**//C programming of LPC1768 I2C to send data to slave**

#include<stdio.h>

#include <lpc17xx.h>

void i2c\_init(void);

int i2c\_start();

int i2c\_write(unsigned char buff);

void i2c\_stop(void);

void wait(unsigned int delay);

void i2c\_init(void)

{

//Select alternate function 2 (I2C) for pins P0.10 and P0.11

LPC\_PINCON->PINSEL0 |= 0x02<<20; //Pin P0.10 allocated to alternate function 2

LPC\_PINCON->PINSEL0 |= 0x02<<22; //Pin P0.11 allocated to alternate function 2

//Enable power to peripheral I2C2

LPC\_SC->PCONP |= 1 << 26; //re-enable POWER to I2C\_2 if required

//SEt the peripheral clock for I2C2 to the system clock frequency (CCLK).

LPC\_SC->PCLKSEL1 |= 1<<20; //pclk = cclk

//I2C2 pins as output, open drain and no pull up or pull down resistors.

LPC\_GPIO0->FIODIR |= 1<<10; //Bit P0.10 an output

LPC\_GPIO0->FIODIR |= 1<<11; //Bit P0.11 an output

LPC\_PINCON->PINMODE0 &= ~(3<<20); //P0.10 has pull up/down resistor

LPC\_PINCON->PINMODE0 |= (2<<20); //omit to use internal pull up

LPC\_PINCON->PINMODE0 &= ~(3<<22); //P0.11 has pull up/down resistor

LPC\_PINCON->PINMODE0 |= (2<<22); //omit to use internal pull up

LPC\_PINCON->PINMODE\_OD0 |= 1<<10; //Bit P0.10 is open drain

LPC\_PINCON->PINMODE\_OD0 |= 1<<11; //Bit P0.11 is open drain

//Determine possible values of I2CSCLH and I2CSCLL to give standard (100kHz) I2C rate given

LPC\_I2C2->I2SCLH = 60; //100kHz from 12MHz

LPC\_I2C2->I2SCLL = 60; //100kHz from 12MHz

//Enable I2C2 interface

LPC\_I2C2->I2CONSET =(1<<6); //I2EN bit

}

int i2c\_start() //I2C communication start

{

LPC\_I2C2->I2CONCLR=(1<<3);//SIC bit

LPC\_I2C2->I2CONSET=(1<<5);//STA bit

while(!(LPC\_I2C2->I2STAT==0x08));

return 0;

}

int i2c\_write(unsigned char buff) //Data writing through I2C

{

LPC\_I2C2->I2CONSET|=(1<<3);//SI bit

LPC\_I2C2->I2DAT=buff;//write data

wait(5000);

LPC\_I2C2->I2CONCLR=(1<<3);//SIC bit

wait(5000);

return 0;

}

void i2c\_stop(void) //Stop the I2C communication

{

LPC\_I2C2->I2CONSET=(1<<4);//STO bit

LPC\_I2C2->I2CONCLR=(1<<6);//I2ENC bit

}

void wait(unsigned int delay)

{

while(delay--);

}

int main(void)

{

while(1)

{

i2c\_init();

i2c\_start();

i2c\_write('0');

wait(500000);

i2c\_write('1');

wait(500000);

i2c\_write('2');

wait(500000);

i2c\_write('4');

wait(500000);

i2c\_stop();

}

}

**//C programming of LPC1768 I2C to send data to slave and port1**

#include<stdio.h>

#include <lpc17xx.h>

void i2c\_init(void);

int i2c\_start();

int i2c\_write(unsigned char buff);

void i2c\_stop(void);

void wait(unsigned int delay);

void i2c\_init(void)

{

//Select alternate function 2 (I2C) for pins P0.10 and P0.11

LPC\_PINCON->PINSEL0 |= 0x02<<20; //Pin P0.10 allocated to alternate function 2

LPC\_PINCON->PINSEL0 |= 0x02<<22; //Pin P0.11 allocated to alternate function 2

//Enable power to peripheral I2C2

LPC\_SC->PCONP |= 1 << 26; //re-enable POWER to I2C\_2 if required

//SEt the peripheral clock for I2C2 to the system clock frequency (CCLK).

LPC\_SC->PCLKSEL1 |= 1<<20; //pclk = cclk

//I2C2 pins as output, open drain and no pull up or pull down resistors.

LPC\_GPIO0->FIODIR |= 1<<10; //Bit P0.10 an output

LPC\_GPIO0->FIODIR |= 1<<11; //Bit P0.11 an output

LPC\_PINCON->PINMODE0 &= ~(3<<20); //P0.10 has pull up/down resistor

LPC\_PINCON->PINMODE0 |= (2<<20); //omit to use internal pull up

LPC\_PINCON->PINMODE0 &= ~(3<<22); //P0.11 has pull up/down resistor

LPC\_PINCON->PINMODE0 |= (2<<22); //omit to use internal pull up

LPC\_PINCON->PINMODE\_OD0 |= 1<<10; //Bit P0.10 is open drain

LPC\_PINCON->PINMODE\_OD0 |= 1<<11; //Bit P0.11 is open drain

//Determine possible values of I2CSCLH and I2CSCLL to give standard (100kHz)

//I2C rate given

LPC\_I2C2->I2SCLH = 60; //100kHz from 12MHz

LPC\_I2C2->I2SCLL = 60; //100kHz from 12MHz

//Enable I2C2 interface

LPC\_I2C2->I2CONSET =(1<<6); //I2EN bit

}

int i2c\_start() //I2C communication start

{

LPC\_I2C2->I2CONCLR=(1<<3);//SIC bit

LPC\_I2C2->I2CONSET=(1<<5);//STA bit

while(!(LPC\_I2C2->I2STAT==0x08));

return 0;

}

int i2c\_write(unsigned char buff) //Data writing through I2C

{

LPC\_GPIO1->FIODIR = 0xff;

LPC\_I2C2->I2CONSET|=(1<<3);//SI bit

LPC\_I2C2->I2DAT=buff;//write data

**LPC\_GPIO1->FIOPIN=LPC\_I2C2->I2DAT;//write data to port1**

wait(5000);

LPC\_I2C2->I2CONCLR=(1<<3);//SIC bit

wait(5000);

return 0;

}

void i2c\_stop(void) //Stop the I2C communication

{

LPC\_I2C2->I2CONSET=(1<<4);//STO bit

LPC\_I2C2->I2CONCLR=(1<<6);//I2ENC bit

}

void wait(unsigned int delay)

{

while(delay--);

}

int main(void)

{

while(1)

{

i2c\_init();

i2c\_start();

i2c\_write('0');

wait(500000);

i2c\_write('1');

wait(500000);

i2c\_write('2');

wait(500000);

i2c\_write('4');

wait(500000);

i2c\_stop();

}

}