



ComAir5 IOTX Library User's Manual V1.0

08/21/2013

0 Table of Content

0	TABLE OF CONTENT	2
1	REVISION HISTORY.....	3
1.1	Document History	3
2	COMAIR5 IOTX SUMMARY	4
2.1	Transmission Distance.....	4
2.2	Speaker and Microphone Selection	4
2.3	Environment Consideration.....	4
2.4	Filter in the Electrical Circuit	4
2.5	The Data Format.....	4
3	LIBRARY SERVICE LOOPS	5
4	API OF COMAIR5 IOTX.....	6
4.1	Hardware Dependent Functions Function: Initializes ComAir5 IOTX.....	6
4.1.1	Function: Initialize ComAir5 IOTX module, sets Interrupt source, Timer before running	6
4.2	ComAir5 IOTX Control	6
4.2.1	Function: Start Transmitting.....	6
4.2.2	Function: Get ComAir5 IOTX Status.....	6
4.3	ISR Functions: Interrupt service routine for ComAir5 IOTX	7
4.4	User Functions: for ComAir5 IOTX	7
4.4.1	Function: Hardware Initial for ComAir5_IOTX	7
4.4.2	Function: Get Timer Data of ComAir5_IOTX.....	8
4.4.3	Function: Write Timer Data of ComAir5_IOTX.....	8
4.4.4	Function: Read Timer Data of ComAir5_IOTX.....	8
4.4.5	Function: INT OFF of ComAir5_IOTX.....	8
4.4.6	Function: INT ON of ComAir5_IOTX	9
5	RESOURCES LIST OF COMAIR5_IOTX.....	10
5.1	TABLE 1: RAM Size (Unit: Decimal Word).....	10
5.2	TABLE 2: ROM Size (Unit: Decimal Word).....	10
5.3	TABLE 3: Hardware Resources VS Library	10
5.4	TABLE 5: CPU Usage Rate (approximate).....	10
5.5	TABLE 6: Name of Overlap RAM in the library	10

1 Revision History

1.1 Document History

Revision	Date	By	Remark
V1.0	08/21/2013	Porter Yang	Original Version

2 ComAir5 IOTX Summary

The ComAir5 IOTX algorithm provides the ability to transmit commands through the air. This algorithm can be used in communication areas. Below are some recommended considerations developers should consider.

2.1 Transmission Distance

The maximum transmission distance between transmitter and receiver depends on the room acoustics, echoes, and environment noise. Also the transmission power or sound amplitude plays an important role. Generally speaking, if the separation distance is up to 5meters, the transmission power should be larger than 50dB.

2.2 Speaker and Microphone Selection

The separation distances talked above assume selection of proper high-quality microphones and speakers. Many brands of electret microphones have flat frequency responses to 20 kHz, which is more than adequate. However, not all microphones meet this specification. Avoid microphones that do not have flat frequency responses to 20 kHz.

Selection of speakers is more difficult because many, but not all, inexpensive speakers are satisfactory and meet specifications. Thus, the best approach is to try several speakers and select one that produces audio signals that are sufficiently loud and that works as a sonic tone component to the distances required by the application.

2.3 Environment Consideration

If there is any sound absorbing material in front of the microphone or speaker, the performance can degrade a lot. The material covering microphones and speakers must be the minimum possible.

2.4 Filter in the Electrical Circuit

The electrical filters connected to microphone or speaker should have upper cut off frequency not less than working frequency. The proper frequency response can be achieved by proper selection of resistors and capacitors. resistors and capacitors.

2.5 The Data Format

1 tone	3 tone	1 tone
SYNC	DATA	CHECKSUM

SYNC: Synchronization signal.

DATA: Total of 80 command

CHECKSUM: check the accuracy of data transmitted.

To send a command takes 1.1 second.

3 Library Service loops

The ComAir5 IOTX library can support the foreground service loops. In foreground service loop, users have to put the service loop in the main routine to keep entering the service regularly. Inside the ComAir5 service loop, there will be a mechanism to decide any task should be carried on. Some overhead will produced inevitably. The amount of overhead varies and depends on the payload of CPU.

Example:

Foreground service loop:

In main.c

```
int main()
{
    System_Initial();                // System initial
    Ret = ComAir5_IOTX_Init(17000, 200); // ComAir5 IOTX initial
    ComAir5_IOTX_Send(Command);        // Start transmitting command
    while(ComAir5_IOTX_Status() == 0)
    {
        System_ServiceLoop();        // Service loop for watchdog clear
    } // end of while
    return 0;
} // end of main
```

In isr.asm:

```
_FIQ:
    push R1, R5 to [SP];            // save registers
    call F_ISR_Service_ComAir5_IOTX // Interrupt service routine
    R1 = C_IRQ6_4096Hz;              // Timebase = 4096Hz
    [P_INT_Status] = R1;
    pop R1, R5 from [SP];           // restore registers
    reti;
```

4 API of ComAir5 IOTX

4.1 Hardware Dependent Functions Function: Initializes ComAir5 IOTX

4.1.1 Function: Initialize ComAir5 IOTX module, sets Interrupt source, Timer before running

Syntax:

C: void ComAir5_IOTX_Init(int CentFreq, int DeltaFreq)

R1 = CentFreq

R2 = DeltaFreq

ASM: Call F_ComAir5_IOTX_Init

Parameters: CentFreq: 4000~19000Hz.

DeltaFreq: User directly sets delta frequency

Return Value: None

Library: ComAir5_IOTX_Vxxx.lib

Remark: 1. This function initializes the Kernel of ComAir5 IOTX. It also initializes the Timer A, Timebase and IOPWM which used to send ComAir5 signal.

2. The hardware setting is opened for user's reference (see F_ComAir5_IOTX_HW_Init in ComAir5_IOTX_User.asm).

3. Available central frequency is from 4000Hz to 19000Hz

4.2 ComAir5 IOTX Control

4.2.1 Function: Start Transmitting

Syntax:

C: void ComAir5_IOTX_Send (int command)

ASM: R1 = command

Call F_ComAir5_IOTX_Send

Parameters: command: 0~79

Return Value: None

Library: ComAir5_IOTX_Vxxx.lib

Remark: Total available quantity is 80.

4.2.2 Function: Get ComAir5 IOTX Status

Syntax:

C: int ComAir5_IOTX_Status (void)

ASM: Call F_ComAir5_IOTX_GetStatus

Parameters: None

Return Value: R1 [b15,b14]: 00: Standby

01: Ramp Up

10: Ramp Down

11: Data Sending

Library: ComAir5_IOTX_Vxxx.lib

Remark: Check IOTX status.

4.3 ISR Functions: Interrupt service routine for ComAir5 IOTX

Syntax:

C: None

ASM: Call F_ISR_Service_ComAir5_IOTX

Parameters: None

Return Value: None

Library: ComAir5_IOTX_Vxxx.lib

Remark:

1. This function is used in assembly only and it can be hooked on the _FIQ, _IRQ1 or _IRQ2: label. (See isr.asm for details)
2. It is possible for users to place user-define function in the same FIQ or IRQ.

EX:

```
_FIQ:
    push r1,r5 to [sp];

    call F_ISR_Service_ComAir5_IOTX
    call F_User_ISR

    R1 = C_IRQ6_4096Hz
    [P_INT_Status] = R1;

    pop r1,r5 from [sp]
    reti
```

3. This function will not destroy the content in R1-R5 register. Programmers have not to protect the registers externally.

4.4 User Functions: for ComAir5 IOTX

4.4.1 Function: Hardware Initial for ComAir5_IOTX

Syntax:

C: None

ASM: Call F_ComAir5_IOTX_HW_Init

Parameters: None

Return Value: None

Library: ComAir5_IOTX_User.asm

- Remark:**
1. Set IOPWM source as TimerA
 2. Set Timebase as 4096Hz

4.4.2 Function: Get Timer Data of ComAir5_IOTX

Syntax:

C: None

ASM: Call F_ComAir5_IOTX_GetTimerData

Parameters: R2: Frequency

Return Value: R1: Timer Data

Library: ComAir5_IOTX_User.asm

Remark: The formula is $0x10000 - ((\text{SystemClock}/\text{Frequency}) + 1)/2$

4.4.3 Function: Write Timer Data of ComAir5_IOTX

Syntax:

C: None

ASM: Call F_ComAir5_IOTX_WriteTimerData

Parameters: R3: Timer Data

Return Value: None

Library: ComAir5_IOTX_User.asm

Remark: 1. Update Timer Data

4.4.4 Function: Read Timer Data of ComAir5_IOTX

Syntax:

C: None

ASM: Call F_ComAir5_IOTX_ReadTimerData

Parameters: None

Return Value: R3: Timer Data

Library: ComAir5_IOTX_User.asm

Remark: 1. Get Timer Data

4.4.5 Function: INT OFF of ComAir5_IOTX

Syntax:

C: None

ASM: Call F_ComAir5_IOTX_INT_OFF

Parameters: None

Return Value: None

Library: COMAIR5_IOTX_User.asm

Remark: Disable interrupt

4.4.6 Function: INT ON of ComAir5_IOTX

Syntax:

C: None

ASM: Call F_ComAir5_IOTX_INT_ON

Parameters: None

Return Value: None

Library: ComAir5_IOTX_User.asm

Remark: Enable interrupt

5 Resources List of ComAir5_IOTX

5.1 TABLE 1: RAM Size (Unit: Decimal Word)

	IRAM	ISRAM	RAM	SRAM	ORAM	OSRAM
ComAir5_IOTX					15	

5.2 TABLE 2: ROM Size (Unit: Decimal Word)

	TEXT	CODE	DATA	USER DEFINE
ComAir5_IOTX	1886	159		

5.3 TABLE 3: Hardware Resources VS Library

	Interrupt	Timer Setting	Audio
ComAir5_IOTX	IRQ6 Timebase	4096Hz	None

5.4 TABLE 5: CPU Usage Rate (approximate)

	GPCE063A CPU Usage Rate (49 MHz) while ComAir5 IOTX
ComAir5_IOTX	2.1% (5.1us/244us)

5.5 TABLE 6: Name of Overlap RAM in the library

Table: Name of Overlap RAM in the library

	Overlap RAM definition	
<i>Algorithm</i>	Overlap RAM Label	Size(word)
ComAir5_IOTX	OVERLAP_ComAir5_IOTX_ORAM	15