AN969
APPLICATION NOTE

SCI COMMUNICATION BETWEEN ST7 AND PC

by Microcontroller Division Applications

INTRODUCTION

This document presents a standard communications interface between a ST7 microcontroller and a PC. This communication is done through the ST7 SCI peripheral and a serial port of the PC using the RS232 protocol.

1 SCI COMMUNICATION

The main features of the SCI communication peripheral are summarized below. Refer to your ST7 datasheet for more details.

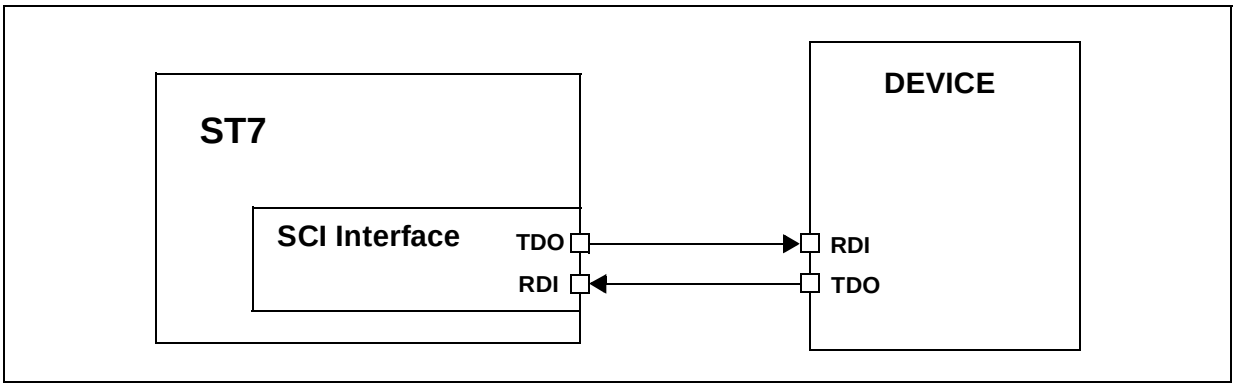
1.1 MAIN FEATURES

The Serial Communication Interface (SCI) offers a flexible means of full-duplex data exchange with external equipment requiring an industry standard NRZ asynchronous serial data format.

The SCI allows a very wide range of baud rates with different baud rates for transmission and reception.

For SCI communication, only two signals are needed, one for transmission and the other for reception. No clock signal is needed as it works in asynchronous mode. Each device has a Transmit Data Output pin (TDO pin) and a Receive Data Input pin (RDI pin). See Figure 1.

Figure 1. ST7 and SCI interface set-up



The user must be very careful in identifying the use of each pin. This can easily be done putting the device in transmission and checking with an oscilloscope if a transmission frame is present or not.

1.2 BAUD RATES

Transmission and reception can be driven by their own baud rate generator. However be aware that to communicate correctly, the receiver must have a reception baud rate strictly equal to the transmission baud rate of the transmitter. **If not, the communication will be corrupted**. As long as this condition is met, a wide range of baud rates is possible.

1.3 FRAMES

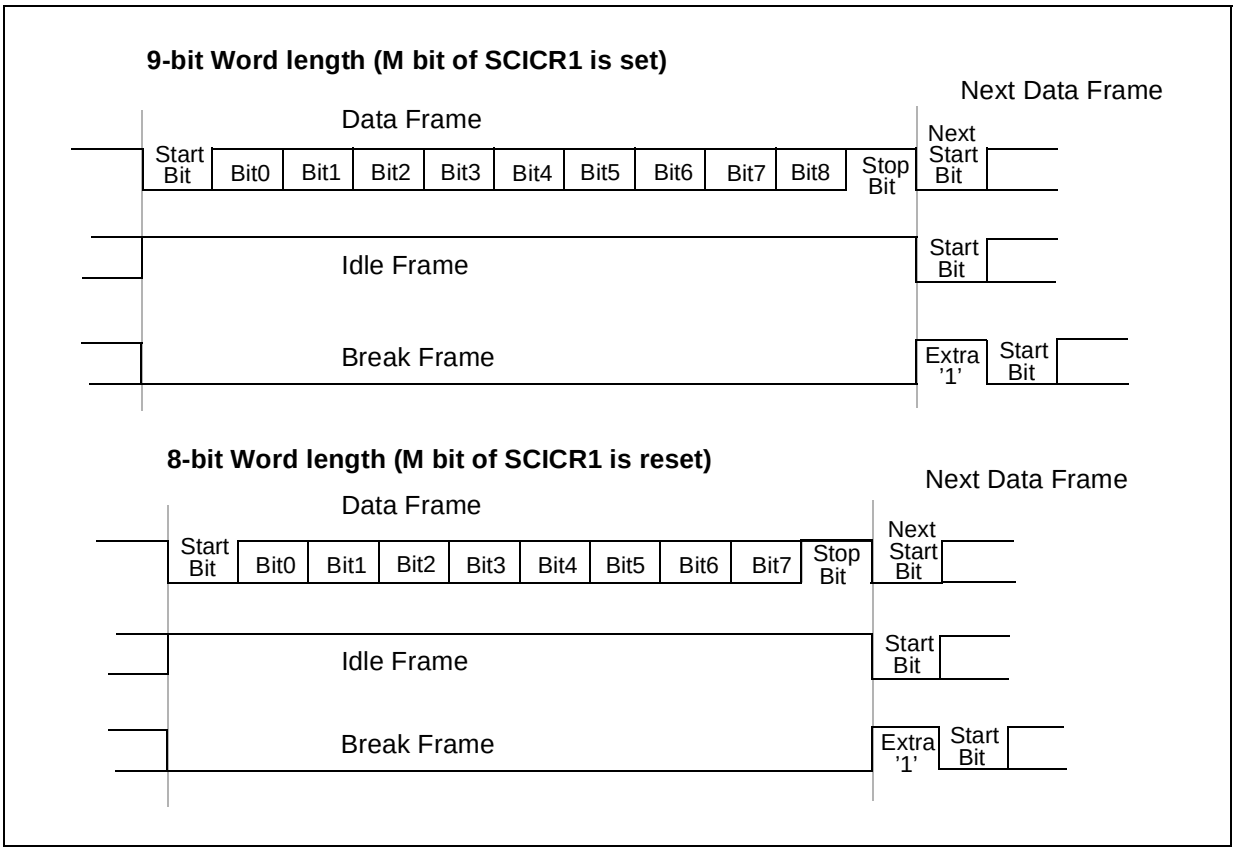
Any transmission is Least Significant Bit first. A data word can be 8 or 9 bits long. A data frame begins with a «start bit», which is a ’0’ bit and ends with a «stop bit», which is a ’1’ bit. See Figure 2.

Two special frames are also managed :

An Idle character is interpreted as an entire frame of ’1’ bits followed by the start bit of the next frame which contains data.

A Break character is detected when receiving ’0’ bits for some multiple of the frame period. At the end of the last break, the transmitter inserts an extra ’1’ bit to acknowledge the next start bit.

Figure 2. Frames and word length



2 RS232 COMMUNICATION WITH A PC

2.1 MAIN FEATURES

The electrical and protocol characteristics of RS232 are different from those of the ST7 SCI peripheral. In RS232 communication, high level is typically +7V and low level is typically -7V, while the SCI peripheral works with TTL levels (0, +5V).

Furthermore, the polarities are different. A ’1’ bit coming from the SCI corresponds to a ’0’ bit in RS232, and a ’0’ bit to a ’1’ bit. **It is true for all bits including the start and stop bit**.

So it is necessary to implement a conversion between the PC and the SCI peripheral. In this application a MAX232 is used for this purpose.

2.2 PC CONFIGURATION

The PC will be used as a terminal. The description below refers to the Windows 98 environment. This terminal application is called hyperterminal.

Under Windows, open the «hyperterminal» application. To configure it, go to the communication parameters menu. The options of this window must be the same as the ones defined with your ST7.

After selecting the right serial communication port, select the same baud rate as the one configured in the micro. As the PC accepts only one baud rate, transmission and reception baud rates will have the same value. Data word length will be 8 bits, but you can choose to use 1 or 2 stop bits. «Flow control» can be either Xon/Xoff or none (the ST72F264.ht file is also provided in the corresponding zip file).

The PC is then correctly configured.

3 ST72F264 CONFIGURATION

This application was implemented with a ST72F264. This microcontroller uses a 16 MHz external clock (quartz). A description of the baud rate selection is given later in this application note.

3.1 GENERAL INITIALIZATION

The ST72F264 internal clock (f_{CPU}) works at 16MHz/2 = 8MHz.

Two pins of the ST7 are used :

- the TDO pin (Transmit Data Output, alternate function of PA7).
- the RDI pin (Receive Data Input, alternate function of PA5).

The PA7, PA5 pins must be initialized as floating inputs (please refer to the datasheet), the value of the Port A Data Register does not matter.

3.2 SCI CONFIGURATION

The peripheral is configured using the 2 SCI Control Registers (SCICR1, SCICR2), and the Baud Rate Register (BRR) or the 2 extended baud rate registers of the ST72F264 (ETPR, ERPR).

As in this application a polling strategy is used, SCI interruptions are disabled (TIE, TCIE, RIE, ILIE bits of SCICR2 register are reset).

3.2.1 Resetting control registers

First, the reset status is forced to the control registers and pending interrupts are cleared.

3.2.2 Baud rates selection

Then the baud rates must be selected. We should always have the same baud rates in the PC and in the SCI.

In this application, the Extended Transmit Prescaler division Register (ETPR) and the Extended Receive Prescaler division Register (ERPR) are not used and so are in reset state. Therefore the baud rates are calculated as follows:

$$TX = f_{CPU}/(16*PR*TR) \qquad RX = f_{CPU}/(16*PR*RR)$$

where PR = 1, 3, 4 or 13 (SCP0 and SCP1 bits in the BRR register), TR = 1, 2, 4, 8, 16, 32, 64 or 128 (SCT0, SCT1 and SCT2 bits of the BBR register) and RR = 1, 2, 4, 8, 16, 32, 64 or 128 (SCR0, SCR1 and SCR2 bits of the BRR register).

For example, in our application, if we want 9600 as baud rate, we have to choose PR = 13 and TR = RR = 4.

Note: ETPR and ERPR can be used to finetune the baud rate. When ETPR and ERPR contain any value other than 0, the BRR register is disabled and the baud rates are given by the formula:

$$TX = f_{CPU}/(16*EPTR*PR*TR)\ RX = f_{CPU}/(16*ERPR*PR*RR)$$

where EPTR and ERPR = 1 to 255.

3.2.3 Word length

The word length is selected by the M bit in the SCICR1 register.

7							0
R8	T8	SCID	M	WAKE	PCE	PS	PIE

In the application, data length is always 8 bits, but the user can force a second stop bit. This is done using a 9-bit word length and setting the 9^{th} bit (T8 bit of SCICR1). When receiving data, R8 bit will always be set (reception of 2 stop bits).

Special bits (PCE,PS,PIE) are also available to use the SCI with parity (not used in this application). Refer to the datasheet for more information.

3.2.4 Selecting reception and transmission modes

The selection is made using the SCICR2 register.

7							0
TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK

Transmission is enabled or disabled depending on the value of the TE bit of SCICR2 (TE = 1, transmission enabled). When disabled, the TDO pin is free for general purpose I/O.

Reception is enabled or disabled depending on the value of the RE bit of SCICR2 (RE = 1, reception enabled). When disabled, the RDI pin is is free for general purpose I/O.

When reception is disabled, no data must be sent to the ST7. If data are to be received at this time, the RE bit must be set. Otherwise a wrong byte might be detected when going back into reception mode.

4 SCI COMMUNICATION BETWEEN ST7 AND PC VIA RS232

The software for this application is the SCI driver. The complete software can be found in the the ST internet website. It is of course only an example. It is up to the user to adapt it to his specific application.

4.1 GENERAL DESCRIPTION

In this application, a ST72F264 is connected to a PC. The communication is performed using the «hyperterminal» application of Windows 98.

Characters are sent by the PC in ASCII code to the ST7. Every 4 characters received by the ST7 are sent back to the PC and can be read on its screen.

The software is a polling SCI communication between an ST72F264 and a PC via RS232. The communication works at 9600bps and has 1 stop bit.

The first part of the program performs the initialization of the ST72F264, core and SCI peripheral.

Then it enters transmission mode, sends the content of a 4-byte buffer to the PC. The initialized value of the buffer is «test» in ASCII code. When the 4 characters are sent, transmission mode is ended and the SCI peripheral enters reception mode.

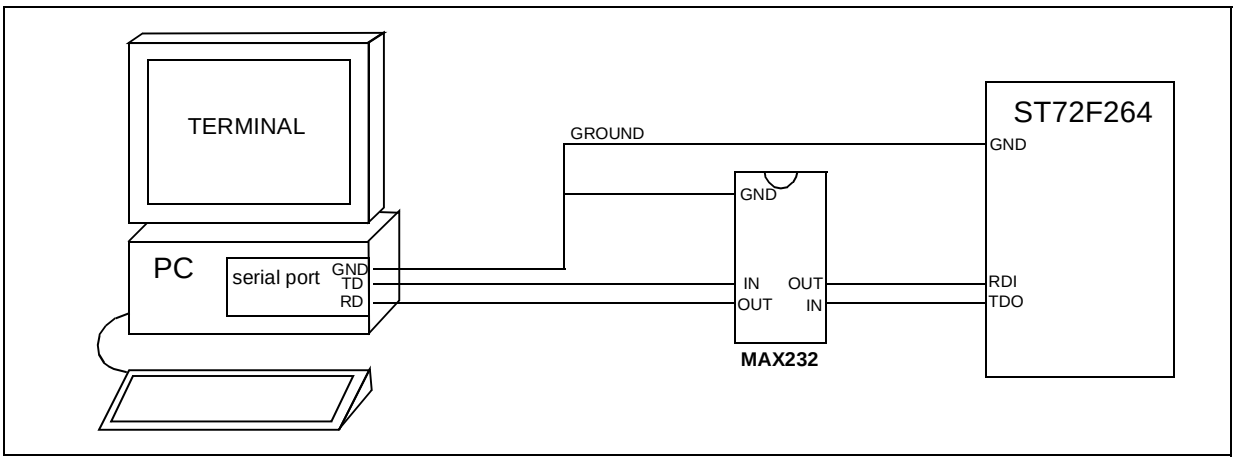
In reception mode, the software waits for the user to send 4 ASCII characters from the PC. Each character is put in the buffer until the fourth is received. Then reception mode is ended, and goes back to transmission mode which will send the new contents of the buffer and so on.

As the ST7 sends the same characters as received, it is very easy to check if the communication is correct. **Note that this program does not manage communication errors.**

4.2 HARDWARE

The ST7 SCI peripheral cannot be directly connected to a PC, as it uses the RS232 protocol. The conversion between SCI and RS232 can be done using a MAX232. An overview schematic is presented below (Figure 3.).

Figure 3. Overview Schematic



Be sure that the three main devices (PC, ST7, MAX232) have the same electrical reference (GND).

The Receive Data pin (RD) of the serial port of the PC must correspond to the TDO pin of the ST7, and the Transmit Data pin (TD) to the RDI pin.

4.3 FLOWCHARTS

The flowcharts of the application are presented below (Figure 4., Figure 5.).

Figure 4. Initialization

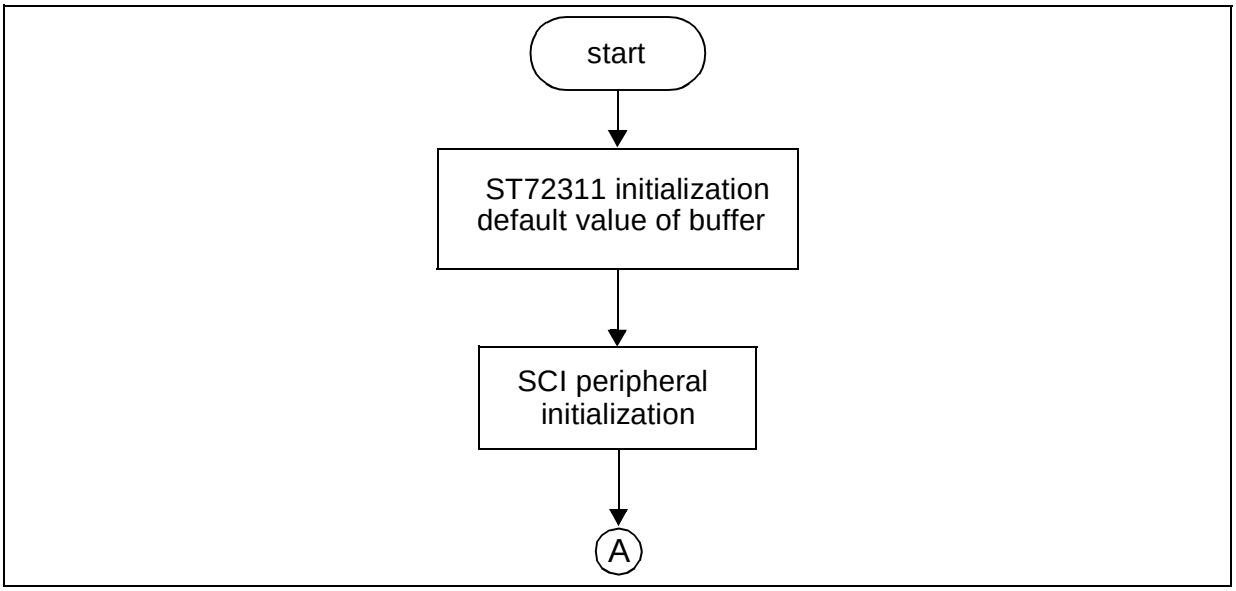
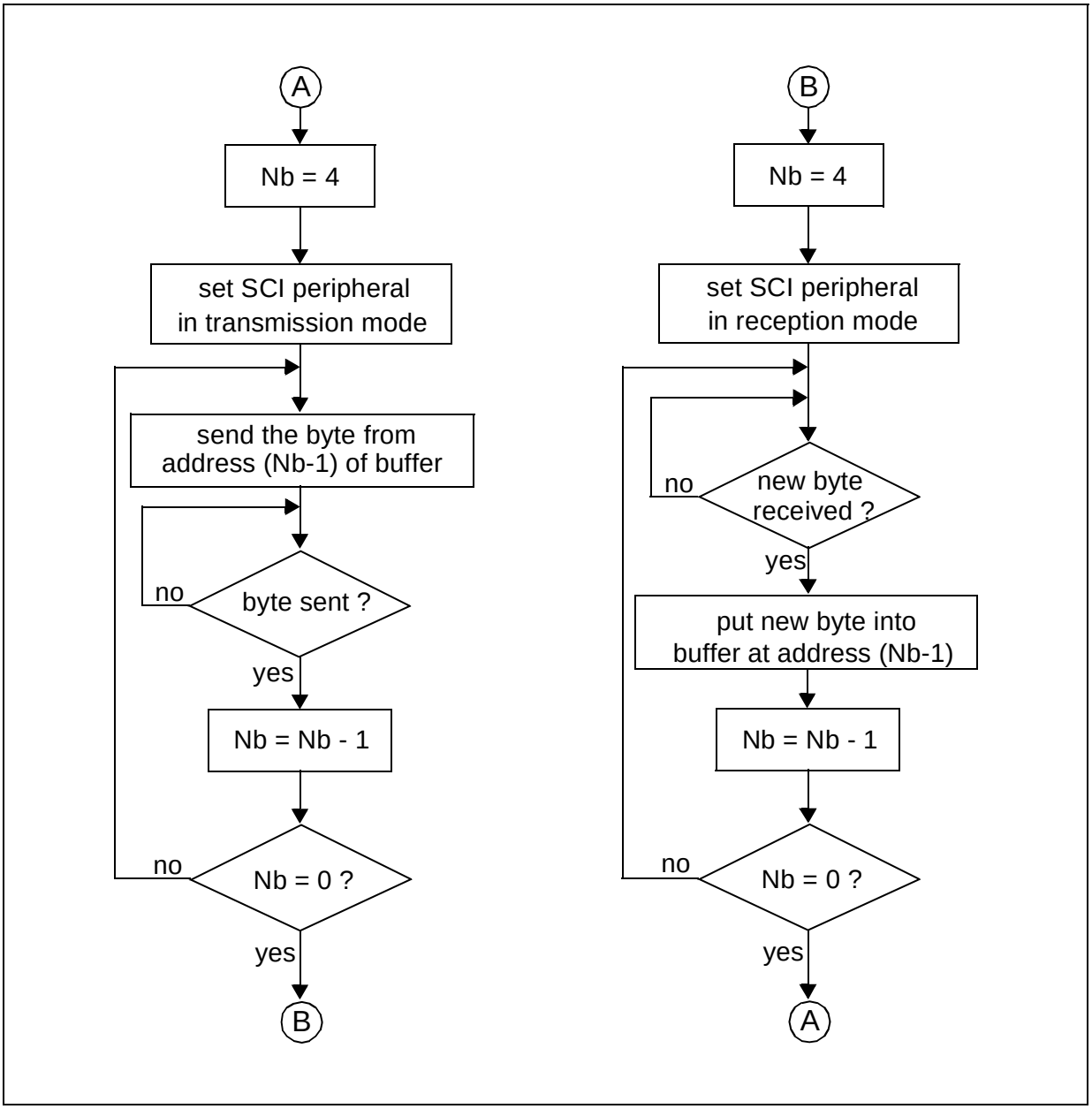


Figure 5. Transmission mode, Reception mode



4.4 SOFTWARE

All the source files in assembly code are given in the zip file with application note.

The source files are for guidance only. STMicroelectronics shall not be held liable for any direct, indirect or consequential damages with respect to any claims arising from use of this software.

“THE PRESENT NOTE WHICH IS FOR GUIDANCE ONLY AIMS AT PROVIDING CUSTOMERS WITH INFORMATION REGARDING THEIR PRODUCTS IN ORDER FOR THEM TO SAVE TIME. AS A RESULT, STMICROELECTRONICS SHALL NOT BE HELD LIABLE FOR ANY DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WITH RESPECT TO ANY CLAIMS ARISING FROM THE CONTENT OF SUCH A NOTE AND/OR THE USE MADE BY CUSTOMERS OF THE INFORMATION CONTAINED HEREIN IN CONNEXION WITH THEIR PRODUCTS.”

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics

©2002 STMicroelectronics - All Rights Reserved.

Purchase of I^2C Components by STMicroelectronics conveys a license under the Philips I^2C Patent. Rights to use these components in an I^2C system is granted provided that the system conforms to the I^2C Standard Specification as defined by Philips.

STMicroelectronics Group of Companies
Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan
Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - U.S.A.

http://www.st.com