Default BaudRate: 9600bps

Get Infomation								
Command Frame								
Command	Header	Type[1]	Command	PL(MSB)	PL(LSB)[2]	Parameter	Checksum[3]	End
Get Hardware version	BB	00	03	00	01	00	04	7E
Get Software version	BB	00	03	00	01	01	05	7E
Get Manufacturers	BB	00	03	00	01	02	06	7E
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Info Type[4]	Info	
Response of get Hardware version	BB	01	03	00	0B	00	4D ('M')	31 ('1')
	30 ('0')	30 ('0')	20 (' ')	56 ('V')	31 ('1')	2E ('.')	30 ('0')	30 ('0')
	Checksum[5]	End						
	22	7E						
[1] 00: Command Frame; 01: Respone Frame; 02: Notification Frame								
[2] The length of Parameter								
[3] Calculating the sum from type to parameter, take the LSB, is the value of Checksum								
[4] 00: Command Frame; 01: Respone Frame; 02: Notification Frame								
[5] Calculating the sum from type to info, take the LSB, is the value of Checksum								
Single polling instruction								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
Single polling instruction	BB	00	22	00	00	22	7E	
Notification Frame[6]								
Notification	Header	Type	Command	PL(MSB)	PL(LSB)	RSSI	PC(MSB)	PC(LSB)
Single polling instruction	BB	02	22	00	11	C9	34	00
	EPC(MSB)							
	30	75	1F	EB	70	5C	59	04
				EPC(LSB)	CRC(MSB)	CRC(LSB)	Checksum	End
	E3	D5	0D	70	3A	76	EF	7E
Response Frame[7]								
Response	Header	Type[1]	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Error	BB	01	FF	00	01	15	16	7E
[6] The chip will return a notification with RSSI, PC, EPC and CRC, if the CRC is correct. You will get multiple notification if you offer more RFID.								
[7] The chip will return a response with a error code 0x15, if the CRC is not correct or no RFID read.								
Multiple polling instructions								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Reserved	CNT(MSB)	CNT(LSB)[8]
Multiple polling instructions	BB	00	27	00	03	22	27	10
	Checksum	End						
	83	7E						
Notification Frame								
Notification	Header	Type	Command	PL(MSB)	PL(LSB)	RSSI	PC(MSB)	PC(LSB)
	BB	02	22	00	11	C9	34	00

Multiple polling instructions	EPC(MSB)							
	30	75	1F	EB	70	5C	59	04
				EPC(LSB)	CRC(MSB)	CRC(LSB)	Checksum	End
	E3	D5	0D	70	3A	76	EF	7E
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Error	BB	01	FF	00	01	15	16	7E
[8] 0-65535 times								
Stop multiple polling instructions								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
Stop multiple polling	BB	00	28	00	00	28	7E	
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	28	00	01	00	2A	7E
Set the SELECT parameter instruction								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	SelParam[*]	Ptr(MSB)[*]	
Set the SELECT parameter	BB	00	0C	00	13	01	00	00
		Ptr(LSB)	MaskLen[*]	Truncate	Mask(MSB)			
	00	20	60	00	30	75	1F	EB
								Mask(LSB)
	70	5C	59	04	E3	D5	0D	70
	Checksum	End						
	AD	7E						
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	0C	00	01	00	0E	7E
SelParam: 2'b00 RFU; 2'b01 EPC; 2'b10 TID; 2'b11 USER;								
Ptr: begin from EPC (For example 0x00000020 bits)								
MaskLen: For example 0x60 is 96 bits, 6 words								
Get the SELECT parameter								
Command Frame								
Command	Header	Type[1]	Command	PL(MSB)	PL(LSB)	Checksum	End	
Get the SELECT parameter	BB	00	0B	00	00	0B	7E	
Response Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	SelParam[*]	Ptr(MSB)[*]	
Respons	BB	00	0B	00	13	01	00	00
		Ptr(LSB)	MaskLen[*]	Truncate	Mask(MSB)			
	00	20	60	00	30	75	1F	EB
								Mask(LSB)
	70	5C	59	04	E3	D5	0D	70
	Checksum	End						
	AD	7E						

Set the SELECT mode								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Mode	Checksum	End
Stop multiple polling	BB	00	12	00	01	01	14	7E
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	0C	00	01	00	0E	7E

Set communication baud rate								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Pow(MSB)	Pow(LSB)[9]	Checksum
Set communication baud rate	BB	00	11	00	02	00	C0	D3
	End							
	7E							

[9] BaudRate/100. example: 19200/100=192=0xC0

Set work area								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Region[10]	Checksum	End
Set work area	BB	00	07	00	01	01	09	7E
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	07	00	01	00	09	7E

[10] 01: China 900MHz; 04: China 800MHz; 02: US; 03: EU; 06: Korea

Acquire work locations								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
Acquire work locations	BB	00	08	00	00	08	7E	
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Region[10]	Checksum	End
work locations	BB	01	08	00	01	01	0B	7E

Set up working channel								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	CH Index[11]	Checksum	End
Set up working channel	BB	00	AB	00	01	01	AC	7E
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	AB	00	01	00	AD	7E

[11] CH_Index(CN,900MHz) = (Freq_CH-920.125M)/0.25M; CH_Index(CN,800MHz) = (Freq_CH-840.125M)/0.25M;
CH_Index(US) = (Freq_CH-902.25M)/0.5M; CH_Index(EU) = (Freq_CH-865.1M)/0.2M; CH_Index(Korea) = (Freq_CH-917.1M)/0.2M

Get the working channel								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
Get the working channel	BB	00	AA	00	00	AA	7E	
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter[12]	Checksum	End
working channel	BB	01	AA	00	01	00	AC	7E

[12] Freq_CH(CN,900MHz) = CH_Index * 0.25M + 920.125M; Freq_CH(CN,800MHz) = CH_Index * 0.25M + 840.125M;
Freq_CH(US) = CH_Index * 0.5M + 902.25M; Freq_CH(EU) = CH_Index * 0.2M + 865.1M; Freq_CH(Korea) = CH_Index * 0.2M + 917.1M

Set to automatic frequency hopping mode

Command Frame

Command	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter[13]	Checksum	End
Set to automatic frequency hopping mode	BB	00	AD	00	01	FF	AD	7E

Response Frame

Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	AD	00	01	00	AF	7E

[13] FF: Set to automatic frequency hopping mode; 00: Disable automatic frequency hopping mode

Insert the working channel

Command Frame

Command	Header	Type	Command	PL(MSB)	PL(LSB)	CH Cnt[14]	CH list (MSB)	
Insert the working channel	BB	00	A9	00	06	05	01	02
			CH list (LSB)	Checksum	End			
	03	04	05	C3	7E			

Response Frame

Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	A9	00	01	00	AB	7E

[14] Channel Count(The chip will set to automatic frequency hopping mode if set to 0)

Acquire transmitting power

Command Frame

Command	Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
Acquire transmitting power	BB	00	B7	00	00	B7	7E	

Response Frame

Response	Header	Type	Command	PL(MSB)	PL(LSB)	Pow(MSB)	Pow(LSB)[15]	Checksum
transmitting power	BB	01	B7	00	2	7	D0	91
	End							
	7E							

[15] Convert to decimal(For example 0x07D0 = 2000 = 20dBm)

Set the transmitting power

Command Frame

Command	Header	Type	Command	PL(MSB)	PL(LSB)	Pow(MSB)	Pow(LSB)[15]	Checksum
Set the transmitting power	BB	00	B6	00	2	7	D0	8F
	End							
	7E							

Response Frame

Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	B6	00	01	00	B8	7E

Set up transmitting continuous carrier

Command Frame

Command	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter[16]	Checksum	End
Set up transmitting continuous carrier	BB	00	B0	00	01	FF	B0	7E

Response Frame

Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End

Success	BB	01	B0	00	01	00	B2	7E
[16] FF: enable continuous carrier; 00: Disable continuous carrier								
Module hibernation[17]								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
Module hibernation	BB	00	17	00	00	17	7E	
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
Success	BB	01	17	00	01	00	19	7E
[17] The chip will wake up if you send data through serial								
Read label data storage area								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
Read label data storage area	BB	00	39	00	9	00	00	FF
	AP(LSB)	MemBank	SA(MSB)	SA(LSB)	DL(MSB)	DL(LSB)	Checksum	End
	FF	03	00	00	00	02	45	7E
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
Success	BB	01	39	00	13	0E	34	00
	EPC(MSB)							
	30	75	1F	EB	70	5C	59	04
				EPC(LSB)	Data(MSB)			Data(LSB)
	E3	D5	0D	70	12	34	56	78
	Checksum	End						
	B0	7E						
Failed	Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	Checksum	End
	BB	01	FF	00	01	09	0A	7E
Write the label data store								
Command Frame								
Command	Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
Write the label data store	BB	00	49	00	0D	00	00	FF
	AP(LSB)	MemBank	SA(MSB)	SA(LSB)	DL(MSB)	DL(LSB)	DT(MSB)	
	FF	03	00	00	00	02	12	34
		DT(LSB)	Checksum	End				
	56	78	6D	7E				
Response Frame								
Response	Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
Success	BB	01	49	00	10	0E	34	00
	EPC(MSB)							
	30	75	1F	EB	70	5C	59	04
				EPC(LSB)	Parameter	Checksum	End	
	E3	D5	0D	70	00	A9	7E	
Label not read	Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	Checksum	End
	BB	01	FF	00	01	10	0A	7E

Access Password Wrong	Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	UL	PC(MSB)
	BB	01	FF	00	10	16	0E	34
	PC(LSB)	EPC(MSB)						
	00	30	75	1F	EB	70	5C	59
					EPC(LSB)	Checksum	End	
	04	E3	D5	0D	70	75	7E	
Return error codes according to EPC Gen2 Protocol	Header	Type	Command	PL(MSB)	PL(LSB)	Error Code[18]	UL	PC(MSB)
	BB	01	FF	00	10	B3	0E	34
	PC(LSB)	EPC(MSB)						
	00	30	75	1F	EB	70	5C	59
					EPC(LSB)	Checksum	End	
	04	E3	D5	0D	70	12	7E	

[18] Error Code = error codes | 0xB0