SGS-THOMSON
MICROELECTRONICS

APPLICATION NOTE

HOW TO USE THE DUAL PORT RAM FOR PARALLEL DATA TRANSFER WITH THE ST75C502

By Laurent CLARAMOND

CONTENTS

Page

I **INTRODUCTION**. 1
I.1 OVERALL 1
I.2 CAUTIONS 1
I.3 NOTATIONS 1

II **PARALLEL DATA EXCHANGE**. 2
II.1 OVERVIEW. 2
II.2 SELECT PARALLEL MODE 3
II.3 TRANSMIT BUFFERS 3
II.4 RECEIVE BUFFERS. 3
II.5 INTERRUPTION 4
II.6 TRANSMITTING IN PARALLEL MODE 5
II.6.1 Description 5
II.6.1.1 XMIT Command 5
II.6.1.2 STOP Command. 5
II.6.1.3 Timing 6
II.6.1.4 FSK Full Duplex Mode 6
II.6.2 Modem Flow Chart 6
II.6.3 Host Flow Chart 7
II.6.4 Error Detection 8
II.6.5 Data Transmission 8
II.6.5.1 Description 8
II.6.5.2 Status Word Format 8
II.7 RECEIVING IN PARALLEL MODE. 8
II.7.1 Description 8
II.7.1.1 Initialization 8
II.7.1.2 Loss of Carrier. 8
II.7.1.3 FSK Synchronization. 8
II.7.2 Modem Flow Chart 9
II.7.3 Host Flow Chart 9
II.7.4 Error Detection 10
II.7.5 Data Reception 10
II.7.5.1 Description 10
II.7.5.2 Status Word Format 10

I - INTRODUCTION

I.1 - Overall

This application note describes the way to use the ST75C502 Data Buffers.

These Data Buffers are implemented into the DUAL RAM of the ST75C502. They are shared between the Host processor and the ST75C502. A mechanism of Flags and interrupts is associated with them to allow an easy management of the Data.

I.2 - Cautions

The mechanism described below is only valid while in regular Data Transmission, not in Handshaking neither in Call progress (or DTMF) tone detection modes.

I.3 - Notations

Any ***bold_italic*** command refers to reserved Nam.
"Host" refers to the Micro-controller connected to the ST75C502 Data Pump.
"Transmit Data" (or Tx) means Data transfered by the ST75C502, via the modulator, to the telephone line, and "Receive Data" (or Rx) means Data coming from the telephone line and demodulated by the ST75C502.

II - PARALLEL DATA EXCHANGE

II.1 - Overview

While in Parallel Mode, the transmitted (respectively Received) Data to (from) the telephone line are exchanged between the Host and the ST75C502.

Two totally independent channels are provived for Transmit and Receive Data. Even while using Half Duplex modes of operation, the transmitted data comes from the Transmit buffers and the receive Data arrives in the Receive buffers.

Two independent Interrupts, ***IT2*** (for Transmit) and ***IT3*** (for Receive) are available for synchronizing the ST75C502 and the Host. An additional ***IT0*** interrupt will signal the errors in the synchronization mechanism.

The equivalent Data Flow is shown in Figure 1.

II.2 - Select Parallel Mode

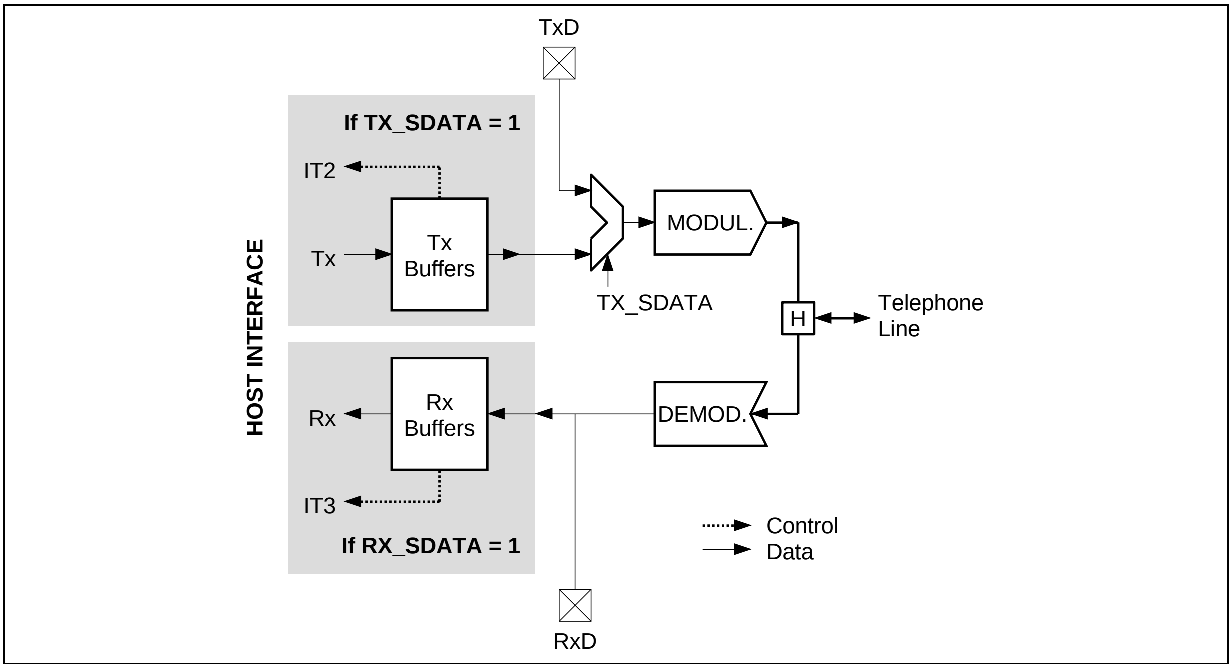
The ***SERIAL*** command allows the independent selection of the parallel mode for the Transmit and/or Receive Data path. The parameter syntax is as follows :

Field	Byte	Pos.	Value	Definition
TX_SDATA	1	0	0* 1	Use Serial Link for Tx Data Use Parallel Link for Tx Data
RX_SDATA	1	1	0* 1	Use only Serial Link for Rx Data Use both Serial an Parallel Link for Rx Data

Note : Even if the parallel mode is selected for the Receiver, the Received Bit Stream is available on the RxD pin of the ST75C502.

This command must be sent in Data Mode, when the Transmit or Receive data links are established. This corresponds with the presence of the signals ***STA_106*** (for Tx) and/or ***STA_109*** (for Rx).

Figure 1



AN684-01.EPS

II - PARALLEL DATA EXCHANGE (continued)

II.3 - Transmit Buffers

Two identical buffers are provided to exchange the data between the Host interface and the ST75C502. When the Host is writing data into a buffer, the ST75C502 is transmitting the other one. After that, both the Host and the ST75C502 switch to use the other buffer. This mechanism, called "Double-Buffering", ensures that the host has the maximum time to fill one buffer. The DUAL RAM area associated with the transmit buffers is as follows :

Name	Address	Description
DTTBS0	$2E	Buffer 0 Status Byte
DTTBF0[0]	$2F	Buffer 0 Data Byte 0
DTTBF0[1]	$30	Buffer 0 Data Byte 1
DTTBF0[2]	$31	Buffer 0 Data Byte 2
DTTBF0[3]	$32	Buffer 0 Data Byte 3
DTTBF0[4]	$33	Buffer 0 Data Byte 4
DTTBF0[5]	$34	Buffer 0 Data Byte 5
DTTBF0[6]	$35	Buffer 0 Data Byte 6
DTTBF0[7]	$36	Buffer 0 Data Byte 7
DTTBS1	$37	Buffer 1 Status Byte
DTTBF1[0]	$38	Buffer 1 Data Byte 0
DTTBF1[1]	$39	Buffer 1 Data Byte 1
DTTBF1[2]	$3A	Buffer 1 Data Byte 2
DTTBF1[3]	$3B	Buffer 1 Data Byte 3
DTTBF1[4]	$3C	Buffer 1 Data Byte 4
DTTBF1[5]	$3D	Buffer 1 Data Byte 5
DTTBF1[6]	$3E	Buffer 1 Data Byte 6
DTTBF1[7]	$3F	Buffer 1 Data Byte 7

The Bit 0 (LSB) of the Buffer 0 Data Byte 0 is the first in time to be transmited.

A value of 0 in the Status Byte of the Buffer signals to the Host that a buffer is empty. This value is set by the ST75C502 each time it has emptied the buffer. After having used one buffer, the host must select the other buffer for the next operation. The host must start with the Buffer 0 as soon as the ***ST_106*** signal is on and the ***SERIAL Tx*** is selected and BEFORE the ***XMIT 1*** command is sent.

A mechanism of interruption (***IT2*** for Transmit) is associated with the Data Buffer management. Each time a Buffer is emptied by the ST75C502 it generates an interrupt.

II.4 - Receive Buffers

Symetrically two identical buffers are provided to exchange receive data between the ST75C502 and the Host processor. While the ST75C502 is filling one of the buffers with the receive bits, the Host processor is reading the other buffer. As soon as the host has emptied a buffer it frees it by writing 0 in the Buffer Status Byte.

The DUAL RAM area associated with the receive buffers is as follows :

Name	Address	Description
DTRBS0	$1C	Buffer 0 Status Byte
DTRBF0[0]	$1D	Buffer 0 Data Byte 0
DTRBF0[1]	$1E	Buffer 0 Data Byte 1
DTRBF0[2]	$1F	Buffer 0 Data Byte 2
DTRBF0[3]	$20	Buffer 0 Data Byte 3
DTRBF0[4]	$21	Buffer 0 Data Byte 4
DTRBF0[5]	$22	Buffer 0 Data Byte 5
DTRBF0[6]	$23	Buffer 0 Data Byte 6
DTRBF0[7]	$24	Buffer 0 Data Byte 7
DTRBS1	$25	Buffer 1 Status Byte
DTRBF1[0]	$26	Buffer 1 Data Byte 0
DTRBF1[1]	$27	Buffer 1 Data Byte 1
DTRBF1[2]	$28	Buffer 1 Data Byte 2
DTRBF1[3]	$29	Buffer 1 Data Byte 3
DTRBF1[4]	$2A	Buffer 1 Data Byte 4
DTRBF1[5]	$2B	Buffer 1 Data Byte 5
DTRBF1[6]	$2C	Buffer 1 Data Byte 6
DTRBF1[7]	$2D	Buffer 1 Data Byte 7

The Bit 0 (LSB) of the Buffer 0 Data Byte 0 is the first received bit in time (the oldest).

A value of 0 in the Status byte of the Buffer signals to the ST75C502 that a buffer is empty. This value is set by the Host each time it has emptied the buffer. After having used one buffer, the host must select the other buffer for the next operation. The Host must start with the Buffer 0 as soon as the ***STA_109*** signal is on and the ***SERIAL Rx*** is selected.

A mechanism of interruption (***IT3*** for Receive) is associated with the Data Buffer management. Each time a buffer is filled by the ST75C502 it generates an interrupt.

II - PARALLEL DATA EXCHANGE (continued)

II.5 - Interruption

Two Interrupt signals are provided in order to synchronize the Data Buffer Exchanges. ***IT2*** is associated with the Transmit Buffer mechanism and ***IT3*** with the Receive Buffer mechanism.

In order to enable these interrupts, the Host processor must set the bit 2 (for ***IT2***) and the bit 3 (for ***IT3***) of the ***ITMASK*** Register to 1. It must also set the Bit 7 of the ***ITMASK*** register to 1 in order to globally enable all the selected sources of interruption.

When an Interrupt occurs (low level on $\overline{SINTR}$ pin) the user must read the ***ITSRCR*** Register to determine the source of the interrupt, either ***IT2*** for Tx (if the bit 2 is 1) or ***IT3*** for Rx (if the bit 3 is 1).

Once the Interrupt has been serviced, the host must acknowledge it by writing a $00 value into the register ***ITRES2*** for ***IT2***, or ***ITRES3*** for ***IT3***.

These registers have the following address :

Name	Address	Type	Description
ITRES2	$42	Write Only	Clear IT2
ITRES3	$43	Write Only	Clear IT3
ITMASK	$4F	Read/Write	Interrupt Mask
ITSRCR	$50	Read Only	Interrupt Source

Notes : 1. The ST75C502 does not check that the interrupt has been acknowledged.

2. Even if the Host does not use the interruption, the ST75C502 will set the bit 2 (for ***IT2***) and/or bit 3 (for ***IT3***) of the ***ITSRCR***.

3. The ST75C502 uses only the Data Buffer Status Bytes to detect Overrun or Underrun Error. These errors are reported into the ***SYSERR*** byte, and could generate an interrupt ***IT0***.

The equivalent schematic is shown in Figure 2.

The interrupt mechanism assumes that the Host processor uses a Level sensitive interrupt (active low). The Flow chart of the Host interrupt service routine looks generaly like Figure 3.

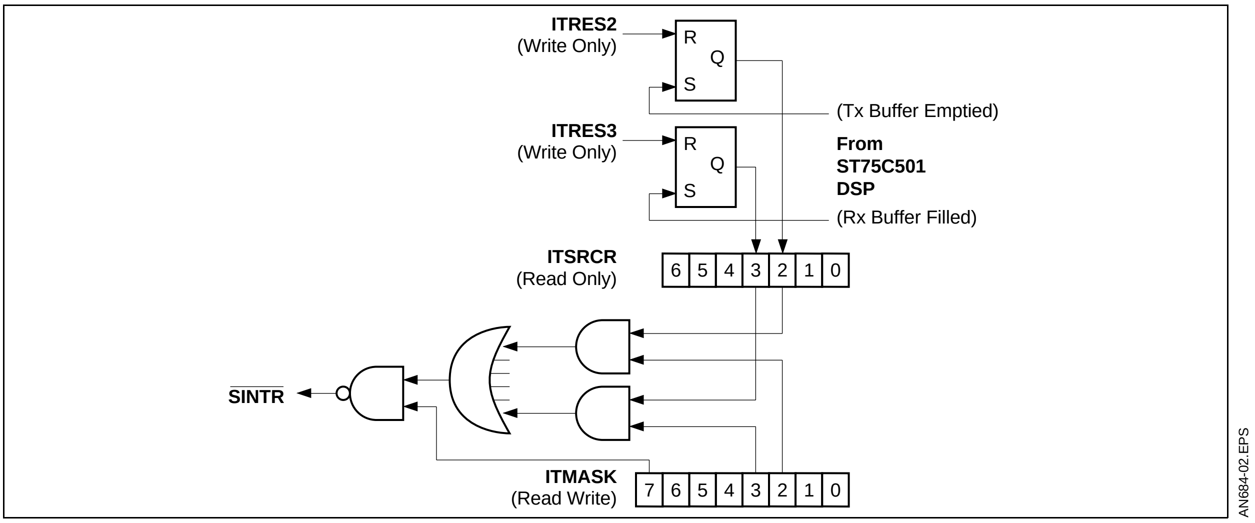
The transmitter reads the bits in the DUAL RAM Buffer ***DTTBFx*** (starting with the Bit 0 of Byte 0 of Buffer 0) and sends them over the telephone line. The Buffer Status Byte ***DTTBSx*** contains the number of Data Bytes to transmit.

The receiver write the received bits comming from the Telephone line and write them into the DUAL RAM Buffer ***DTRBFx*** (starting with the Bit 0 of the Byte 0 of the Buffer 0). The Buffer Status Byte ***DTRBSx*** contains the number of Data Bytes received (generaly 8).

The time between each ***IT2*** interrupts (or ***IT3***) is equal to 64 bit if the number of Data Bytes is set to 8. The Host has the full 64 bits time to serve the interrupt :

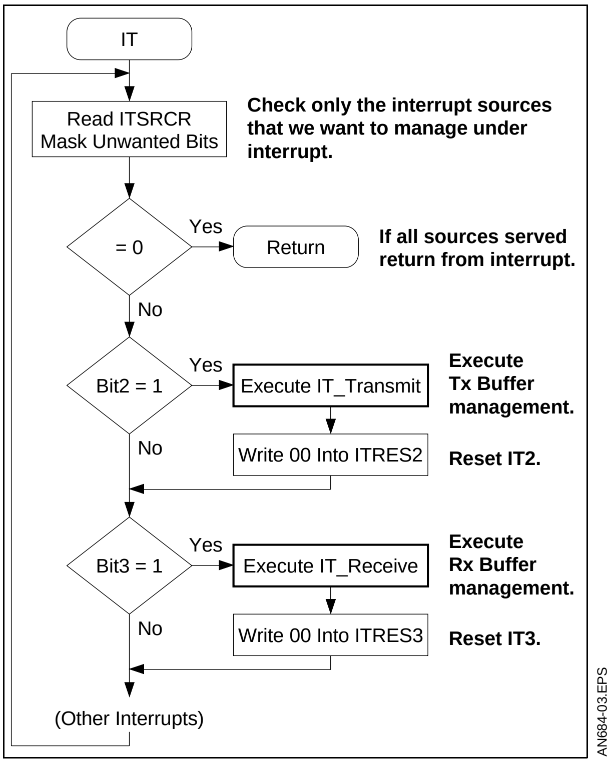
	Bit Rate (bps)	Interrupt Time (ms)
Modem	14400	4.4
	12000	5.3
	9600	6.6
	7200	8.8
	4800	13.3
	2400	26.6
	1200	53.3
	300	213.3
	75	853.3
Voice	A-Law	1.1
ADPCM	28.8k	2.2
	21.6k	2.9
	14.4k	4.4

Figure 2



II - PARALLEL DATA EXCHANGE (continued)

Figure 3



II.6 - Transmitting in Parallel Mode

II.6.1 - Description

When the ***STA_106*** (CTS) signal is on, the user must select the parallel mode by enabling the parallel link with ***TX_SDATA*** bit set in ***SERIAL*** command. After that the ST75C502 will start transmitting continuous "1". Note that for a proper operation each time the ***STA_106*** signal goes on, the ***SERIAL*** command must be sent.

II.6.1.1 - XMIT Command

The ***XMIT*** Command works like a CTS signal for the Parallel process.

When ***XMIT*** is off, the ST75C502 transmits continuous "1". When on the ST75C502 transmits Data and starts to manage the Data Buffer.

This command can be sent at any time, while in Data Mode.

Field	Byte	Pos.	Value	Definition
TX_START	1	0	0* 1	(OFF) Send continuous "1" [(1)] (ON) Send data according

Note : 1. The XMIT Off command takes effect only when the two Transmit buffers are empty : ***DTTBF0*** and ***DTTBF1*** equal to $00.

II.6.1.2 - STOP Command

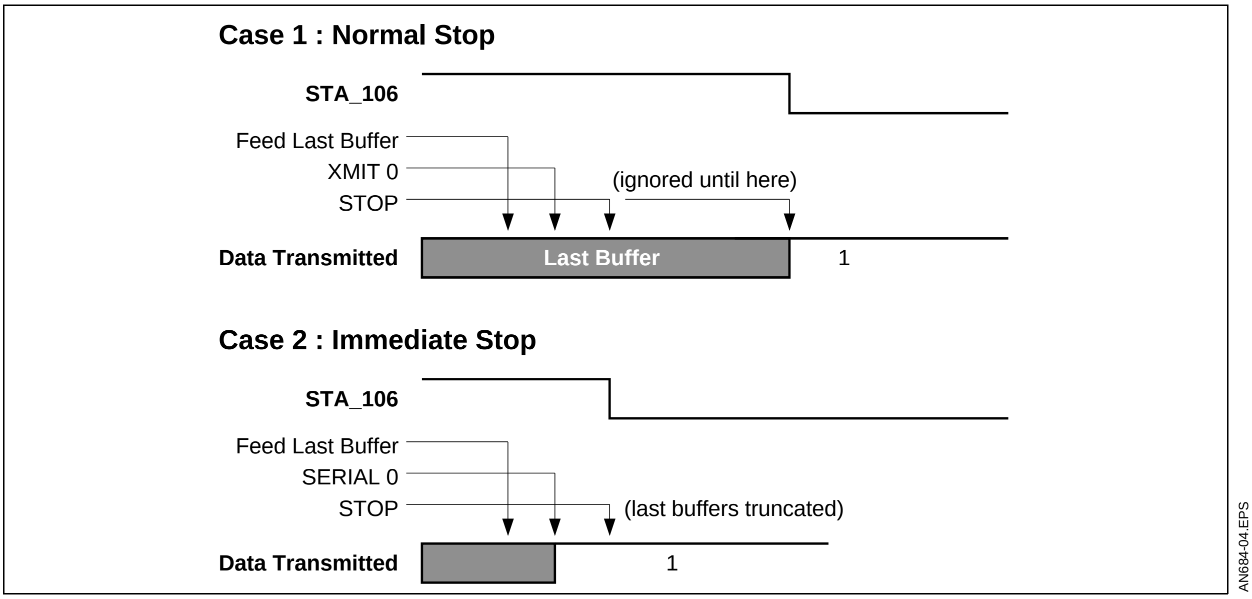
The ***STOP*** command is used in FAX Mode, at the end of the transmission, to stop sending the carrier on the telephone line.

Prior to the ***STOP*** command the user must have stop the parallel transmission with a ***XMIT off*** command.

II.6.1.3 - Timing

Here are regular sequences to stop properly the transmission (See Figure 4).

Figure 4



II - PARALLEL DATA EXCHANGE (continued)

II.6.1.4 - FSK Full Duplex Mode

In FSK Full Duplex Mode the Parallel Mode assumes that the bit time duration is the nominal bit rate. Each bit element from the Transmit buffer is maintained during the full bit time. The nominal bit clock is defined as follows :

FSK Standard	Nominal Transmit Bit Rate (Hz)[1]	Bit Clock on Rx Pin (Hz)	Bit Clock on Tx Pin (Hz)
V.21	300	9600	7200
Bell 103	300	9600	7200
V.23 Originate	75	9600	7200
V.23 Answer	1200	9600	7200

Note : 1. The accuracy of the Bit clock is given by the ST75C502 oscillator, and must better than 50ppm.

II.6.2 - Modem Flow Chart

When in the Parrallel Data Mode, each time the ST75C502 need a bit to transmit it executes the following routine (See Figure 5).

Where x starts with the value 0 and toggle thereafter between 1 and 0.

II.6.3 - Host Flow Chart

Here after are Flowcharts to :

- Establish a V.29 transmission.
- Send synchronous continuous "$AA, $55, $AA, $55, ..." sequence. The management of the Buffers are done under interrupt.
- Stop properly the transmission.

Establish a V.29 transmission and send the very first Buffer (see Figure 6).

Figure 5

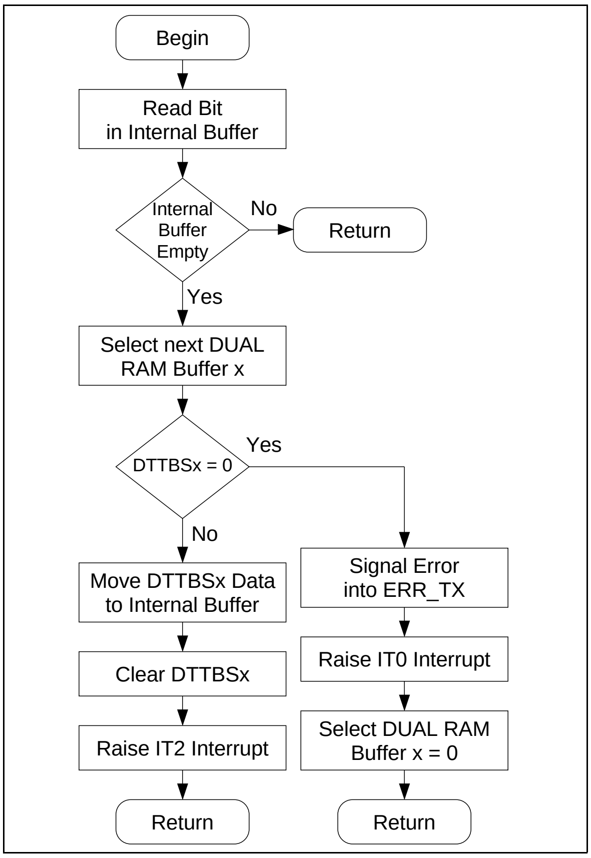
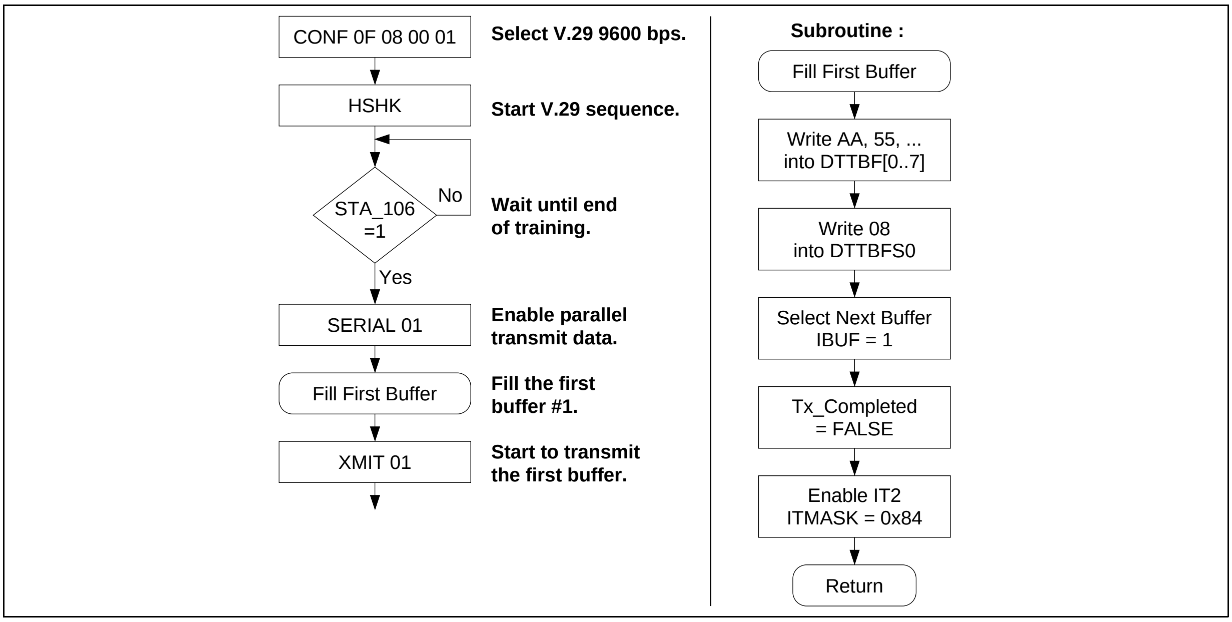


Figure 6



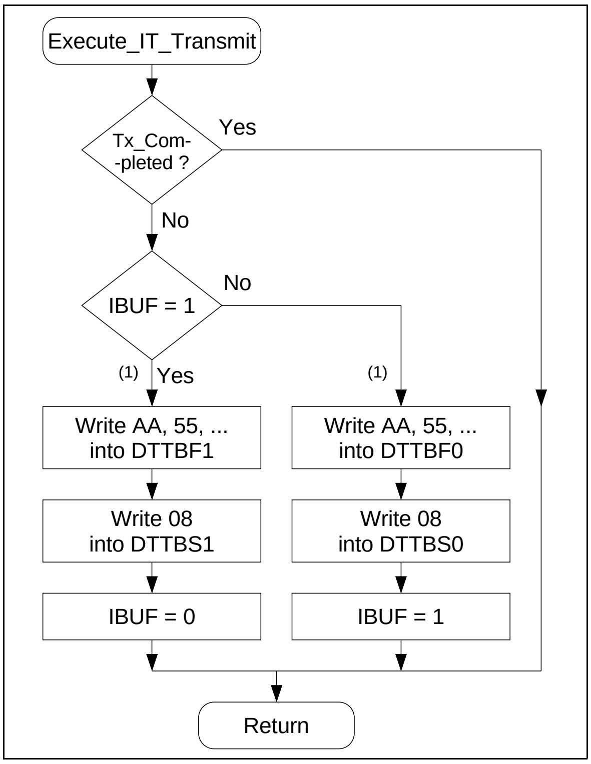
II - PARALLEL DATA EXCHANGE (continued)

These flowcharts show two CPU variables labeled ***IBUF*** and ***Tx_Completed***, they are necessary for the understanding of the mechanism, but there is different manners to implement it. These two variables have the following meanning :

- ***IBUF*** : This is the number of the DUAL RAM Buffer currently in use by the Host processor. It starts with 0 and then alternate 1, 0, 1, 0, ...
- ***Tx_Completed*** : This is a Flag to dialog with the interrupt process in order to stop properly the transmission.

The other Buffers are sent under interrupt control (refer to the interrupt flow chart, Figure 7).

Figure 7

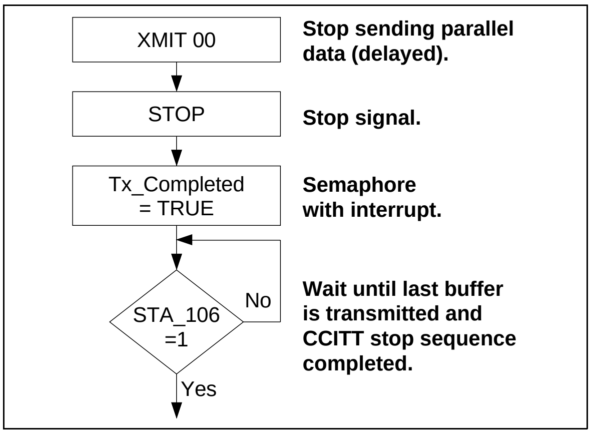


AN684-07.EPS

Note : 1. At this step the host can check that the corresponding ***DTTBSx*** buffer is empty (equal to $00), otherwise it is an error.

To stop properly the transmission, without loss of Data (See Figure 8).

Figure 8



AN684-08.EPS

II.6.4 - Error Detection

Error occurs when the ST75C502 need some bits from the transmit buffer ***DTTBSx*** and this buffer is empty. This condition is called "Underflow".

This error is signaled in the bit ***ERR_TX*** of the ***SYSERR*** byte, and generates an interrupt ***IT0***. To clear the error a ***CSE 01*** command must be issued.

An Underflow condition occurs when the host processor "forgets" to feed the current ***DTTBSx*** buffer.

When an Underflow condition occur the Host must restart the whole parallel initialization, as explained above.

II.6.5 - Data Transmission

II.6.5.1 - Description

The ST75C502 transmits the bits contained in the DUAL RAM Buffer without any modification. It starts with the Bit 0 of the ***DTTBF0[0]*** byte.

II.6.5.2 - Status Word Format

The Transmit Status Bytes ***DTTBS0*** or ***DTTBS1*** have the same meaning (See Table 1).

This status byte must be written by the Host, after filling the corresponding data buffer ***DTTBFx[0..7]*** with the right number of data bytes to transmit.

This status byte is cleared by the ST75C502, just before generating the ***IT2*** interrupt.

Table 1 : DTTBSx in Synchronous Mode

Field	Pos.	Value	Definition
BUFF_LENG	3..0	0 1 2 ... 8 Other	Buffer empty 1 Byte to transmit (***DTTBFx[0]***) 2 Bytes to transmit (***DTTBFx[0]*** and ***DTTBFx[1]***) ... 8 Bytes to transmit (***DTTBFx[0..7]***) Not allowed
Other	7..4	0	Reserved, must be 0

AN684-01.TBL

II - PARALLEL DATA EXCHANGE (continued)

II.7 - Receiving in Parallel Mode

II.7.1 - Description

When the ***STA_109*** (CD) signal goes on, the user must select the parallel mode by enabling the parallel link with ***RX_SDATA*** set in the ***SERIAL RX*** command. After that the ST75C502 will write received data into the DUAL RAM buffer ***DTRBS0***. Note that for a proper operation, each time the ***STA_109*** goes on the ***SERIAL*** command must be send.

II.7.1.1 - Initialization

The host processor must enable the ***IT3*** receive interrupt first.

Then it must empty the two ***DTRBS0*** and ***DTRBS1*** registers by writting $00 at these locations.

Then it must send the ***SERIAL RX*** command.

As soon as the first ***IT3*** interrupt appears, the host must proceed with the ***DTRBS0*** buffer.

II.7.1.2 - Loss of Carrier

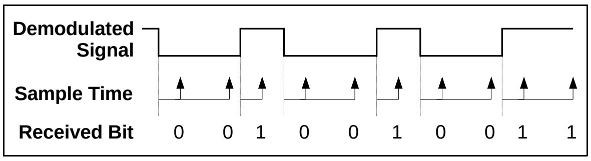
Each time a loss of carrier appears the ST75C502 stops updating the data buffer.

If the carrier reappers the host must proceed again with the INITIALIZATION sequence.

II.7.1.3 - FSK Synchronization

The FSK Full Duplex demodulator uses an algorithm based on the transitions of the received signal. The synchronization mechanism is adjusted with each signal transition in order to sample the demodulated signal at the midle of the bit (see Figure 9).

Figure 9



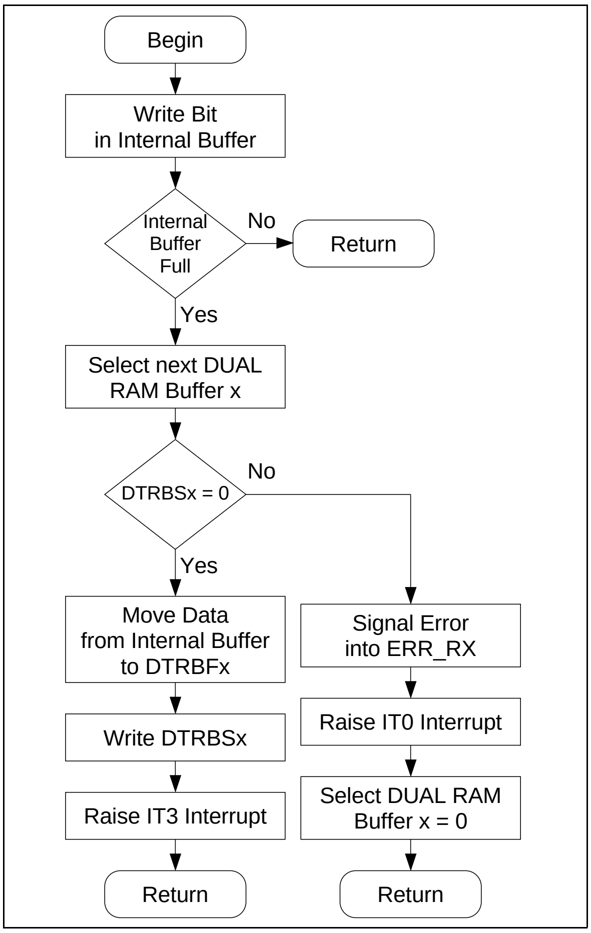
AN684-09.EPS

II.7.2 - Modem Flow Chart

When in parallel data mode, each time the ST75C502 has received some bit of data it executes the following routine (see Figure 10).

Where x start with the value 0 and toggle between 1 and 0.

Figure 10



AN684-10.EPS

II.7.3 - Host Flow Chart

Hereafter are flowcharts to :

- Establish a V.29 reception.
- Receive synchronous data. This task is performed under interrupt.
- Handle properly some temporary loss of carrier.

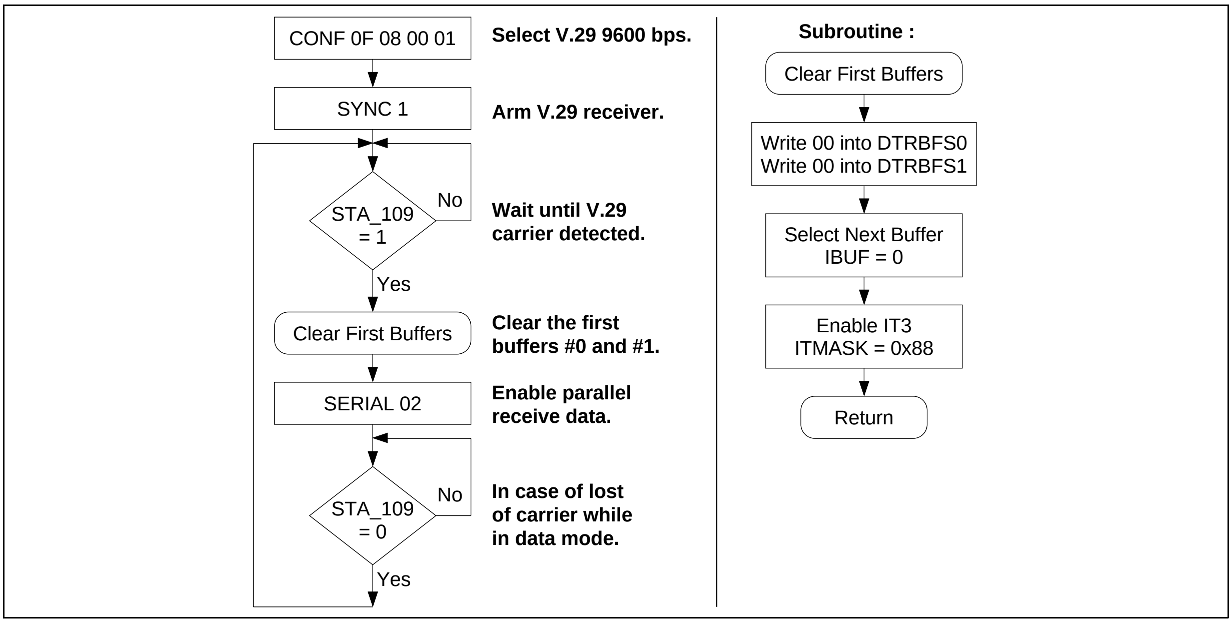
Establish the reception (see Figure 11).

These flowcharts show one CPU variable labeled ***IBUF*** which is necessary for the understanding of the mechanism, but there are different manners to implement it.

- ***IBUF*** : this is the number of the DUAL RAM buffer currently in use by the Host processor. It starts with 0 an then alternates 1, 0, 1, 0, ...

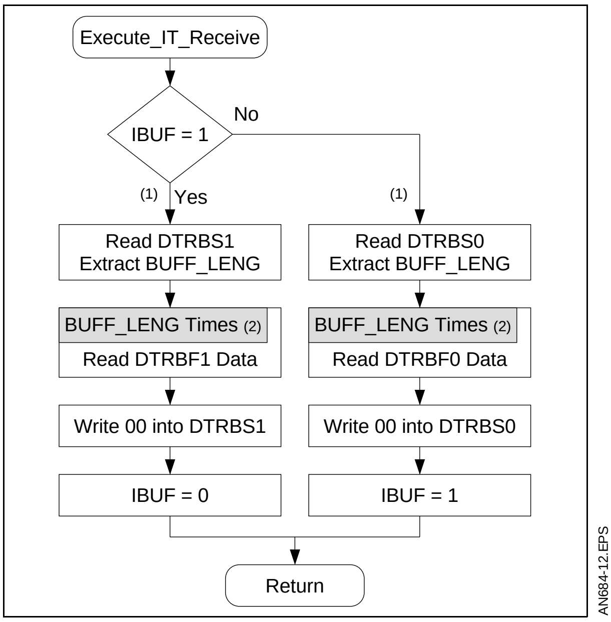
II - PARALLEL DATA EXCHANGE (continued)

Figure 11



The received bits are read by an interrupt routine (see Figure 12).

Figure 12



Notes : 1. At that step the host can check that the corresponding ***DTRBSx*** buffer is full (different from $00), otherwise it is an error.

2. This means read ***BUFF_LENG*** bytes, inside the Receive buffer ***DTRBFx*** starting from location ***DTRBFx[0]*** to ***DTRBFx[BUFF_LENG - 1]***.
The ***BUFF_LENG*** is always 8 bytes, except when a ***STA_109*** lost appears in the middle of the buffer.

II.7.4 - Error Detection

Error occurs when the ST75C502 has received some bits and that the buffer ***DTRBSx*** is not empty, this condition is called "Overflow".

This error is signaled in the bit ***ERR_RX*** of the ***SYSERR*** byte, and generates an interrupt ***IT0***. To clear the error a ***CSE 02*** command must be issued.

An Overflow condition occurs when the host processor "forgets" to empty the current ***DTRBSx*** buffer.

When an Overflow condition occurs the host must restart the whole parallel INITIALIZATION.

II.7.5 - Data Reception
II.7.5.1 - Description

The ST75C502 writes the received bit into the DUAL RAM Buffer without any modification. It starts with the Bit 0 of the ***DTRBF0[0]*** byte.

II.7.5.2 - Status Word Format

The receive Status Byte ***DTRBS0*** or ***DTRBS1*** have the same meaning (see Table 2).

The ***BUFF_LENG*** is always 8 except when a lost of carrier (***STA_109*** going to 0) happens.

This status byte is set by the ST75C502, just before generating the ***IT3*** interrupt.

II - PARALLEL DATA EXCHANGE (continued)

Table 2 : DTRBSx in Synchronous Mode

Field	Pos.	Value	Definition
BUFF_LENG	3..0	0 1 2 ... 8 Other	Buffer empty 1 Byte to transmit (***DTRBFx[0]***) 2 Bytes to transmit (***DTRBFx[0]*** and ***DTRBFx[1]***) ... 8 Bytes to transmit (***DTRBFx[0..7]***) Not used
Other	7..4	0	Not used

AN684-02.TBL

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics 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 licence is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectronics.

Purchase of I^{2}C Components of SGS-THOMSON Microelectronics, conveys a license under the Philips I^{2}C Patent. Rights to use these components in a I^{2}C system, is granted provided that the system conforms to the I^{2}C Standard Specifications as defined by Philips.

SGS-THOMSON Microelectronics GROUP OF COMPANIES
Australia - Brazil - China - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco
The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.