13.56M RFID Reader QU-60x Serial

Manual

(version 1.5)

ISO14443A

- **English**

Web: www.quio-rfid.de
EMAIL: kontakt@quio-rfid.de
Phone/Fax: +49 (0) 202 404329

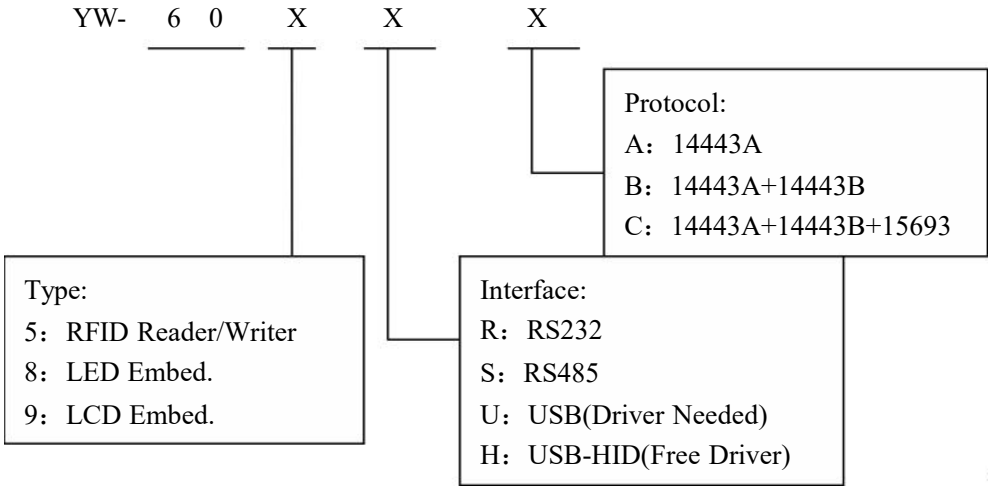
Content

1 Introduction

YW-605 RFID reader/Writer is the 13.56M HF tags Reader and Writer. She Can support ISO14443A, ISO14443B and ISO15693 RFID tags.

2 YW-605 Serial

13.56M RFIDReader/Writer



3 Feature

☞Distance of read tags is 5-10cm

☞DLL API supported. VC, VB, Delphi, C++Builder, C#.net, WEB example

☞Beep and LED embed, which can be controlled by API.

4 Specifications

- Baud:19200BPS
- Power: DC5V ± 10%
- Max Con: 1.5W
- Temp: -30℃ ～ +70℃
- Humidity: 35% ～ 95%
- Size: 120 * 84 * 25 (mm)
- Weight: 100g

5 Program Manual

DLL API and functions

1. Get the version of DLL

prototype:int stdcall YW_GetDLLVersion(void);

Param:无

Return:version Number is success, else is fail.

2. DES

prototype:int stdcall DES(unsigned char cModel, unsigned char *pkey,
unsigned char *in, unsigned char *out);

Param:

Param	Type	
cModel	unsigned char	Direction: 0 Encryption，1 Decryption

pkey	unsigned char*	Key , 8 Bytees
in	unsigned char*	Data ,8 bytes
out	unsigned char*	Data after Des, 8bytes

Return:0

3. 3DES

prototype:int stdcall DES3(unsigned char cModel, unsigned char *pKey, unsigned char *In, unsigned char *Out);

Param:

Param	Type	
cModel	unsigned char	Direction: 0 Encryption，1 Decryption
pkey	unsigned char*	Key, 16Bytes
in	unsigned char*	Data ,8 bytes
out	unsigned char*	Data after Des, 8bytes

Return:0

4. 3DES with vector

prototype:int stdcall DES3_CBC(unsigned char cModel, unsigned char *pKey,unsigned char *In, unsigned char *Out, unsigned char *pIV);

Param:

Param	Type	
cModel	unsigned char	Direction: 0 Encryption, 1 Decryption
pkey	unsigned char*	Key, 16Bytes
in	unsigned char*	Data ,8 bytes
out	unsigned char*	Data after Des, 8bytes
pIV	unsigned char*	Vector , 8Bytes

Return:0

5. Initial Serial Port

prototype:int stdcall YW_ComInitial(int PortIndex, int Bound);

Param:

Param	Type	
PortIndex	int	Port Number, 1--255
Bound	int	Baud, 2400—115200, 19200default

Return:1 success, 0 fail

6. Free Serial port

prototype:int stdcall YW_ComFree(void);

Param:

Return:1 success, 0 fail

7. Initial HID Port

prototype:int stdcall YW_USBHIDInitial(void);

Param:

Return:1 success, 0 fail

8. Free HID Port

prototype:int stdcall YW_USBHIDFree(void);

Param:

Return:1 success, 0 fail

9. Change the serial port baud

prototype:int stdcall YW_ComNewBound(int ReaderID ,int NewBound);

Param:

Param	Type	
ReaderID	int	Reader ID, default ID=0
NewBound	int	New Baud 0x01->9600bps 0x02->14400bps 0x03->19200bps 0x04->28800bps 0x05->38400bps 0x06->57600bps 0x07->115200bps

Return:1 success, 0 fail

10. Set Reader ID

prototype:int stdcall YW_SetReaderID(int OldID, int NewID);

Param:

Param	Type	
OldID	int	Old Reader ID
NewID	int	New Read ID

Return:1 success, 0 fail

11. Get Reader ID

prototype:int stdcall YW_GetReaderID(int ReaderID);

Param:

Param	Type	
ReaderID	int	ReaderID=0

Return:>=0 success and is Reader ID, other is fail

12. Get Reader Fireware version

prototype:int stdcall YW_GetReaderVersion(int ReaderID);

Param:

Param	Type	
ReaderID	int	Reader ID

Return:>=0 success and is version, other is fail

13. Get Reader serial No.

prototype:int stdcall YW_GetReaderSerial(int ReaderID, char *ReaderSerial);

Param:

Param	Type	
ReaderID	int	Reader ID
ReaderSerial	Char *	Reader Serial, 8 Bytes

Return:1 success, 0 fail

14. Beep control

prototype:int stdcall YW_Buzzer(int ReaderID,int Time_ON, int Time_OFF, int Cycle);

Param:

Param	Type	
ReaderID	int	Reader ID
Time_ON	int	Beep on time, unit:100ms
Time_OFF	int	Beep off time, unit:100ms
Cycle	int	The weeks of the beep on and off

Return:1 success, 0 fail

15. LED control

prototype:int stdcall YW_Led(int ReaderID,int LEDIndex, int Time_ON, int Time_OFF, int Cycle, int LedIndexOn);

Param:

Param	Type	
ReaderID	int	Reader ID
LEDIndex	int	LED index 01:Read 02:Green 04:Yellow
Time_ON	int	LED On Time, unit:100ms
Time_OFF	int	LED Off Time, unit:100ms
Cycle	int	The weeks of the Led on and off
LedIndexOn	int	The index of Led on last: 00:all off 01:Read 02:Green 04:Yellow

Return:1 success, 0 fail

16. Set Content of 8 LED display

prototype:int stdcall YW_LEDDisplay(int ReaderID,int Alignment,char *LEDText);

Param:

Param	Type	
ReaderID	int	Reader ID
Alignment	int	Alignment 1:Left 2:Center 3:Right
LEDText	Char *	The string to be displayed The char can be below 0123456789AbCdEF.-

Return:1 success, 0 fail

17. Set the Status of Antenna

prototype:int stdcall YW_AntennaStatus(int ReaderID,bool Status);

Param:

Param	Type	
ReaderID	int	Reader ID
Status	bool	True: Open RF Antenna False:Close RF Antanna

Return:1 success, 0 fail

18. Set Work Mode of Tags

prototype:int stdcall YW_SearchCardMode(int ReaderID,int SearchMode);

Param:

Param	Type	
ReaderID	int	Reader ID
SearchMode	char	Tags 0x41-----ISO14443A 0x42----- ISO14443B 0x31----- ISO15693 0x53------ST Serail 0x52------AT88RF020

Return:1 success, 0 fail

Functions of ISO14443A Tags

19. Request card of ISO14443A

prototype:intstdcall YW_RequestCard(intReaderID,charRequestMode,
unsigned short *CardType);

Param:

Param	Type	
ReaderID	int	Reader ID
RequestMode	char	Request Mode 0x52----- All Card 0x26----- Active Card
CardType	unsigned short *	Return Card Type 0x4400 = Ultralight/UltraLight C /MifarePlus(7Byte UID) 0x0400 = Mifare Mini/Mifare 1K (S50) /MifarePlus(4Byte UID) 0x0200 = Mifare_4K(S70)/ MifarePlus(4Byte UID) 0x0800 = Mifare_Pro 0x0403 = Mifare_ProX

		0x4403 ->Mifare_DESFire 0x4200 -> MifarePlus(7Byte UID)

Return:1 success, 0 fail

20. AntiCollide of ISO14443A Tags

prototype:int stdcall YW_AntiCollide(int ReaderID,unsigned char *LenSNO, unsigned char *SNO)

Param:

Param	Type	
ReaderID	int	Reader ID
LenSNO	unsigned char*	Length of card No
SNO	unsigned char *	The card No of card requested

Return:1 success, 0 fail

21. Select Card of ISO14443A

prototype:int stdcall YW_CardSelect(int ReaderID,char LenSNO, unsigned char *SNO)

Param:

Param	Type	
ReaderID	int	Reader ID
LenSNO	unsigned char	Length of SNO
SNO	unsigned char *	Card No. of selected

Return:1 success, 0 fail

22. AntiCollide and Select a Card of ISO14443A

prototype:int stdcall YW_AntiCollideAndSelect(int ReaderID, unsigned char MultiCardMode, unsigned char *CardMem, int *SNLen, unsigned char *SN);

Param:

Param	Type	
ReaderID	int	Reader ID
MultiCardMode	unsigned char	Return mode of Multi Card 0: return error 1:return a Card
CardMem	unsigned char *	Card Memory
SNLen	int *	Length of SN
SN	unsigned char *	Card No.

Return:1 success, 0 fail

23. Request AntiCollide and Select a card of ISO14443A

prototype:int stdcall YW_RequestAntiandSelect(int ReaderID,int SearchMode,int MultiCardMode,unsigned short *ATQA,unsigned char *SAK,unsigned char *LenSNO,unsigned char *SNO);

Param:

Param	Type	
ReaderID	int	Reader ID
RequestMode	unsigned char	Request Mode 0x52----- All Card 0x26----- Active Card
MultiCardMode	unsigned char	Return mode of Multi Card 0: return error 1:return a Card

ATQA	unsigned short *	ATQA
SAK	unsigned char *	SAK
SNLen	int *	Length of SN
SN	unsigned char *	Card No. selected.

Return:1 success, 0 fail

24. AntiCollide Card with level of ISO14443A

prototype:int stdcall YW_AntiCollide_Level(int ReaderID,int Leveln,char *LenSNO, unsigned char *SNO);

Param:

Param	Type	
ReaderID	int	Reader ID
Leveln	int	Level of AntiCollide, <=3
LenSNO	unsigned char*	Length of SNO
SNO	unsigned char *	Card No. antiCollided

Return:1 success, 0 fail

25. Select Card with level of ISO14443A

prototype:int stdcall YW_SelectCard_Level(int ReaderID,int Leveln,unsigned char *SAK)

Param:

Param	Type	
ReaderID	int	Reader ID
Leveln	int	Level of Select card, <=3
SAK	unsigned char*	SAK

Return:1 success, 0 fail

Functions of S50/S70 Tags

26. Download the key to flash

prototype:int stdcall YW_DownLoadKey(int ReaderID, int KeyIndex,char * Key);

Param:

Param	Type	
ReaderID	int	Reader ID
KeyIndex	int	The index of Key, 0..31
Key	char *	Key(6 Byes)

Return:1 success, 0 fail

27. Authrize Key with Download Key

prototype:int stdcall YW_KeyDown_Authorization (int ReaderID, char KeyMode , int BlockAddr,int KeyIndex);

Param:

Param	Type	
ReaderID	int	Reader ID
KeyMode	char	KeyMode=0x60 Key A KeyMode=0x61 Key B
BlockAddr	int	Block ID of Authrized
KeyIndex	int	Key Index, 0..31

Return: 1 success, 0 fail

28. Authrize Key with Key

prototype:int stdcall YW_KeyAuthorization (int ReaderID,char

KeyMode, int BlockAddr, unsigned char *Key);

Param:

Param	Type	
ReaderID	int	Reader ID
KeyMode	char	KeyMode=0x60 Key A KeyMode=0x61 Key B
BlockAddr	int	Block ID of Authrized
Key	unsigned char *	Key(6 Byes)

Return: 1 success, 0 fail

29. Read a Block

prototype:int stdcall YW_ReadaBlock (int ReaderID, int BlockAddr, int LenData, unsigned char *Data);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID
LenData	int	Length of data
Data	unsigned char *	Data of read

Return: 1 success, 0 fail

30. Write a Block

prototype:intstdcall YW_WriteaBlock (intReaderID, intBlockAddr, int LenData, unsigned char *Data);

Param:

Param	Type	
ReaderID	int	Reader ID

BlockAddr	int	Block ID
LenData	int	Length of Data
Data	unsigned char *	Data will be written

Return: 1 success, 0 fail

31. Initial e-purse of block

prototype:int stdcall YW_Purse_Initial (int ReaderID,int BlockAddr,int IniMoney);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID
IniMoney	int	The initial value of e-purse

Return: 1 success, 0 fail

32. Read value of e-purse

prototype:int stdcall YW_Purse_Read (int ReaderID,int BlockAddr,int *Money);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID
Money	Int *	The value read

Return: 1 success, 0 fail

33. Decrease value of e-purse

prototype:int stdcall YW_Purse_Decrease (int ReaderID,int

BlockAddr,int Decrement);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID
Decrement	Int	The value will be decreased

Return: 1 success, 0 fail

34. Increase value of e-purse

prototype:int stdcall YW_Purse_Charge (int ReaderID,int BlockAddr,int Charge);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID
Charge	Int	The value will be increased

Return: 1 success, 0 fail

35. Restore

prototype:int stdcall YW_Restore (int ReaderID,int BlockAddr);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID

Return: 1 success, 0 fail

36. Transfer

prototype:int stdcall YW_Transfer (int ReaderID,int BlockAddr);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockAddr	int	Block ID

Return: 1 success, 0 fail

37. Read multi blocks

prototype:int stdcallYW_ReadM1MultiBlock(int ReaderID, int StartBlock, int BlockNums, int *LenData, char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
StartBlock	int	Block begin
BlockNums	int	Number of block
LenData	Int*	Length of data
Data	unsigned char *	Data read

Return: 1 success, 0 fail

38. Write Multi Blocks

prototype:int stdcallYW_WriteM1MultiBlock(int ReaderID, int StartBlock, int BlockNums, int LenData, char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
StartBlock	int	Block begin
BlockNums	int	Number of block

LenData	int	Length of data
Data	unsigned char *	Data will be written

Return: 1 success, 0 fail

Functions of UltrLight Tags

39. read Block of UltraLight Tags

prototype:int stdcall YW_UltraLightRead(int ReaderID, int BlockID, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	Block ID
pData	unsigned char *	Data read(4 Bytes)

Return: 1 success, 0 fail

40. Write Block of UltraLight

prototype:int stdcall YW_UltraLightWrite(int ReaderID, int BlockID, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	Block ID
pData	unsigned char *	Data will be written(4 bytes)

Return: 1 success, 0 fail

Functions of ISO14443A CPU Tags

41. Reset CPU Card of ISO14443A

prototype:int stdcall YW_TypeA_Reset(int ReaderID, unsigned char Mode, unsigned char MultiMode, int *rtLen, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
Mode	unsigned char	Request Mode 0x52----- All Card 0x26----- Active Card
MultiMode	unsigned char	Return mode of Multi Card 0: return error 1:return a Card
rtLen	int *	Length of pData
pData	unsigned char *	Data of reset infomation

Return: 1 success, 0 fail

42. Excute COS Command of ISO14443A CPU Card

prototype:int stdcall YW_TypeA_COS(int ReaderID, int LenCOS, unsigned char *Com_COS, int *rtLen, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
LenCOS	unsigned char*	Length of COS
Com_COS	unsigned	COS Command

	char*	
rtLen	int *	Length of pData
pData	unsigned char *	Data returned after excute COS command

Return: 1 success, 0 fail

Fucntions of Mifare Plus Tags

43. Write Data at Level 0 of Mifare Plus Tags

prototype:intstdcall YW_MFP_L0_WritePerso(int ReaderID, int Address, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
Address	unsigned char	The address of data written
Com_ pData	unsigned char*	Data will be written (16 bytes)

Return: 1 success, 0 fail

44. CommitPerso From Level 0 to Level 3 of Mifare Plus Tags

prototype:int stdcallYW_MFP_L0_CommitPerso(int ReaderID);

Param:

Param	Type	
ReaderID	int	Reader ID

Return: 1 success, 0 fail

45. Switch Level from low level to high level of Mifare Plus tags

prototype:int stdcallYW_MFP_SwitchToLevel(int ReaderID, int DesLevel,unsigned char *SwitchKey);

Param:

Param	Type	
ReaderID	int	Reader ID
DesLevel	unsigned char	Level of Destination (max id 3)
SwitchKey	unsigned char*	Switch key (16 bytes)

Return: 1 success, 0 fail

46. Authorization at level 3 of Mifare Plus tags

prototype:int stdcallYW_MFP_L3_Authorization(int ReaderID, int KeyMode,int BlockID,unsigned char *Key);

Param:

Param	Type	
ReaderID	int	Reader ID
KeyMode	unsigned char	KeyMode=0x60 Key A KeyMode=0x61 Key B
BlockID	unsigned char	Block ID
Key	unsigned char*	Key(16 bytes)

Return: 1 success, 0 fail

47. Read Data at level 3 of Mifare Plus Tags

prototype:int stdcall YW_MFP_L3_Read(int ReaderID, int StartBlock,int BlockNums,int *DataLen, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
StartBlock	int	Block begin
BlockNums	int	Number of blocks
DataLen	int*	Length of pData
pData	unsigned char*	Data read

Return: 1 success, 0 fail

48. Write Data at level 3 of Mifare Plus Tags

prototype:int stdcall YW_MFP_L3_Write(int ReaderID, int StartBlock,int BlockNums, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
StartBlock	int	Block begin
BlockNums	int	Number of blocks
pData	unsigned char*	Data will be written , the length is 16*BlockNums

Return: 1 success, 0 fail

49. Initial e-purse at level 3 of Mifare Plus tags

prototype:int stdcall YW_MFP_L3_Purse_Initial(int ReaderID, int BlockID,int InitialValue);

Param:

Param	Type	
ReaderID	int	Reader ID

BlockID	int	Block ID
InitialValue	int	Initial value of e-purse

Return: 1 success, 0 fail

50. read e-purse at level 3 of Mifare Plus tags

prototype:int stdcall YW_MFP_L3_Purse_Read(int ReaderID, int BlockID,int *Value);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	Block ID
Value	Int *	The value read

Return: 1 success, 0 fail

51. Decrease value at level 3 of Mifare Plus tags

prototype:int stdcall YW_MFP_L3_Purse_Charge(int ReaderID, int BlockID,int Value);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	Block ID
Value	Int	The value will be decreased

Return: 1 success, 0 fail

52. Increase value at level 3 of Mifare Plus tags

prototype:int stdcall YW_MFP_L3_Purse_Decrease(int ReaderID, int BlockID,int Value);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	Block ID
Value	Int	The value will be increased

Return: 1 success, 0 fail

53. Backup e-purse atr level 3 of Mifare Plus Tags

prototype:int stdcall YW_MFP_L3_Purse_Backup(int ReaderID, int BlockID,int DesBlockID);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	Block id backup from
DesBlockID	Int	Destination Block id backup to

Return: 1 success, 0 fail

54. The First Authorization of Common read at level 3 of Mifare Plus Tags

prototype:intstdcall YW_MFP_Authorization_First(int ReaderID, int AESKeyAddr,unsigned char *AESKey);

Param:

Param	Type	
ReaderID	int	Reader ID
AESKeyAddr	int	The Address of authorized
AESKey	unsigned char *	Key(16 bytes)

Return: 1 success, 0 fail

55. The Second Authorization of Common read at level 3 of Mifare Plus Tags

prototype:int stdcall YW_MFP_Authorization_Follow(int ReaderID, int AESKeyAddr,unsigned char *AESKey);

Param:

Param	Type	
ReaderID	int	Reader ID
AESKeyAddr	int	The Address of authorized
AESKey	unsigned char *	Key(16 bytes)

Return: 1 success, 0 fail

56. Common read at level 3 of Mifare Plus tags

prototype:int stdcall YW_MFP_CommonRead(int ReaderID, int BlockID, int BlockNums,int *DataLen,unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	The Block Address
BlockNums	int	Number of blocks
DataLen	Int*	Length of pData
pData	unsigned char *	Data read

Return: 1 success, 0 fail

57. Common write at level 3 of Mifare Plus tags

prototype:int stdcall YW_MFP_CommonWrite(int ReaderID, int BlockID, int BlockNums, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
BlockID	int	The Block Address
BlockNums	int	Number of blocks
pData	unsigned char *	Data will be written , the length is 16* BlockNums

Return: 1 success, 0 fail

Functions of SAM

58. SAM Baud

prototype: int __stdcall YW_SAM_ResetBaud(int ReaderID, int SAMIndex, int BaudIndex);

Param:

Param	Type	
ReaderID	int	Reader ID
SAMIndex	int	SAM Index
BaudIndex	int	0x00->9600 (Default) 0x01->19200 0x02->38400 0x03->55800 0x04->57600 0x05->115200

Return: 1 success, 0 fail

59. SAM Reset

prototype: int __stdcall YW_SAM_Reset(int ReaderID, int SAMIndex, int *rtLen, unsigned char *pData);

Param:

Param	Type	

ReaderID	int	Reader ID
SAMIndex	int	SAM Index
rtLen	int *	Length of PData
pData	unsigned char *	Data return of Sam Reset

Return: 1 success, 0 fail

60. Excute COS Command of SAM

prototype: int __stdcallYW_SAM_COS(int ReaderID, int SAMIndex, int LenCOS, unsigned char *Com_COS, int *rtLen, unsigned char *pData);

Param:

Param	Type	
ReaderID	int	Reader ID
SAMIndex	int	SAM Index
LenCOS	int	Length of COS Command
Com_COS	unsigned char *	COS Command
rtLen	unsigned char *	Length of pData
pData	unsigned char *	Data return after excution of COS Command

Return: 1 success, 0 fail

61. SAM PPS Baud

prototype: int __stdcall YW_SAM_PPSBaud(int ReaderID, int SAMIndex, int BaudIndex);

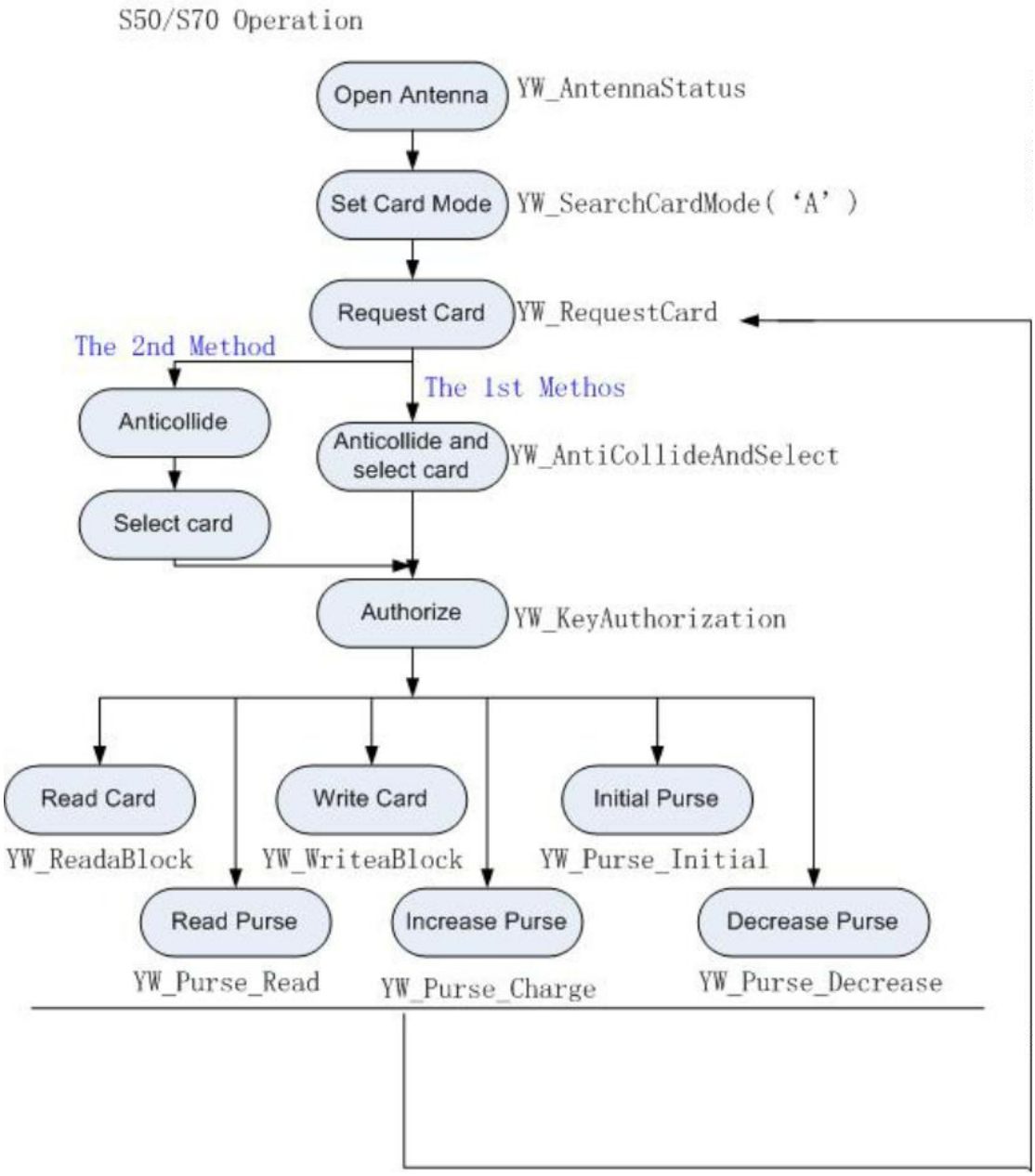
Param:

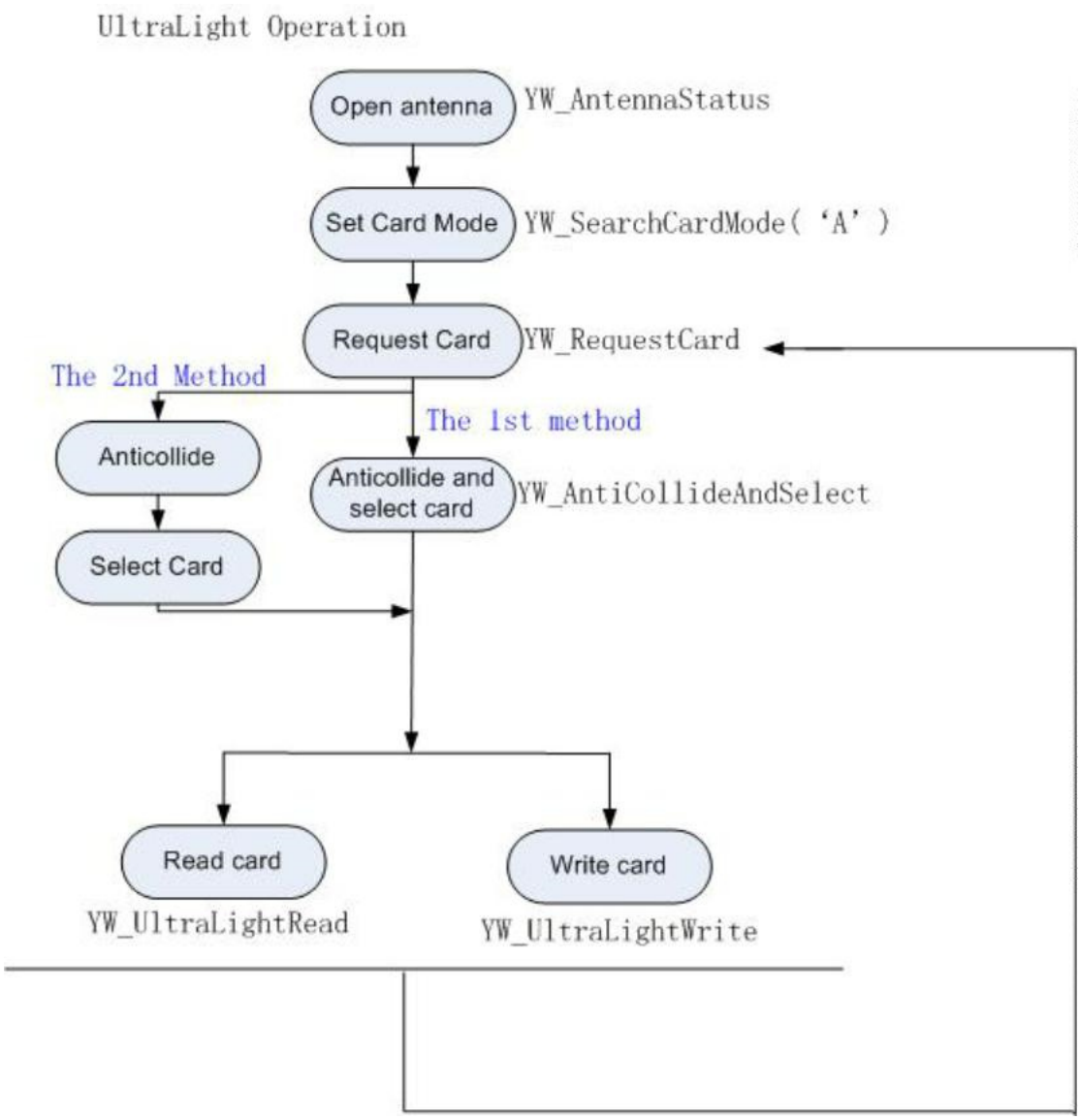
Param	Type	
ReaderID	int	Reader ID

SAMIndex	int	SAM Index
BaudIndex	int	0x00->9600 (Default) 0x01->19200 0x02->38400 0x03->55800 0x04->57600 0x05->115200

Return: 1 success, 0 fail

6 Card operation of S50/S70




UltraLight Operation
Open antenna
YW_AntennaStatus
Set Card Mode
YW_SearchCardMode('A')
Request Card
YW_RequestCard
The 2nd Method
The 1st method
Anticollide
Anticollide and select card
YW_AntiCollideAndSelect
Select Card
Read card
Write card
YW_UltraLightRead
YW_UltraLightWrite

7 Order Infomation

Quick-Ohm Küpper & Co. GmbH

WEB: www.quio-rfid.de

Phone/Fax: 049-0202-404329

Email: kontakt@quio-rfid.de