T. Y. B. Tech (ECE)

Semester: VI **Subject:** ESD&RTOS

Name: Satej Zunjarrao **Division: B**

Roll No: PB-30 **Batch: B3**

Experiment No: 05

Name of the Experiment: Programming UART of LPC 2148

Performed on:

Submitted on:

Marks	Teacher's Signature with date

Aim: Write Embedded C program to interface LPC2148 based ARM EPB7 with PC using UART.

a) Transfer data from LPC2148 to PC.

b) Receive data sent from PC on LPC2148 and display on hyper terminal.

Part List:

- Educational practice board for ARM7 LPC2148
- +9V Power supply
- USB A to B type cable
- PC
- Eclipse IDE
- Flash Magic Utility

Theory:

LPC2148 has two inbuilt UART (Universal Asynchronous Receiver Transmitter) modules viz. UART0 and UART1. UART0 provides only standard transmit and receive data lines. UART1 provides a full modem control handshake interface along with standard transmit and receive data lines. UART0 in asynchronous mode is used to connect the LPC2148 to PC serial port for the purpose of full duplex serial data transfer. Baud rate is a significant factor for serial communication

of microcontroller with other devices. For communication with PC the baud rate of 9600 is selected.

Hardware Connection:

UART0 signals	Signal
TXD0	P0.0
RXD0	P0.1

Program:

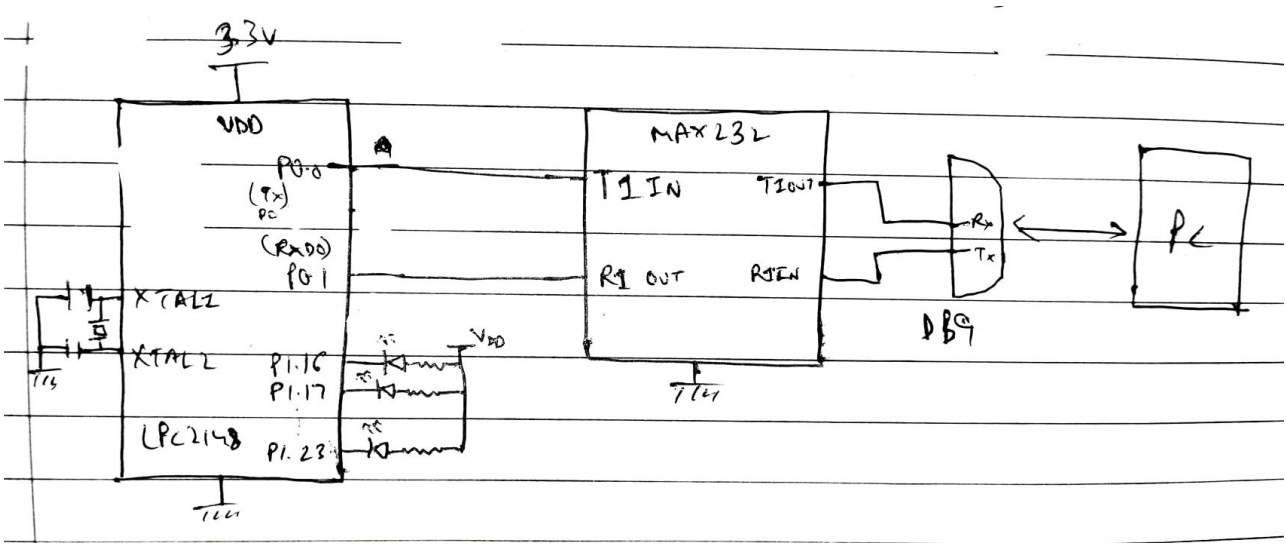
```
#include "lpc214x.h"
#define FOSC 12000000
void Transm(char);
char Receivem();

int main()
{
        //char data
        PINSEL0 = 0X00000005;
        VPBDIV = 0x01;
        IO0DIR = 0x01;
        U0FCR  =  0x07;
        U0LCR  =  0x87;
        U0DLL  =  0x4E;
        U0DLM = 0x00;
        U0LCR  =  0x07;
        U0TER = 0x80;
        Transm('A');
        while(1)
```

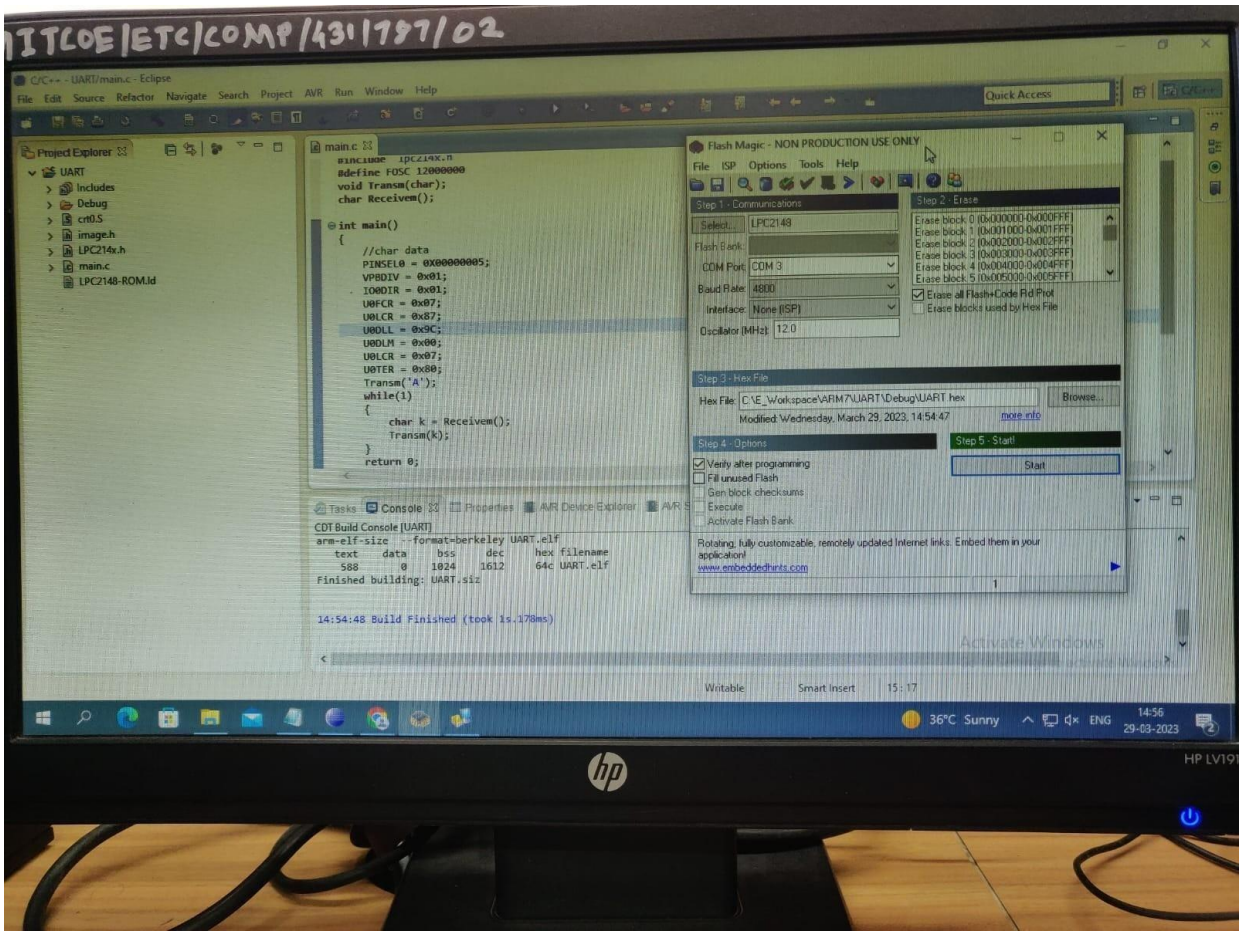
```
        {
                char k = Receiver();
                Transm(k);
        }
        return 0;
}

void Transm(char a)
{
        U0THR=a;
        while (!(U0LSR & 0x20));
}

char Receivem()
{
        while(!(U0LSR & 0x01));
        char b = U0RBR;
        return b;
}
```

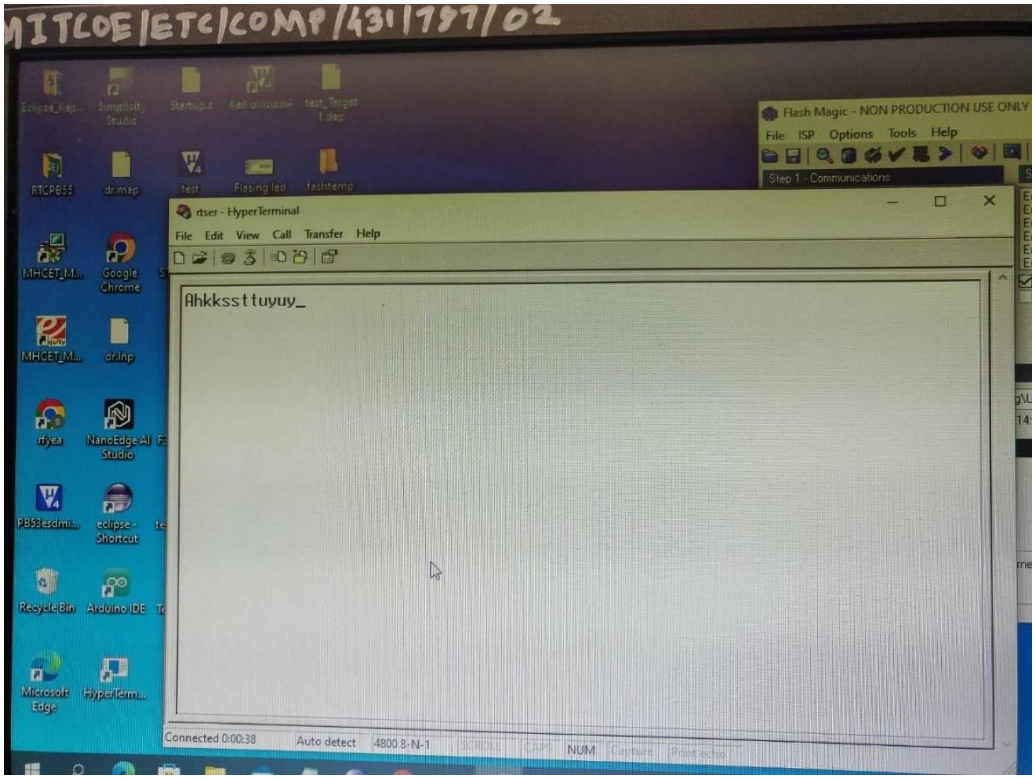
Result:

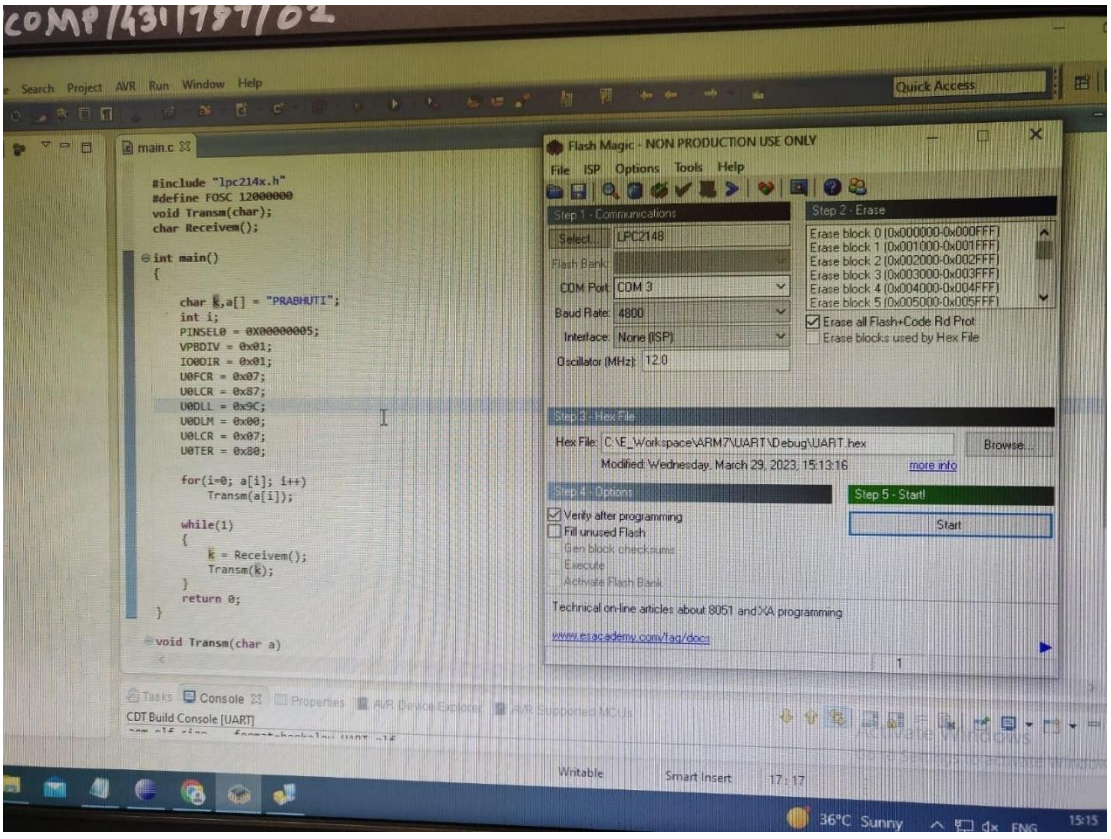
i) The string should be displayed on HyperTerminal.
ii) The received character should be displayed on HyperTerminal.

Interfacing Diagram:



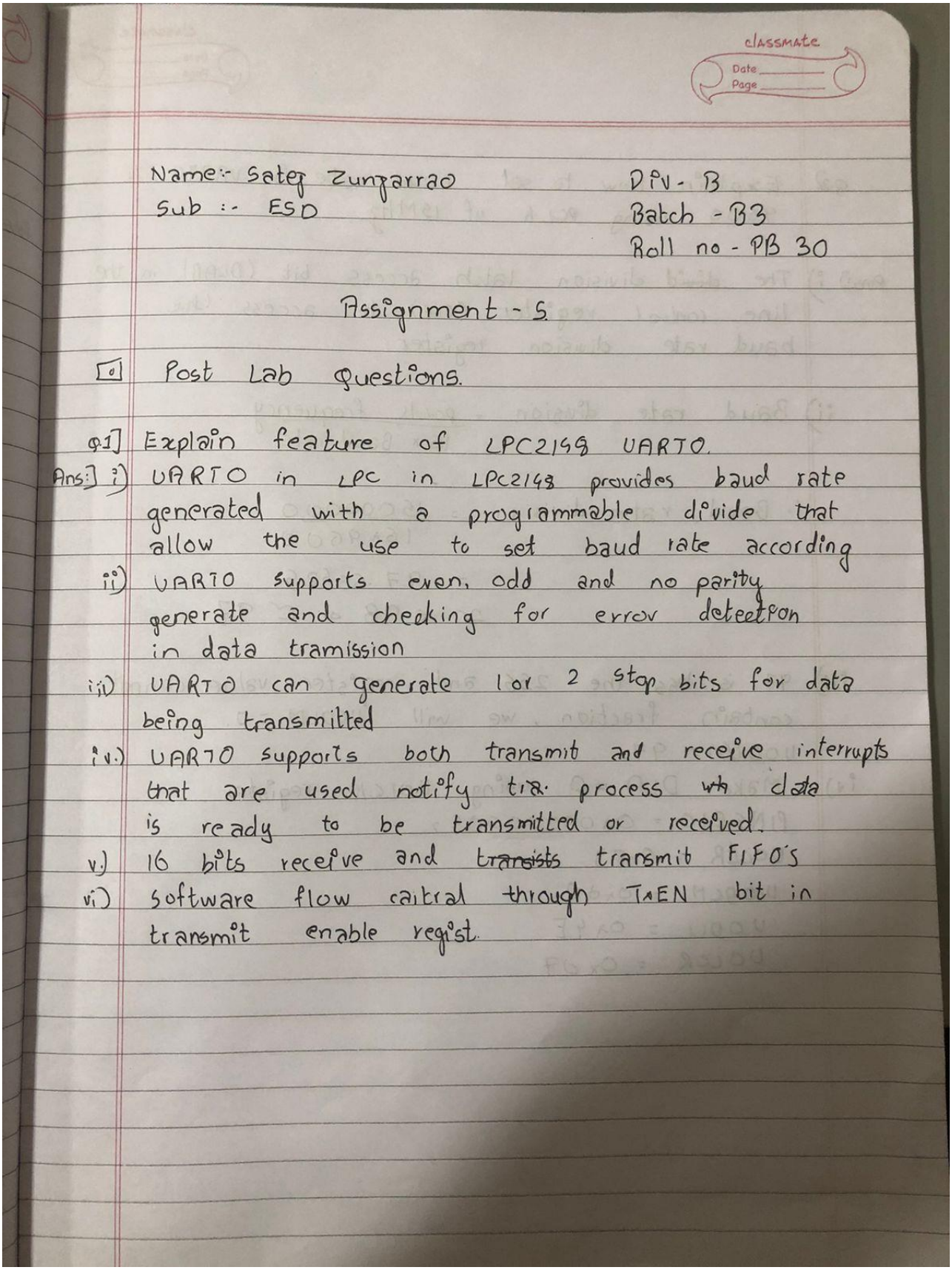



MITCOE/ETC/COMP/431/787/02
Flash Magic - NON PRODUCTION USE ONLY
rtser - HyperTerminal
Ahkkssttuyuy_
Connected 0:00:38
Auto detect
4800 8-N-1


COMP/431/787/02
main.c
#include "lpc214x.h"
#define FOSC 12000000
void Transm(char);
char Receivem();
int main()
char k,a[] = "PRABHUTI";
Flash Magic - NON PRODUCTION USE ONLY
Step 1 - Communications
Step 2 - Erase
Step 3 - Hex File
Step 4 - Options
Step 5 - Start!
Start
Technical on-line articles about 8051 and XA programming
CDT Build Console [UART]
36°C Sunny

Conclusion:

We wrote an embedded C program to interface LPC2148 based ARM EPB7 with a PC using UART. We transmitted from UART to PC and vice versa as well as transmitted our names andreceived them back.



classmate
Date
Page

Name:- Sateq Zunjarrao
Sub :- ESD

Div - B
Batch - B3
Roll no - PB 30

Assignment - 5

Post Lab Questions.

Q1] Explain feature of LPC2148 UART0.

Ans:] i) UART0 in LPC in LPC2148 provides baud rate generated with a programmable divide that allow the use to set baud rate according

ii) UART0 supports even, odd and no parity generate and checking for error detection in data tramission

iii) UART0 can generate 1 or 2 stop bits for data being transmitted

iv.) UART0 supports both transmit and receive interrupts that are used notify the process when data is ready to be transmitted or received.

v.) 16 bits receive and ~~transists~~ transmit FIFO's

vi) Software flow caitral through TxEN bit in transmit enable regist.

classmate
Date
Page

Q.22 Explain how to set baud rate is UART0 to 9600 using PCLK of 15MHz

Ans:) i) The ~~divid~~ division latch access bit (DLAB) in the line control register (CLE) to access the baud rate division register.

ii) Baud rate division = $\frac{\text{PCLK frequency}}{16 \times \text{Baud rate.}}$

$\therefore$ Baud rate division = $\frac{15000000}{16 \times 9600}$

$= 97.65625$

$\simeq = 98$ or $\simeq 97$

iii) 97 is less the 256 and register value cannot conbain fraction, we will U0DLM = 0.

U0DLM = 97.

iv) Make DLA = 0 using U0LCK register

PINSEL0 = 0x00000005;

U0CLR = 0x83

U0BCM = 0x00

U0DLL = 0x4E

U0LCR = 0x07