4.6.5 OMR_SetPrinterMode

Prototype	BOOL OMR_SetPrinterMode(int iMode)	
Process	Use the printer setting command PR to set the print mode.	
Parameter	iMode	Print Mode SR_PRINTER_MODE_AFTER_FEED : Print after feeding SR_PRINTER_MODE_FEED_AND_PRINT : Print while feeding
Return Value	TRUE	Successful
	FALSE	Failure

4.6.6 OMR_GetPrinterMode

Prototype	BOOL OMR_GetPrinterMode(int* piMode)	
Process	Use the printer setting command PR to get the print mode.	
Parameter	piMode	Address storing the print mode
Return Value	TRUE	Successful
	FALSE	Failure

4.6.3 OMR_SetPrintOrder

Prototype	BOOL OMR_SetPrintOrder (int iFirst, int iSecond, int iThird)	
Process	Use the printer setting command PR to set the print order. This setting will clear after printing.	
Parameter	iFirst	Number of the first buffer to print. SR_PRINT_BUFFER_NULL : No printing SR_PRINT_BUFFER_1 : Buffer #1 SR_PRINT_BUFFER_2 : Buffer #2 SR_PRINT_BUFFER_3 : Buffer #3
	iSecond	Number of the second buffer to print.
	iThird	Number of the third buffer to print.
Return Value	TRUE	Successful
	FALSE	Failure

4.6.4 OMR_GetPrintOrder

Prototype	BOOL OMR_GetPrintOrder (int* pFirst, int* pSecond, int* pThird)	
Process	Use the printer setting command PR to get the print mode.	
Parameter	pFirst	Pointer storing the number of the first buffer to print.
	pSecond	Pointer storing the number of the second buffer to print.
	pThird	Pointer storing the number of the third buffer to print.
Return Value	TRUE	Successful
	FALSE	Failure

4.6.7 OMR_SetPrintPosition

Prototype	BOOL OMR_SetPrintPosition (int iPosition)	
Process	Use the printer setting command PR to set the print position.	
Parameter	iPosition	Print Position (Unit: mm) * : For range of value settings, please refer to the appropriate user's manual.
Return Value	TRUE	Successful
	FALSE	Failure

4.6.8 OMR_GetPrintPosition

Prototype	BOOL OMR_GetPrintPosition(int* piPosition)	
Process	Use the printer setting command PR to get the print position.	
Parameter	piMode	Address storing the print position
Return Value	TRUE	Successful
	FALSE	Failure

4.5.5 OMR_GetMachineName

Prototype	BOOL OMR_GetMachineName(CHAR * pResult)	
Process	Use machine name command MN to store the machine name text string to the region defined by the parameter pointer.	
Parameter	*pResult	Storage address of text string for machine name
Return Value	TRUE	Successful
	FALSE	Unsuccessful
Details	Example: CHAR pBuf[16]; OMR_GetMachineName(pBuf); print(pBuf);	

4.5.6 OMR_GetVersion

Prototype	BOOL OMR_GetVersion(int iTarget, CHAR *pResult)	
Process	Use GetVersion command FV to store the machine name text string to the region defined by the parameter pointer.	
Parameter	iTarget	Select the unit for which version information is needed. SR_UNIT_MAIN : Main body SR_UNIT_FRONT : Front sensor unit SR_UNIT_BACK : Back sensor unit SR_UNIT_BARCODE : Barcode unit SR_UNIT_PRINTER : Printer unit SR_UNIT_STACKER : Stacker unit
	*pResult	Version storage address
Return Value	TRUE	Successful
	FALSE	Unsuccessful
Details	Example: Gain main body version data CHAR pBuf[16]; OMR_GetVersion(SR_UNIT_MAIN,pBuf); print(pBuf);	

4.5.4 OMR_GetDeviceInfo

Prototype	DWORD OMR_GetDeviceInfo(void)
Process	Use the GetDeviceInfo command DI to check what unit(s) are connected.
Parameter	None
Return Value	0xXXXXXXXX Successful (see details below)
	SR_FUNCTIONAL_FAIL Failure
Details	When the return value selects a value other than SR_SENSOR_FAIL, DWORD type 32 bit will respond to the on/off data of each bit per sensor. Each bit response is as listed below.

Bit	Content		Mask Constant	Constant Value
Bit31 Bit30 Bit29 Bit28	Stacker unit	0x0: Not Connected 0x1: Connected 0x8: No Cartridge	SR_DEVICE_UNIT_STACKER_MASK SR_DEVICE_UNIT_STACKER SR_DEVICE_UNIT_STACKER_ERR	0xf0000000 0x10000000 0x80000000
Bit27 Bit26 Bit25 Bit24	Printer unit	0x0: Not Connected 0x1: Connected 0x8: Not Cartridge	SR_DEVICE_UNIT_PRINTER_MASK SR_DEVICE_UNIT_PRINTER_ SR_DEVICE_UNIT_PRINTER_ERR	0x0f000000 0x01000000 0x08000000
Bit23 Bit22 Bit21 Bit20	Barcode unit	0x0: Disconnected 0x1: Connected vertical 0x2: Connected horizontal 0x8: Connecting Error	SR_DEVICE_UNIT_BARCODE_MASK SR_DEVICE_UNIT_BARCODE_V SR_DEVICE_UNIT_BARCODE_H SR_DEVICE_UNIT_BARCODE_ERR	0x00f00000 0x00100000 0x00200000 0x00800000
Bit19 Bit18 Bit17 Bit16	Back Sensor unit	0x0: Not Connected 0x1: Connected	SR_DEVICE_UNIT_BACK_MASK SR_DEVICE_UNIT_BACK	0x000f0000 0x00080000
Bit15 : Bit8	(aria out of use)			
Bit7 Bit6 Bit5 Bit4	Sensor type	0x0: Visible red 0x1: Near infra red 0x8: Connecting Error	SR_DEVICE_SENSOR_TYPE_MASK SR_DEVICE_SENSOR_TYPE_ERR	0x000000f0 0x00000080
Bit3 Bit2 Bit1 Bit0	Sensor pitch type	0x1:1/6 inches 0x2: 0.2 inches 0x3: 0.2 inches S 0x4: 0.25 inches 0x5: 0.3 inches 0x6: 0.3 inches F 0x7: 6mm 0x8: 0.2 Inch K 0x9: 0.2 Inch Special 0xA: 0.2 Inch C 0xF: Connecting Error	SR_DEVICE_SENSOR_PITCH_MASK SR_DEVICE_SENSOR_PITCH_ERR	0x0000000f 0x0000000f

Example (Stacker Unit Connection Conformation)

```
DWORD dev_info;
dev_info=OMR_GetDeviceInfo0;
if(dev-info==SR_FUNCTION_FAIL){
   //enter error procedure here
}
if((dev_info&SR_DEVICE_UNIT_STACKER)!=0){
   //stacker unit is connected at this point
}
```