The DSP Based Acoustic UART

Juraj PANČÍK
Dept. of quantitative methods and informatics.
Bankovní institut vysoká škola a.s. Praha
organizačná zložka Banská Bystrica
Banská Bystrica, Slovak Republic
juraj.pancik@gmail.com

Abstract – **The paper describes the software for 16-bit fixed-point DSP (Analog Devices, ADSP21XX), which served for processing of acoustic signals. The DSP in the same time generated and received acoustic (standard and nonstandard) DTMF tones and characters in the UART function (Acoustic UART). The DSP program worked well as a core for communication as I2C slave peripheral. The I2C communication is way to drive of DTMF DSP based periphery through I2C addressable control registers. Tasks thus directed DSP described in two specific implementation projects where described DTMF peripherals were used with success (as the POS terminal in the GSM acoustic channel and as the sound based control unit for long distance light at the airport).**

Keywords : acoustic UART, DSP, DTMF, I2C peripheral, POS terminal, airport light control

I. INTRODUCTION

In this paper we describe the integration of the DSP to the host system via a synchronous serial interface, which is known as I2C bus. The literature describes how the program emulated I2C peripherals (1), (2), (3) and (4). The last source became our model for implementing the program in 16-bit fixed-point DSP ADSP2186M (manufacturer Analog Devices), which emulates the I2C peripheral, and access to the registry of DTMF DSP I2C periphery. DSP software was developed in Visual Studio DSP++ ver.3.5. It was used in the development of a hardware emulator ICE and development board EZ-KIT Lite ADSP2189M (manufacturer Analog Devices). Specific applications in projects made it impossible to use circuits for generating and receiving DTMF characters to the standard Q.23 (5). In another project the DTMF DSP I2C peripherals worked as acoustic UART (AUART) (project of the payment terminal). The complexity of the program-implemented I2C peripherals is discussed in (2). According to this source, if we want to create a program emulated interface which correspond to the standard I2C peripheral, it is necessary to implement it into DSP program with state machine with 28 states.

II. DTMF DSP I2C PERIPHERAL REGISTER ACCESS

Functions of DTMF DSP I2C receiver and transmitter were provided by own register architecture (Table 6). The Figure 1 shows a simplified block diagram of DTMF DSP I2C periphery. In addition to I2C bus signals there are two important hardware signals RX BUSY FLAG and TX BUSY FLAG which serve to detect the condition in which is located the DTMF DSP I2C periphery. Due to the use of DTMF peripherals for real-time viewing of the final data rate for I2C bus, it is necessary to inform the controller about the state of the receiver and transmitter peripherals using these two hardware signals. As a complementary signal (in the direction from DSP to control microcontroller) it is used the FLAG_uC_DSP signal. It is also necessary the signal for management hardware periphery reset (HW RESET DSP). The master controller accessed by I2C bus to I2C periphery registers in each DSP AUART by the linear part of the program carried out in the main loop of DSP program (i.e. outside of services of interruptions generated by CODEC-s ADC and DAC). Interruptions from the CODEC are served at 8000 times per second. The break of DSP main program implemented of state machines which are described below to ensure activities of acoustic DTMF receiver and transmitter. Table 5 shows the codes of characters and the basic components of dual tone multi-frequency modulation for each defined character (for the project described in section IV.B). The main program in the DSP ensures service supervisory circuits (watchdog) and other functions. In addition to basic I2C registers in the periphery AUART, the set of auxiliary registers is used for adaptive reception and broadcasting and in establishing a connection (project of the payment terminal project, section IV.A). The DSP program fulfills the test functions for frequency analysis of the input audio signal and change transmission or reception parameters. The total number of lines of assembler code for ADSP2186 is more than 4500 lines. Source code of control program for I2C master controller (ATMEL AT89C52) is written in C (1300 lines of code). Controller, beside its standalone operation mode and beside its implementation details, serves also as bridge between buses. It manages the DTMF DSP I2C periphery also via access from parallel peripheral bus (it convert 8 bit parallel bus with data and control signals ~CS, ~RD and ~WR in to the I2C bus) and also via serial communication (it convert TTL RS232 in to I2C bus) with a command-driven and robust control serial protocol.

III. THE DTMF DSP I2C PERIPHERAL PROGRAMM DESCRIPTION WITH STATE MACHINES

Final state machines (Finite State Machine, FSM, or even just state machines) are also used to describe the activities of such peripherals thus also such as communication peripherals type UART (Universal Asynchronous Transceiver). It is a known periphery adequate for asynchronous serial communications in the PC category of computers, but also in single chip microcontrollers. Although the UART can be purchased as an IP ("Intellectual Property"), for example in the form of source code in VHDL and can be integrated into the ASIC design circuit it is still available on the market as a component like NSC16C550 (National Semiconductor). The program

implementation of the UART as the complex gate arrays (CPLD) is described in (6) as a reference design communication peripherals using FSM. We developed communication peripheral AUART (Acoustic UART) and it is designed for processing „come" or "income" DTMF characters and for generate or receive auxiliary continuous tones.

The difference in the function (or activity) between UART and AUART is indeed significant, but there is a common approach to description of the operations of these both peripherals. Periphery AUART thus provides the physical layer of communication or signaling in the acoustic band (addition function to ensure of establishing communication in acoustic range between two nodes). The logical layer protocol is oriented to provide treatment and error messages and the higher application layer ensures secured communication by control microcontroller (i.e. the master I2C bus which is superior to the AUART periphery).

A. *The AUART DTMF transmitter*

DTMF transmitter in the periphery provides two registers (Table 6). Control Status Register for management and sending DTMF characters (CSR_TX) and Transmit register (TX). Transmit DTMF code register contains the code of character to be send as DTMF character (Table 5) or code of tone in the case of continuously transmitting DTMF tone. DTMF tone is transmitted continuously until the stop transmission is ordered, while sending DTMF character ends after a defined period for the duration of the DTMF character. Control bit in Register CSR_TX ensure the difference that it will be send a DTMF character (TX bit = 1) or whether it will be send DTMF tone (bit CAT_ TX = 1). End of DTMF tone transmission happens if CAT_TX bit is set to zero. By according block diagram Fig.1 broadcasting activity is monitored by setting bit BUSY BIT and the hardware signal TX BUSY FLAG to log 1. Fig 2. describe the state machine which ensure activity of DTMF transmitter. The DTMF transmitter was designed as three states according to Table 1. Transitions between states of DTMF transmitter can be described according to Table 2. The transition No.5 should be noted that it occurs not only writing zeroes to I2 TX register from the I2C master, but in practice more often, if writing zeroes to the TX DSP by itself after a fixed period of the duration of transmitting the DTMF character.

TABLE I. DTMF TRANSMITTER STATES

Initialisation INIT state :	Hardware reset of DSP or set 7.bit in the universal register UR
StateTX 0 :	WAIT. State TX BUSY BIT and hw TX: BUSY_FLAG are nonactive. In the so called universal register UR, is on position 3.–5. bits set combination bits 000
State TX 1 :	TRANSMITING OF DTMF continual tone. According to writing DTMF code to the I2C register TX. State bits indicators are active, UR has for TX set combination bits 001
State TX 2 :	TRANSMITING OF DTMF character. Sending of DTMF character after writing of the DTMF code to the I2C register TX, state bits indicators are active only until transmitting of the character, UR has for TX set combination bits 002

TABLE II. TRANSITIONS BETWEEN STATES IN THE DTMF TRANSMITTER

1.	Transition TX0 → TX1 : CAT_TX = 1 and TX = 0 and register TX != 0
2.	Transition TX1 → TX0 : CAT_TX = 0 and TX = 0 and register TX != 0
3.	Transition TX1 → TX0 : UR register bit 7. = 1 (SW reset AUART)
4.	Transition TX0 → TX2 : CAT_TX = 0 and TX = 1 and register TX != 0
5.	Transition TX2 → TX0 : CAT_TX = 0 and TX = 0 and register TX != 0
6.	Transition TX2 → TX0 : UR register bit 7. = 1 (SW reset AUART)

B. *The AUART DTMF receiver*

The AUART detects of the DTMF tone when DSP processes of the input audio signal and by calculating its to determine (using the discrete cosine transform, Goertzel algorithm) if in the input audio signal are present two energy-relevant frequency components. In order to identify of the DTMF character in the input acoustic signal the DSP periphery detects a gap during measuring the presence of DTMF tone. The presence of the DTMF character in the input acoustic signal occurs when the detected DTMF tone occupies defined time and also was recorded some defined time gap between two characters. The difference between detection of DTMF character and DTMF tone recognition is that the DTMF code is detected when the receiver identifies a DTMF tone for a defined period and subsequently identifies silence (space) also during a defined period. Thus there are three stages leads to finding DTMF character in the input acoustic signal (detection DTMF tone - identification DTMF character- recognition DTMF character). DTMF receiver is more complex than the transmitter. The DTMF receiver in the periphery AUART is represented by three I2C registers. They are the following registers: Control Status Register for managing income DTMF tones and characters (CSR_RX), Receiving register to receive DTMF tones (RX), Receiving register to receive DTMF characters (RXCH). Receiving DTMF code register contains a code of character (RXCH) or code of tone (RX). DTMF tone is taken continuously in order to stop receiving bits in Control Register CSR_RX. It ensure resolution whether is accepted a DTMF character (RX bit = 1) or whether is accepted DTMF tone (RX CAT_ bit = 1). End of incoming DTMF tone happens if CAT_RX bit is set to zero. Income activity monitor set bit RX BIT BUSY in CSR_RX register and also the hardware signal RX BUSY FLAG (see block diagram). After reading of the identified DTMF character or tone by master I2C controller from registers RX or RXCH controller is hardware signal RX BUSY FLAG set to log. 0. On the Figure 3 and Figure 4 sketched two state machines provide activities of DTMF tone and characters receiver. While Figure 3 analyzes in detail the state machine for receiving (identification) DTMF character, Figure 4 shows the states in the duration of receiving DTMF tone which consume specified

period - usually n times of the duration of DTMF characters. DTMF receiver for the detecting characters was designed according to Table 3. Transitions between states of DTMF tone receiver can be described by Table 4. Receiver DTMF tones (Figure 4) is a state machine, which after its initialization (setting bits in register CSR_RX: CAT_RX = 1) recognizes the first DTMF tone after its presence of over 25 ms and continues DTMF tone recognition until it senses that the total duration of the DTMF tone did not less for a specified period. In other words, the communication protocol required for establishing a connection between two nodes the recognition of DTMF tone with duration of 600 ms (tolerance 5 percent). State machine in the Figure 4 reliably ensure of the establishing such connection. When is establish a connection between two here described AUART-s , they both allow simultaneous transmission and reception of DTMF characters and DTMF tones from the opposite node. Details of the communication protocol and the possibility to change the parameters of the adaptive transmission and reception via AUART DSP peripherals are outside of the scope of this article.

TABLE III. DTMF CHARACTER RECEIVER STATES

Initialisation INIT state :	Hardware reset of DSP or set 7.bit in the universal register UR
State RX 0 :	WAIT. The analysing of acoustic input signal and waiting for the first DTMF character, bit in CSR_SR : RX = 1. State RX BUSY BIT and RX_ BUSY_FLAG are nonactive. In the so called universal register UR, is on position 0.–2. bits set combination bits 000.
State RX 1 :	RECEIVING OF DTMF character. It was identified first DTMF, it is loaded on to register RX .and it is waiting for master controller reading. Another settings - CSR_RX: bit RX = 1, state bits are active (CSR_RX: BUSY_BIT =1, RX : BUSY_FLAG : = 1), UR has for RX1 state combination 001. Into this state the receiver gets after setting the DTMF character recognition and after identification the first DTMF character.
State RX 2 :	The analysing of acoustic input signal and waiting for the next DTMF character. In this state master controller read the identified character from RX register. State bits ar e nonactive (CSR_RX: BUSY_BIT a BUSY_FLAG = 0).

TABLE IV. TRANSITIONS BETWEEN STATES IN THE DTMF CHARACTER RECEIVER

1.	Transition RX0 → RX1 : bits CSR_RX: CAT_RX=0 and RX=1 and BUSY_BIT=1 (DTMF character was identified by first time)
2.	Transition RX1 → RX0 : CAT_TX = 0 and TX = 0
3.	Transition RX1 → RX0 : UR register bit 7. = 1 (SW reset AUART)
4.	Transition RX1 → RX2 : bity CSR_RX: CAT_RX=0 and RX=1 and BUSY_BIT=0 (DTMF character was read from AUART)
5.	Transition RX2 → RX1 : bity CSR_RX: CAT_RX=0 and RX=1 and BUSY_BIT=1 (DTMF character was re-read from AUART)
6.	Transition RX2 → RX0 : bity CSR_RX : CAT_RX = 0 and RX = 0
7.	Transition RX2 → RX0 : UR register bit 7. = 1 (SW reset AUART)

TABLE V. NON STANDARD DTMF CODES OF CHARACTERS AND BASIC FREQUENCIES FOR DUALTONE MODULATION

		1. 705 Hz	2. 900 Hz	3. 1150 Hz	4. 1300 Hz	5. 1470 Hz	6. 1660 Hz	7. 795 Hz
1.	705 Hz	-	12	13	14	15	16	17
2.	900 Hz	-	-	23	24	25	26	27
3.	1150 Hz	-	-	-	34	35	36	37
4.	1300 Hz	-	-	-	-	45	46	47
5.	1470 Hz	-	-	-	-	-	56	57
6.	1660 Hz	-	-	-	-	-	-	67
7.	795 Hz	-	-	-	-	-	-	-

IV. USING THE DSP AUART AND CONCLUSION

A. Payment terminal with data transfer in the acoustic band of GSM phone

The payment terminal (POS, Point Of Sale) creates a communication channel during the execution of the transaction using acoustic modem AUART in the POS terminal and any GSM mobile phone. POS has as input audio interface microphone and speaker as the output interface. Acoustic modem works on a physical level with DTMF modulation. Used DTMF modulation is not standard but uses standard DTMF frequencies. The another communication node is a multi-channel DSP PC card, which carries out operation in parallel incoming GSM calls. These PC card provides also DTMF modulation and demodulation. There are number of specific requirements for the transport protocol (referred here just a few): • Absolutely flawless transmission of the transaction.• Reliable, fast and effective way of establishing a connection recovery• Reliable sided confirmation on the correct execution of transactions• The highest possible resistance to transmission failures in the acoustic band caused outages and transmission channel errors • Resistance to ambient noise in the area of practical use payment terminal. In the test operation we achieve the frequency transmission / reception: 7.35 DTMF character / s (period of transmission / reception: 136.1 ms / DTMF character.

B. Generatingt and receiving acoustic signals unit for ground airport security.

Another project where the AUART was used was the solution for architectural design of the voice communication system for dispatchers at major airports and between airports in the Russian Federation. This new facility is a part of a complex of several modules, some of which already exist. In addition to voice communication, the system uses a control (for example, airport light switching) and alarm system based on the non-standard DTMF modulation. Thus these DTMF characters and tones are defined in accordance with special (corporate)

standard of foreign customer equipments (see Table 5).

TABLE VI. BASIC CONTROL, STATUS AND DATA REGISTERS OF DSP AUART PERIPHERY DEDICATED FOR RECEIVING AND TRANSMITTING OF DTMF TONES AND CHARACTERS

I2C DSP regsiter	Register description	Bit 7.	Bit 6.	Bit 5.	Bit 4.	Bit 3.	Bit 2.	Bit 1.	Bit 0.
FSR	Register for undirect addressing								
CSR_TX	DTMF character transmitting control and status sregister						TX	CAT TX	TX BUSY BIT
TX	Transmitting register	Transmitter register with DTMF character dedicated for transmitting							
UR	Universal register	Soft RESET	-	Status of receiver DTMF characters			Status of transceiver DTMF characters		
CSR_RX	DTMF character receiving control and status sregister						RX	CAT RX	RX BUSY BIT
RX	Receiving register for receive DTMF tones	Receiving register with the code received DTMF tone							
RXCH	Receiving register for receive DTMF characters	Receiving register with the code received DTMF character							

Figure 1. The DSP AUART block scheme based on the 16 bit fixed point DSP ADSP 21865M from Analog Devices

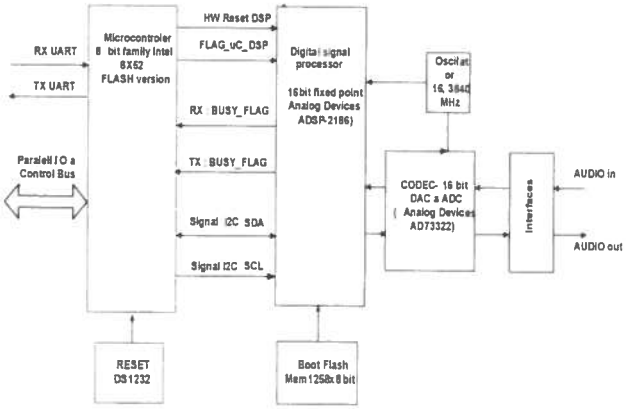


Figure 2. The AUART state machine ensures activity of the DTMF transmitter

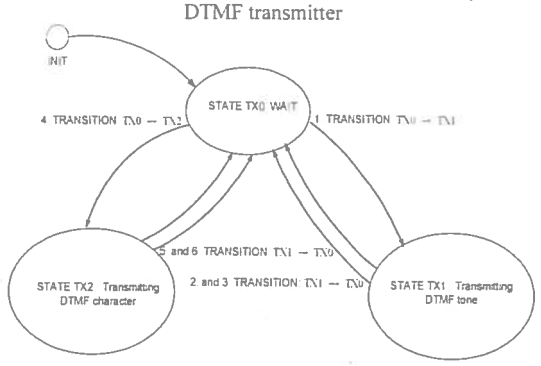


Figure 3. The AUART state machine ensures activity of the **DTMF character** receiver

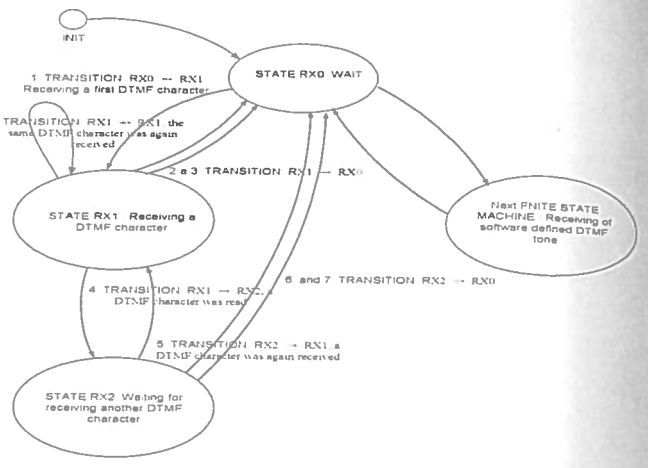
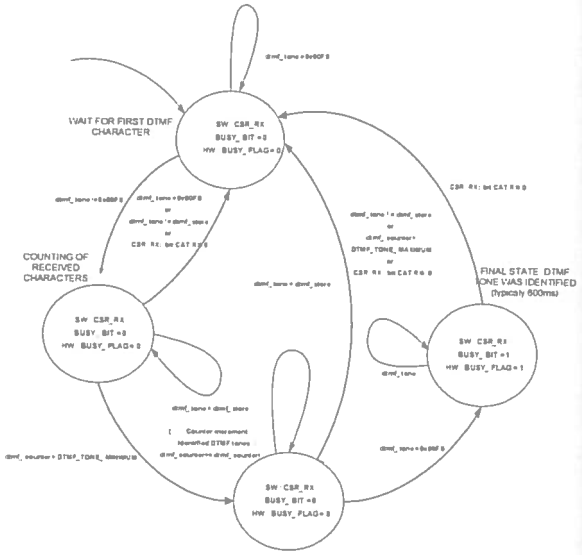


Figure 4. The AUART state machine ensures activity of the **DTMF tone** receiver



REFERENCES

[1] LATTICE. Universal Asynchronous Receiver/Transmitter February 2002 Reference Design 1011 . [Online] 2 2002. [Dátum: 10. 3 2013.] http://www.msc-ge.com/download/lattice/files/rd1011.pdf.

[2] ITU_T. Recommendation Q.23 (11/88). Q.23 : Technical features of push-button telephone sets. [Online] 19. 05 1995. [Dátum: 10. 03 2013.] http://www.itu.int/rec/T-REC-Q.23-198811-I/en.

[3] BRANDOLINO, J. AN435 I2C byte oriented system driver IC20 data . [Online] 1994. [Dátum: 10. 3 2013.] http://www.grifo.com/PRESS/DOC/Philips/AN435.pdf.

[4] QUARLES, S.D. APPLICATION NOTE AP-476 How to Implement I2C Serial Communication Using Intel MCS-51 Microcontrollers. [Online] 1993. [Dátum: 10. 3 2013.] http://www.noise.inf.u-szeged.hu/Online/Microcontroller/I2C8051.pdf.

[5] ATMEL. Two-Wire Peripheral Expansion for the AT89C2051 Microcontroller. [Online] 12 1997. [Dátum: 10. 3 2013.] http://www.noise.inf.u-szeged.hu/Online/Microcontroller/AT89C2051_I2C.pdf.

[6] LEKEI, D. Application Note AN541 : Using a PIC16C5X as a Smart I2C Peripheral. [Online] 1997. [Dátum: 10. 3 2013.] http://ww1.microchip.com/downloads/en/AppNotes/en_005411e.pdf.

18th

2013 International Conference on

Applied Electronics

Pilsen, 10 – 12 September 2013

IEEE Catalog Number CFP1369A–PRT | ISBN 978–80–261–0166–6 | ISSN 1803–7232

Department of Applied Electronics and Telecommunications
University of West Bohemia – Faculty of Electrical Engineering

UNIVERSITY OF WEST BOHEMIA

IEEE Czechoslovakia Section

IEEE

2013 International Conference on Applied Electronics

Published by the University of West Bohemia, Pilsen, September 2013

Editor	Jiří Pinker
Design	Jaroslav Fiřt
Cover photo	Zdeněk Kubík

University of West Bohemia
FEL - KAE
Univerzitni 8
306 14 Pilsen
Czech Republic
email: appel@fel.zcu.cz

Additional copies of this publication are available from

University of West Bohemia
FEL - KAE
Univerzitni 8
306 14 Pilsen
Czech Republic
email: appel@fel.zcu.cz

ISBN 978–80–261–0166–6 (Print)

ISBN 978–80–261–0165–9 (Online)

ISSN 1803–7232 (Print)

ISSN 1805–9597 (Online)

IEEE Catalog Number CFP1369A–PRT (Print)

IEEE Catalog Number CFP1369A–ART (Online)

Table of Contents

Invited Lecture