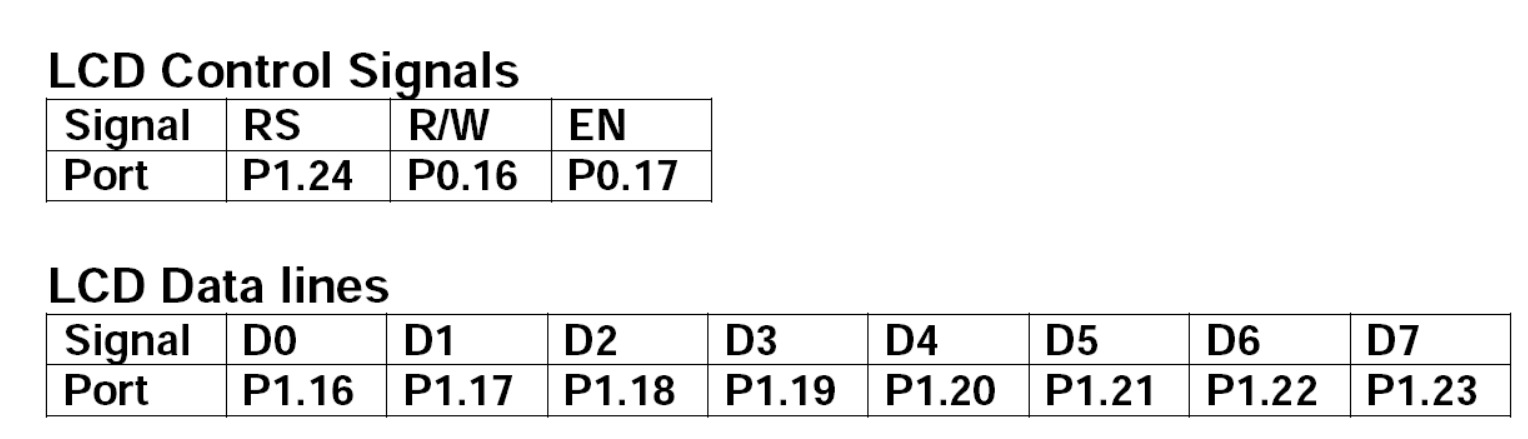
Real Time Clock



#include <lpc214x.h> // Header file for LPC2148

void delay\_ms(unsigned int ms) {

unsigned int i, j;

for (i = 0; i < ms; i++) {

for (j = 0; j < 6000; j++); // Approximate delay loop

}

}

// Send command to LCD

void lcd\_command(unsigned char cmd) {

IO1CLR = (0xFF << 16); // Clear data lines P1.16-P1.23

IO1SET = (cmd << 16); // Send command to data lines

IO1CLR = (1 << 24); // RS = 0 for command

IO0CLR = (1 << 16); // RW = 0 for write

IO0SET = (1 << 17); // Enable high

delay\_ms(2); // Short delay

IO0CLR = (1 << 17); // Enable low

}

// Send data to LCD

void lcd\_data(unsigned char data) {

IO1CLR = (0xFF << 16); // Clear data lines P1.16-P1.23

IO1SET = (data << 16); // Send data to data lines

IO1SET = (1 << 24); // RS = 1 for data

IO0CLR = (1 << 16); // RW = 0 for write

IO0SET = (1 << 17); // Enable high

delay\_ms(2); // Short delay

IO0CLR = (1 << 17); // Enable low

}

// Initialize LCD

void lcd\_init() {

IO1DIR |= (0xFF << 16); // Configure P1.16-P1.23 as output (Data lines)

IO1DIR |= (1 << 24); // Configure P1.24 as output (RS)

IO0DIR |= (1 << 16); // Configure P0.16 as output (RW)

IO0DIR |= (1 << 17); // Configure P0.17 as output (EN)

lcd\_command(0x38); // 8-bit mode, 2 lines, 5x7 font

lcd\_command(0x0C); // Display ON, cursor OFF

lcd\_command(0x06); // Entry mode

lcd\_command(0x01); // Clear display

delay\_ms(2); // Wait for LCD to clear

}

// Display time on LCD

void display\_time(unsigned char hours, unsigned char minutes, unsigned char seconds) {

lcd\_command(0x80); // Move cursor to the beginning of the first line

lcd\_data('0' + hours / 10); // Display hours tens

lcd\_data('0' + hours % 10); // Display hours units

lcd\_data(':');

lcd\_data('0' + minutes / 10); // Display minutes tens

lcd\_data('0' + minutes % 10); // Display minutes units

lcd\_data(':');

lcd\_data('0' + seconds / 10); // Display seconds tens

lcd\_data('0' + seconds % 10); // Display seconds units

}

// Initialize RTC

void rtc\_init() {

CCR = 0x02; // Reset and disable RTC

PREINT = 0x1C8; // Set prescaler integer for 32.768 kHz clock

PREFRAC = 0x61C0; // Set prescaler fraction

SEC = 48; // Set initial seconds to 0

MIN = 14; // Set initial minutes to 0

HOUR = 9; // Set initial hours to 12 (you can change this)

CCR = 0x01; // Enable RTC

}

int main() {

unsigned char hours, minutes, seconds;

lcd\_init(); // Initialize LCD

rtc\_init(); // Initialize RTC

while (1) {

// Read current time from RTC

hours = HOUR;

minutes = MIN;

seconds = SEC;

// Display the current time on the LCD

display\_time(hours, minutes, seconds);

delay\_ms(500); // Update every 500 ms

}

return 0;

}

OUTPUT:

