## Arduino Environment

### Hub

Designed for Arduino Due.

The sketch uses the Scheduler library, that runs on Arduino Due only.

In order to use this sketch on another Arduino replace the Scheduler library with another timer/RTOS enabling the multitasking.

#### Before compiling and uploading the sketch

Set the Serial buffer to 2048 In the Arduino IDE on Arduino DUE

File path \hardware\arduino\sam\cores\arduino\RingBuffer.h

Change #define SERIAL\_BUFFER\_SIZE 2048

Needed libraries:

Xbee

Timer

Scheduler

Choose one:

Set *#define S1* if you use Xbee 1 series

Set *#define S2* if you use Xbee 2 series

#### Commands

The following commands can be send on Arduino serial.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Format | Purpose | Example |
| CR0 | CR0 | Receive sensor data | CR0  Return: sensors data |
| CR1 | CR1 | Receive node data | CR1  Return: nodes data |
| CR2 | CR2 | Receive Xbee addresses data | CR2  Return: Xbee addresses data |
| CR3 | CR3 | Receive actuator data | CR3  Return: actuators data |
| CR5 | CR5 | Receive all data | CR5  Return: nodes,sensors,actuators,methods data |
| IN | INn,p1,p2,p3,p4,p5…  n=#of data sent (# of nodes\*5)  p1=node,p2,p3,p4,p5=0 | Set the nodes list | IN10,1,0,0,0,0,2,0,0,0,0  Return: success CX1, error CX0 |
| IX | IXn,n,high,low  n=# of addresses  n=node  high=Xbee high address  low=Xbee low address | Set the Xbee addresses list | IX1,1,1286656,1085167843  Return: success CX1, error CX0 |
| IS | ISn,p1,p2,p3,p4,p5,p6…  n=#of data sent (# of sensors\*6)  p1=node  p2=sensor  p3,p4,p5=0  p6=1 | Set the sensors list | IS12,1,18,0,0,0,1,19,0,0,0,1  Return: success CX1, error CX0 |
| IA | IAn,p1,p2,p3,p4,p5…  n=#of data sent (# of actuators\*5)  p1=node  p2=actuator  p3,p4,p5=0 | Set the actuators list | IA10,2,4,0,0,0,2,5,0,0,0  Return: success CX1, error CX0 |
| IM | IMn,p1,p2,p3,p4,p5…  n=#of data sent (# of actuators\*5)  p1=node  p2=actuator pin  p3=method  p4=0,p5=0 | Set the actuators list | IA10,9,0,31,0,0,2,5,32,0,0  Return: success CX1, error CX0 |
|  |  |  |  |
| CW | CWnode,actuator,value,output,fading,timer  CWnode,actuator,0,output,0,0  CWnode,actuator,value,output,fading,timer | Set trigger (0)  Reset trigger (1)  Set an actuator in a remote node (2) | CW3,5,1,0,5,5  CW2,6,0,0,0,0  CW2,6,255,1,0,0  CW2,6,1,0,0,10  Return: success CX1, error CX0 |
| CA | CAnode,sensor,alarm,actuator,-999  CAnode,sensor,trigger type,actuator,min,status,value,max,output,action of range,action in range,timer  CAnode,sensor,0,0,0,0,0,0,0,0,0,0 | Reset remote trigger (3)  Set a remote trigger (4)  Delete remote trigger (5) | CW2,24,0,11,-999  CA5,18,0,7,100,0,400,0,0,1,0  CA2,21, 0,0,0,0,0,0,0,0,0,0  Return: success CX1, error CX0 |
| CT | CTnode, day, month, year, hour, minutes | Set remote node time. (only for nodes with RTC) | CT2,31,5,2017,13,46  Return: success CX1, error CX0 |
| CM | CMnode, temp, ext hum, ext press, meteo\*  Meteo codes\*  71 Chance Rain  70 Chance of Flurries  70 Chance of Freezing Rain  71 Chance of Rain  70 Chance of Sleet  69 Chance of Snow  72 Chance of Thunderstorms  72 Chance of a Thunderstorm  57 Clear  61 Cloudy  65 Flurries  68 Fog  65 Freezing Rain  67 Haze  60 Mostly Cloudy  58 Mostly Sunny  62 Overcast  58 Partly Cloudy  59 Partly Sunny  63 Rain  58 Scattered Clouds  65 Sleet  66 Snow  57 Sunny  64 Thunderstorm  64 Thunderstorms  56 Unknown | set current meteo | CM2,12.4,57,1024,2 |
| CF | CFnode, forecast\* day1, forecast\* day2, forecast\* day3, forecast\* day4  Forecast codes are the same of meteo | set meteo forecast | CF2,71,65,60,66 |
| CG | CGnode, tmin1, tmax1, tmin2, tmax2, tmin3, tmax3, tmin4, tmax4  All integers values | set meteo forecast temperature | CG2,10,20,11,21,12,22,13,23 |
| ID0 |  | Disable debug | ID0  Return: success CX1, error CX0 |
|  |  |  |  |
| ID1 |  | Enable debug | ID1  Return: success CX1, error CX0 |
|  |  |  |  |

### End Nodes

#### Generic

Designed for Sparkfun Fio V3 or Arduino Fio.

Needed libraries:

Xbee

Timer

DHTlib

IRremote

ArduMedeaWrapper

LEDFader

SimpleSoftwareServo

##### Before compiling and uploading the sketch

###### Setup compiler

\library\IRRemote\IRRemoteInt.h set -> #define IR\_USE\_TIMER3 in the AVR\_ATmega32U4 section

\hardware\arduino\avr\cores\Tone.cpp set -> #define USE\_TIMER1

\hardware\arduino\libraries\Servo\src\avr\ServoTimers.h set -> #define \_useTimer3

Both servo and IR remote use the Timer 3. In order to avoid conflict choose which one you use.

Sparkfun Fio V3 has 3 Timers 1,3,5 but only the first two are available

Timer 1 is used by the Buzzer (Tone)

Timer 3 is used by Servo or IR Remote

If the Tone is not used Servo and IR Remote can be used concurrently.

comment this if you compile to use a servo, leave this if you use IR remote control.

#define REMOTECONTROL\_MODE 1;

##### Hardware configuration

Sensors: pins 18 (A0), 19 (A1), 20 (A3), 21 (A4), 22 (A5), 23 (A6)

virtual pins 24,25 (used for dht values, temperature and humidity)

Actuators pin D4 dht (dedicated) (input)

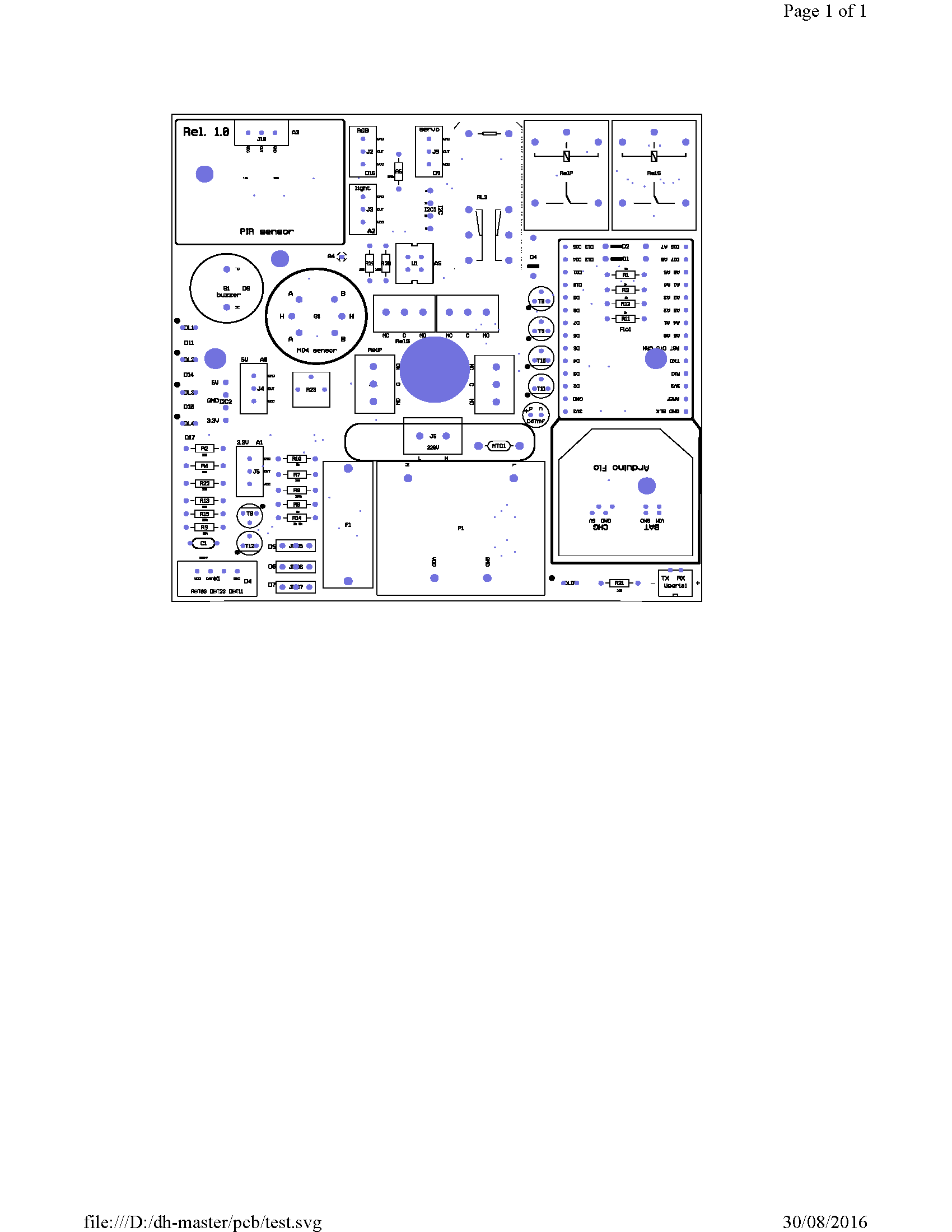
pin D5 used for ir remote (output)

pin D8 tone (dedicated) (output)

pin D6,D7 available (output)

pin D9 servo (output)

pin D10,D11,D14 relays (output)



#### Thermostat

Designed for Arduino Mini Pro

Needed libraries:

<XBee.h>

<Timer.h>

<SoftwareSerial.h>

<Nextion.h>

<EEPROM.h>

<pitches.h>

<Wire.h>

<RTClib.h>

<AM2322.h>

<OneWire.h>

##### Before compiling and uploading the sketch

###### Setup compiler

No specific setup required

##### Hardware configuration

Touch screen: Nextion 2.4” touch screen (NX3224T024)

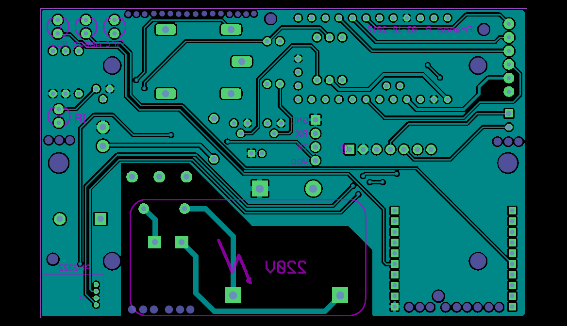
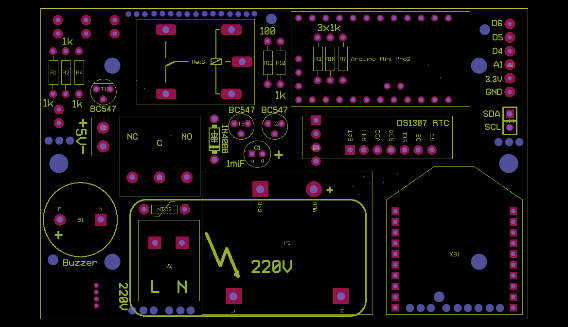
RTC: DS1307

Sensors: AM2322 temperature and humidity sensor

DS18B20 temperature sensor (optional)

Actuators pin D8 Relay

pin D9 Buzzer

#### Switch