

***TMS320C54x
Hands-Free Development Platform
UART Driver
API Reference Guide***

Literature Number: SPRU706
December 2003



IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		Applications	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Copyright © 2003, Texas Instruments Incorporated

Preface

Read This First

About This Manual

Stonestreet One has provided a hardware UART driver for TI's Hands-Free Development Platform project. This API reference contains a description of the programming interfaces for the hardware UART driver.

How to Use This Manual

This document contains the following sections:

- Section 1 introduces the TMS320C54x hands-free development platform UART driver.
- Section 2 focuses on the integrating the UART device driver to the application's configuration.
- Section 3 describes the available UART driver command functions.
- Section 4 contains the source and header files required for the UART implementation.

Related Documentation From Texas Instruments

The following documents provide additional background information on some of the features and functions of the Hands-Free Kit development platform.

TMS320C54x DSP Reference Set, Volume 2: Mnemonic Instruction Set (SPRU172)

DSP/BIOS Device Driver Developer's Guide (SPRU616)

TMS320C54x Assembly Language Tools User's Guide (SPRU102)

TMS320C54x DSP Programmer's Guide (SPRU538)

TMS320C54x DSP/BIOS User's Guide (SPRU326)

XDS560 Emulator Reference Guide (SPRU589)

Trademarks

Code Composer Studio Getting Started Guide (SPRU509)

TLV320AIC20 Data Manual (SLAS363)

TLV320AIC24 Data Manual (SLAS366).

The documentation listed above is available for download from <http://www.ti.com>.

Trademarks

Trademarks are the property of their respective owners.

Bluetopia® is a registered trademark of Stonestreet One, Inc.

Bluetooth™ is a trademark of Bluetooth Special Interest Group (SIG).

Contents

1	Introduction	1
2	Integration	1
3	UART Driver Programming Interface	2
3.1	UART Driver Commands	2
	C54XX_SS1_HWUART_Open	2
	C54XX_SS1_HWUART_Write	3
	C54XX_SS1_HWUART_Read	3
	C54XX_SS1_HWUART_Isr	4
3.2	UART Driver Callback Prototype	4
	C54XX_SS1_HWUART_Receive_Data_Callback_t	4
4	File Distributions	5

Tables

1	UART Driver Commands	2
2	Required Source and Header Files	5

TMS320C54x Hands-Free Development Platform UART Driver

1 Introduction

The hardware UART driver of the C5407 is used to connect to the Bluetooth daughtercard. This API reference contains a description of the programming interfaces for the hardware UART driver.

2 Integration

To use the UART device driver, you must first add the UART interrupt service routine to your applications' configuration file. The HWI object that is added to the interrupt service routine is processor-specific. For the C5407 DSP, the hardware UART is mapped to the HWI_SINT14 HWI object. The function property for the HWI objects for the UART should be set to the **C54XX_SS1_HWUART_Isr()** function. The "use dispatcher" property for this HWI object should be set to TRUE.

C54XX_SS1_HWUART_Open

3 UART Driver Programming Interface

3.1 UART Driver Commands

The available UART driver command functions are listed in Table 1 and are described in the text that follows.

Table 1. *UART Driver Commands*

Function	Description
C54XX_SS1_HWUART_Open	Sets up and opens the UART.
C54XX_SS1_HWUART_Write	Buffers data into the UART driver's transmit buffer.
C54XX_SS1_HWUART_Read	Reads data from the UART driver's receive buffer.
C54XX_SS1_HWUART_Isr	Handles interrupts generated by the hardware UART.

C54XX_SS1_HWUART_Open

Description This function sets up and opens the UART. It should not be called until after DSP/BIOS has begun running.

Prototype int **C54XX_SS1_HWUART_Open**
(C54XX_SS1_HWUART_Setup_Parameters_t *SetupParameters)

Parameters

SetupParameters Pointer to the setup parameters structure containing the information required to setup the UART. This is defined by the following structure:

```
typedef struct _tagC54XX_SS1_HWUART_Setup_t
{
    unsigned long    ProcessorClockFrequency;
    unsigned long    UartBaudRate;
    unsigned int     ReceiveBufferSize;
    unsigned char    *ReceiveBufferPtr;
    unsigned int     TransmitBufferSize;
    unsigned char    *TransmitBufferPtr;
    unsigned long    CallbackParameter;
    C54XX_SS1_HWUART_Receive_Data_Callback_t
        ReceiveDataCallback;
} C54XX_SS1_HWUART_Setup_Parameters_t;
```

where, `ReceiveDataCallback` is the callback described in section 3.2.

C54XX_SS1_HWUART_Read

Return Value Zero if successful.

A negative error code on failure of the following value:

- C54XX_SS1_HWUART_ERROR_PARAMETER_ERROR

C54XX_SS1_HWUART_Write

Description This function buffers data into the UART driver's transmit buffer. The driver must have been successfully opened before calling this function.

Note:

This function will block until all of the indicated data has been written to the driver's transmit buffer.

Prototype int **C54XX_SS1_HWUART_Write**(int DataLength, unsigned char *DataBufferPtr)

Parameters

DataLength Length of data in the buffer to be written.
DataBufferPtr Pointer to the data buffer containing the data to be written.

Return Value Amount of data written if successful (should be full length).

A negative error code on failure, one of the following values:

- C54XX_SS1_HWUART_ERROR_DEVICE_NOT_OPEN
- C54XX_SS1_HWUART_ERROR_PARAMETER_ERROR

C54XX_SS1_HWUART_Read

Description This function reads received data from the UART driver's receive buffer. The driver must have been successfully opened before calling this function.

Note: This function should be called from a software interrupt after the installed callback function indicates data is available. This function should not be called directly from the callback, as the callback is executed from a hardware interrupt.

Prototype int **C54XX_SS1_HWUART_Read**(int DataLength, unsigned char *DataBufferPtr)

C54XX_SS1_HWUART_Isr

Parameters

DataLength	Length of the data buffer in which the read data will be returned.
DataBufferPtr	Pointer to the data buffer in which the received data will be written.

Return Value

Amount of data read if successful.

A negative error code on failure, one of the following values:

- C54XX_SS1_HWUART_ERROR_DEVICE_NOT_OPEN
- C54XX_SS1_HWUART_ERROR_PARAMETER_ERROR

C54XX_SS1_HWUART_Isr

Description

This function is responsible for handling interrupts generated by the hardware UART. It should be installed in the configuration file as the function called for the IRQ_EVT_UART interrupt. The HWI object is processor-specific and can be found in the processor's datasheet.

Prototype

void **C54XX_SS1_HWUART_Isr**(void)

Parameters

none

Return Value

none

3.2 UART Driver Callback Prototype

C54XX_SS1_HWUART_Receive_Data_Callback_t

Description

This function will be called whenever data is available to be read. It passes to the caller the callback parameter that was specified when the UART driver was opened. There can be no multiple invocations of this function (i.e., it is not reentrant). It will be called from a hardware interrupt routine and is therefore as efficient as possible. Ideally, this callback function would simply post a software interrupt and then return. The software interrupt can then run and read the available data from the driver.

Prototype

void (*C54XX_SS1_HWUART_Receive_Data_Callback_t)(unsigned long CallbackParameter)

Parameters

CallbackParameter User-defined parameter (e.g., tag value) that was defined in the **C54XX_SS1_HWUART_Open** call.

4 File Distributions

The source and header files required for the UART implementation are listed in Table 2.

Table 2. Required Source and Header Files

File	Contents/Description
C54xx_ss1_hwuart.h	UART header module (prototypes and constants).
C54xx_ss1_hwuart.c	UART source code module (implementation).

