4[7…0] << 4) | 0x0F；

✧ LF Command (4 bits) : 03h = 0003b

✧ Login ID (4 bytes) : default value = 0xAB 0x52 0x34 0xEF, saved in EEPROM;

3.3.2 UHF Response Message

Preamble (‘111…’)	Start bit (‘0’)	ID
10 bits	1 bit	7 bits

UID number	Function Code	VID			
4 bits	4 bits	VID 0	VID 1	VID 2	VID 3
Byte1		Byte2	Byte3	Byte4	Byte5

IDE				Battery Status	Reserved	CRC8
IDE0	IDE1	IDE2	IDE3	4 bits	4 bits	
Byte6	Byte7	Byte8	Byte9	Byte10		Byte11

Extra Byte
8 bits

- Function Code: 4 Bits, 04h
- UID Number: 4 Bits, UID1 = 0001b
  UID 2 = 0010b
  UID 3 = 0011b
  UID 4 = 0100b
- Battery Status: 4 Bits .

VST3	VST2	VST1	VST0	V_{BAT} [V] (typ)
0	0	0	0	1.90
0	0	0	1	1.99
0	0	1	0	2.08
0	0	1	1	2.17
0	1	0	0	2.26
0	1	0	1	2.35
0	1	1	0	2.44
0	1	1	1	2.53
1	0	0	0	2.62
1	0	0	1	2.71
1	0	1	0	2.80
1	0	1	1	2.89
1	1	0	0	2.98
1	1	0	1	3.07
1	1	1	0	3.16
1	1	1	1	3.25

Figure 3.1 Battery Status

3.4 PEPS RSSI Calibration

3.4.1 LF Challenge Message

Preamble	SYNC	WUPA			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

LF Command	Reserved	Reserved	Reserved	Reserved	Reserved
8 bits					
Byte1	Byte2	Byte3	Byte4	Byte5	Byte6

CRC8	LF Carrier

Byte7	15ms

- ✧ WUP: 32 Bits, **WUPA,** Wake Up ID for EOL. ***WUP4_a[7…0] = (WUP4[7…0] << 4) | 0x0F；*** the default data : 0x20 0x16 0x12 0xEF
- ✧ LF Command (4 bits) : 0Ah =1010b
- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte6.

3.4.2 UHF Response Message

Reserved	Function Code	Gain Range x	Gain Range y	Gain Range z	RSSI x		
4 bits	4 bits				Mant1	Mant2	Exp
Byte1		Byte2	Byte3	Byte4	Byte5	Byte6	Byte7

RSSI y			RSSI z			Reserved	Reserved	CRC8
Mant1	Mant2	Exp	Mant1	Mant2	Exp			
Byte8	Byte9	Byte10	Byte11	Byte12	Byte13	Byte14	Byte15	Byte16

- ✧ Function Code: 4 Bits, 05h = 0101b,
- ✧ RSSI x/y/z: floating point data, = (RSSIadc - Offset)*GAINrange
- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte15.

3.5 PEPS Write EEPROM

3.5.1 LF Challenge Message

Preamble	SYNC	WUPA			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

- ✧ LF Command (4 bits) : 0Bh

3.5.2 UHF Response Message

✧ Function Code: 4 Bits, 06h

3.6 PEPS Read EEPROM

3.6.1 LF Challenge Message

Preamble	SYNC	WUPA			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

✧ LF Command (4 bits) : 0Ch

3.6.2 UHF Response Message

✧ Function Code: 4 Bits, 07h

3.7 PEPS Calculate Input Voltage out of RSSI Results

3.7.1 LF Challenge Message

Preamble	SYNC	WUPA			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

LF Command	Reserved	Reserved	Reserved	Reserved	Reserved
8 bits					
Byte1	Byte2	Byte3	Byte4	Byte5	Byte6

CRC8	LF Carrier
Byte7	15ms

- ✧ WUP: 32 Bits, **WUPA,** Wake Up ID for EOL. ***WUP4_a[7…0] = (WUP4[7…0] << 4) | 0x0F；*** the default data : 0x20 0x16 0x12 0xEF
- ✧ LF Command (4 bits) : 0Dh
- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte6.

3.7.2 UHF Response Message

Reserved	Function Code	Gain Range	RSSI ADC measurement		RSSI ADC Offset	
4 bits	4 bits		ADCL	ADCH	ADCL	ADCH
Byte1		Byte2	Byte3	Byte4	Byte5	Byte6

RSSI convert			Reserved	Reserved	Reserved	Reserved	CRC8
Mant1	Mant2	Exp					
Byte7	Byte8	Byte9	Byte10	Byte11	Byte12	Byte13	Byte14

- ✧ Function Code: 4 Bits, 08h = 1000b,
- ✧ RSSI ADC measurement: ***RSSIadc***, 12bit resolution
- ✧ RSSI ADC Offset: ***Offset*** , 12bit resolution
- ✧ RSSI convert: floating point data, = (RSSIadc - Offset)*GAINrange
- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte13.

3.8 PEPS RSSI Calibration Verify

3.8.1 LF Challenge Message

Preamble	SYNC	WUPA			
		WUP1	WUP2	WUP3	***WUP4_a***
8 bits	9 bits	8 bits	8 bits	8 bits	8 bits

LF Command	Reserved	Reserved	Reserved	Reserved	Reserved
8 bits					
Byte1	Byte2	Byte3	Byte4	Byte5	Byte6

CRC8	LF Carrier
Byte7	15ms

- ✧ WUP: 32 Bits, **WUPA,** Wake Up ID for EOL. ***WUP4_a[7…0] = (WUP4[7…0] << 4) | 0x0F；*** the default data : 0x20 0x16 0x12 0xEF
- ✧ LF Command (4 bits) : 0Eh
- ✧ CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte6.

3.8.2 UHF Response Message

Reserved	Function Code	Gain Range x	Gain Range y	Gain Range z	RSSI^2 x		
4 bits	4 bits				Mant1	Mant2	Exp
Byte1		Byte2	Byte3	Byte4	Byte5	Byte6	Byte7

RSSI^2 x+y			RSSI ^2 x+y+z			Reserved	Reserved	CRC8
Mant1	Mant2	Exp	Mant1	Mant2	Exp			
Byte8	Byte9	Byte10	Byte11	Byte12	Byte13	Byte14	Byte15	Byte16

- ✧ Function Code: 4 Bits, 09h = 1001b,

- RSSI x/y/z: floating point data, = (RSSIadc - Offset)*GAINrange
- CRC8: 8 bits, CRC8, Polynomial (format: x^8+x^2+x+1), Start value 0x00; CRC calculation start from Byte1 to Byte15.

Appendix