VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

(Vellore Campus)

School of Computer Science and Engineering (SCOPE)

Remote Home Control and Monitoring using Zigbee Wireless Sensor Network

Project Under

Professor UMADEVI K S

Team Details

19BCT0081	A N Loganathan
19BCT0149	B Kushwanth Sai
19BCT0158	Dhanesh Shetty

Fall Semester 2021-2022 Wireless and Sensor Networks A1+TA1

TABLE OF CONTENTS

Abstract:

In this project we see a complete demonstration of how a remote control and monitoring system works.We have shown use of microcontroller(Arduino here) for sending data and processing data from sensors to take necessary action.The communication between modules takes place with the help of Xbee(Zigbee module). We have demonstrated the working and energy consumption by zigbee by simulating 802.15.4 which is the basis for the zigbee stack with the help of ns-2 simulation with multiple nodes configured to use various combinations of protocols in various layers of OSI model.

Introduction:

Wireless networks are becoming more important in our day to day life as the world becomes more connected – from controlling remote video systems to monitoring equipment status, from operating sensor systems to communicating with smart devices, these nearly ubiquitous networks enable people to communicate, acquire information, and connect with others. The Internet of Things (IoT), in particular, has lately gained interest as a means of interconnecting billions of objects and collecting and exchanging data.

Usually, there are thousands of sensors in a factory, such as pressure transmitters, flowmeters, temperature transmitters, level transmitters, and so on. Until now, wired networks are used to connect sensors to transfer sampled process data to control systems. Wired networks are very reliable and stable communication systems for instruments and controls. However, the cabling engineering necessary is very costly. Therefore, recently costless wireless networks are more and more strongly required by customers. Many domestic and foreign companies and research institutions are starting to study how to formulate industrial wireless measure and control systems. At present wifi and

bluetooth technologies are popularly used wireless networks in many companies and industries. Even though these wireless networks still have drawbacks in terms of energy consumption and bandwidth utilization. Here comes the zigbee to overcome those drawbacks. ZigBee, the short-range wireless data transmission technology, for its safe, reliable, simple and flexible, low cost, long battery life and other advantages, has shown strong potential and been the research hotspot in recent years in the industrial control field.

Having proposed ZigBee as the wireless technology to replace wire harnesses in any sensing and control application that uses wired signal and data communication as well as to overcome drawbacks of existing wireless networks. The ZigBee (IEEE 802.15.4) is a new technology that permits the implementation of Wireless Personal Area Networks (WPAN). It is very suitable for wireless sensor networks due to the very low power consumption. With industry standardization comes more product innovation, and the emergence of the ZigBee standard is bound to drive the whole market forward.

Literature survey:

Zigbee Based Home Automation

Jitendra Rana (MGM's JNEC Dr. Babasaheb Ambedkar Marathwada University)

ZigBee Based Home Automation Wireless Sensor Network is useful project for adults and physically handicapped persons who are not able to do various activities efficiently when they are at home and need one assistant to perform those tasks. With the ZigBee network we can eliminate the complication of wiring in case of wired automation. ZigBee Home Automation provides operating range much higher as compared to Bluetooth. With the use of ZigBee Home Automation circuit a considerable amount of power saving is possible and it is flexible and compatible with future technologies so it can be

easily customized for individual requirements.

A ZigBee-Based Home Automation System

Shuang-Hua H Yang (Loughborough University) and Fang Yao (University of Surrey)

The use of ZigBee communications technology helps lower the expense of the system and the intrusiveness of the respective system installation. The incorporation of the virtual home concept coordinates the systems security and safety efforts in a clear and consistent manner. The inclusion of a home gateway helps overcome the problems of network interoperability.

IEEE 802.15.4 Ad Hoc Wireless Sensor Networks Simulation Approach

(Wilson T.H. Woon and T.C. Wan)

IEEE 802.15.4 is suitable for simple and non-complex applications. It has the potential of supporting high-end applications but requires extensive enhancements. We proposed the AD-WSN network paradigm as an extension to the standard as well as slot-based (mentioned slightly) channel access to solve the 'hidden node' problem. In a multi-hop environment the end-to-end throughput utilization saturates at 70 kbps traffic load and delivers less than 50% of information.

Features of 802.15.4:

- Data rates of 250 kbps, 40 kbps, and 20 kbps. Symbol rate is 62.5 k symbol/sppm.
- Two addressing modes; 16-bits short (short address) and 64-bit IEEE addressing (long address).
- Optional use Star-topology or Peer to Peer topology, and also supposes Cluster Tree nowadays.

- CSMA-CA channel access
- Automatic network establishment by the coordinator.
- Full handshake protocol for transfer reliability.
- Power management to ensure low power consumption.
- 16 channels in the 2.4GHz ISM band, 10 channels in the 915MHz and one channel in the 868MHz band.
- RSSI (Received signal strength indication) measurement

Proposed Work:

Our project aims to show the complete demonstration of home control and management in all layers.An IoT simulation of Home automation using Zigbee in Proteus software in which we control the appliances remotely and also monitor the sensors using xbee module.We create a NS-2 simulation to show the working of the xbee module,the protocols being used in the form of animation using nam animator and obtain the statistics of the packet delivery using trace file generated.From the statistics we can obtain results about the efficiency,energy consumed,packet delivery ratio, end to end delay of the protocol.

Implementation:

We are using the Proteus software to simulate arduinos,sensors,zigbee modules,appliances like lights and fans. In proteus we created two files,one is the receiver module which is connected to the appliances in a room and other is a sender module or a central hub which sends the instructions to devices and even receives data from the sensors.We have implemented lights,fan and a flame sensor placed in a room connected to a arduino which is in turn connected to xbee.The central hub has a keypad,LCD connected to xbee module via

arduino.On pressing specific keys on the keypad the instruction is sent to the other module using zigbee protocol.The instruction is processed by the arduino using the code we have written and the appropriate lights,fans are switched on/off. When a flame is detected (which can be toggled using an input pin) the flame sensor sends data to the central hub and the LCD displays the warning message.

As zigbee is based on 802.15.4 we have coded a ns2 simulation of 802.15.4. In the simulation we have created 25 nodes. In terms of configurations of ns2 we use a wireless channel using 802.15.4 protocols in physical and MAC layers.Link layer uses logical link layer protocol,using AODV routing protocol and ftp traffic.The nodes are configured to be either a fully functional device(router or coordinator) or reduced functional device(end device).We also specified a energy model to check the energy level after simulation.The positions of the nodes are defined in .scn files.We then send ftp traffic between nodes and it is seen in simulation with the help of colored lines.Different colors show different protocols packets.The trace file is generated from the simulation.The nam file is run to see the visual animation of packets being delivered by multiple hops after path detection.The trace file is analysed with a awk code to give statistics like end to end delay,packet delivery ratio,packets sent,packets received,energy consumed,average hops and total hops.Using the statistics we can comment on suitability of the protocol from home automation.

Arduino Code:

Receiver side:-

```
char arr;
void setup() {
Serial.begin(9600);
pinMode(8,OUTPUT);
pinMode(9,OUTPUT);
pinMode(22,OUTPUT);
pinMode(24,OUTPUT);
pinMode(30,OUTPUT);
pinMode(28,OUTPUT);
pinMode(3,OUTPUT);
pinMode(4,OUTPUT);
pinMode(10,INPUT);
digitalWrite(4,HIGH);
digitalWrite(3,LOW);
digitalWrite(8,HIGH);
digitalWrite(9,LOW);
digitalWrite(22,HIGH);
digitalWrite(24,LOW);
digitalWrite(30,HIGH);
digitalWrite(28,LOW);

}
int high10=0;
void loop() {
  if(Serial.available()>0)
{
arr=Serial.read();
if(arr=='o'){
digitalWrite(8,HIGH);
digitalWrite(9,LOW);
}
if(arr=='p'){
digitalWrite(22,HIGH);
digitalWrite(24,LOW);
Serial.println("bulb 2 on");
}
if(arr=='i'){
digitalWrite(30,HIGH);
digitalWrite(28,LOW);
Serial.println("Bulb 3 is on");
}
if(arr=='u'){
digitalWrite(4,HIGH);
digitalWrite(3,LOW);
Serial.println("Bulb 4 on");
```

```
}

if(arr=='f'){
digitalWrite(8,LOW);
digitalWrite(9,LOW);
digitalWrite(22,LOW);
digitalWrite(24,LOW);
digitalWrite(30,LOW);
digitalWrite(28,LOW);



digitalWrite(22,LOW);
digitalWrite(24,LOW);
digitalWrite(30,LOW);
digitalWrite(28,LOW);
digitalWrite(3,LOW);
digitalWrite(4,LOW);
Serial.println("Bulb off");

}

}
if(digitalRead(10)==1 && high10 ==0){
  Serial.print(1);
  high10 = 1;
}
else if(digitalRead(10)==0 && high10 == 1){
  Serial.print(0);
  high10 = 0;
}
}
```

Sender side:-

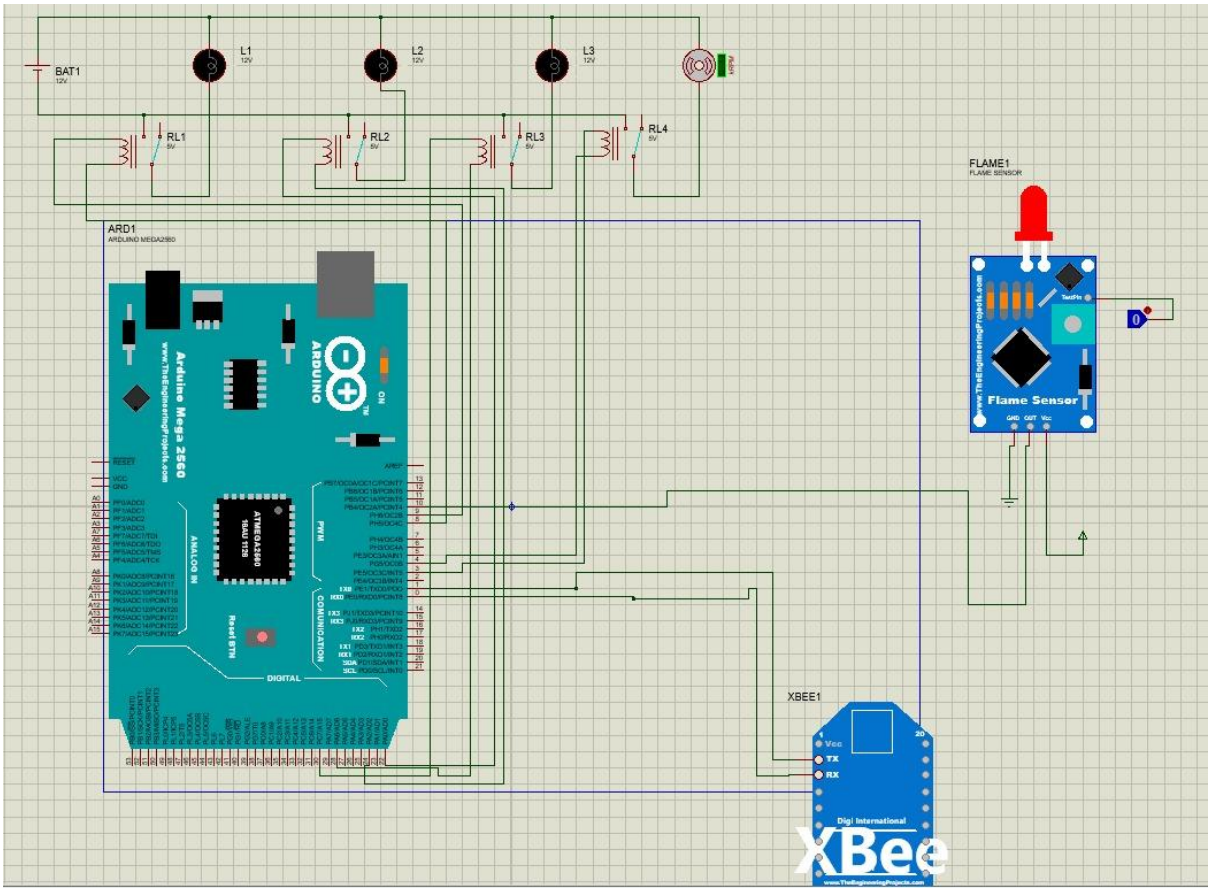
```
#include <Keypad.h>
#include<LiquidCrystal.h>
LiquidCrystal lcd(A0,A1,A2,A3,A4,A5);


const byte ROWS = 4; //four rows
const byte COLS = 3; //three columns
char keys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
```

```
  {'#','0','*'}
};
byte rowPins[ROWS] = {2,3,4,5}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {6,7,8}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup(){
  Serial.begin(9600);
  for(int i=2;i<9;i++){
    pinMode(i,OUTPUT);
  }
  pinMode(10,OUTPUT);
  digitalWrite(10,HIGH);
      lcd.setCursor(0,0);
      digitalWrite(10,LOW);
      lcd.print("Flame NOT DETECTED");
}

void loop(){
  char key = keypad.getKey();

  if (key != NO_KEY){
    if(key=='1'){
      Serial.print('o');
    }
    if(key=='2'){
      Serial.print('p');
    }
    if(key=='3'){
      Serial.print('i');
    }
    if(key=='4'){
      Serial.print('u');
    }
    if(key=='5'){
      Serial.print('f');
    }
    if(key=='6'){
      Serial.print('q');
    }
  }

  if(Serial.available()>0){
    int k = Serial.read();
    if(k == '1'){
      digitalWrite(10,HIGH);
      lcd.setCursor(0,0);
      lcd.print("Flame DETECTED");
    }
```

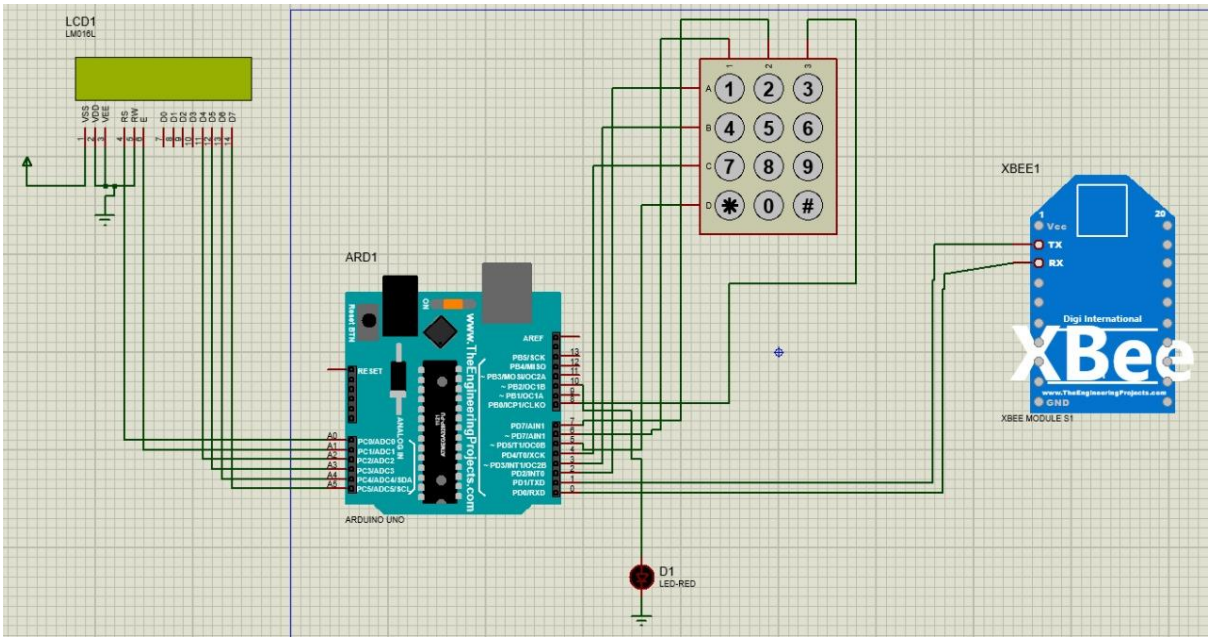
```
    else if(k == '0'){
      lcd.setCursor(0,0);
      digitalWrite(10,LOW);
      lcd.print("Flame NOT DETECTED");
    }
  }
}
```

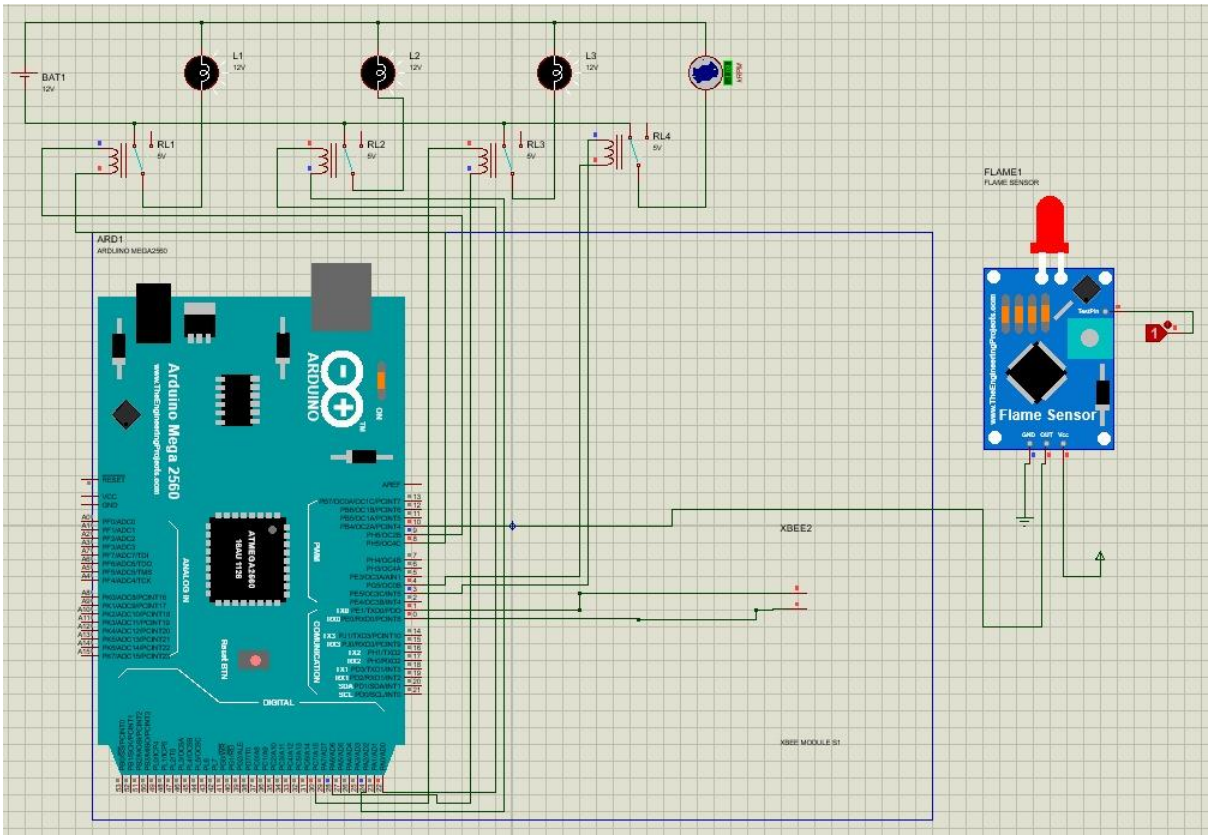
Proteus Screenshots:

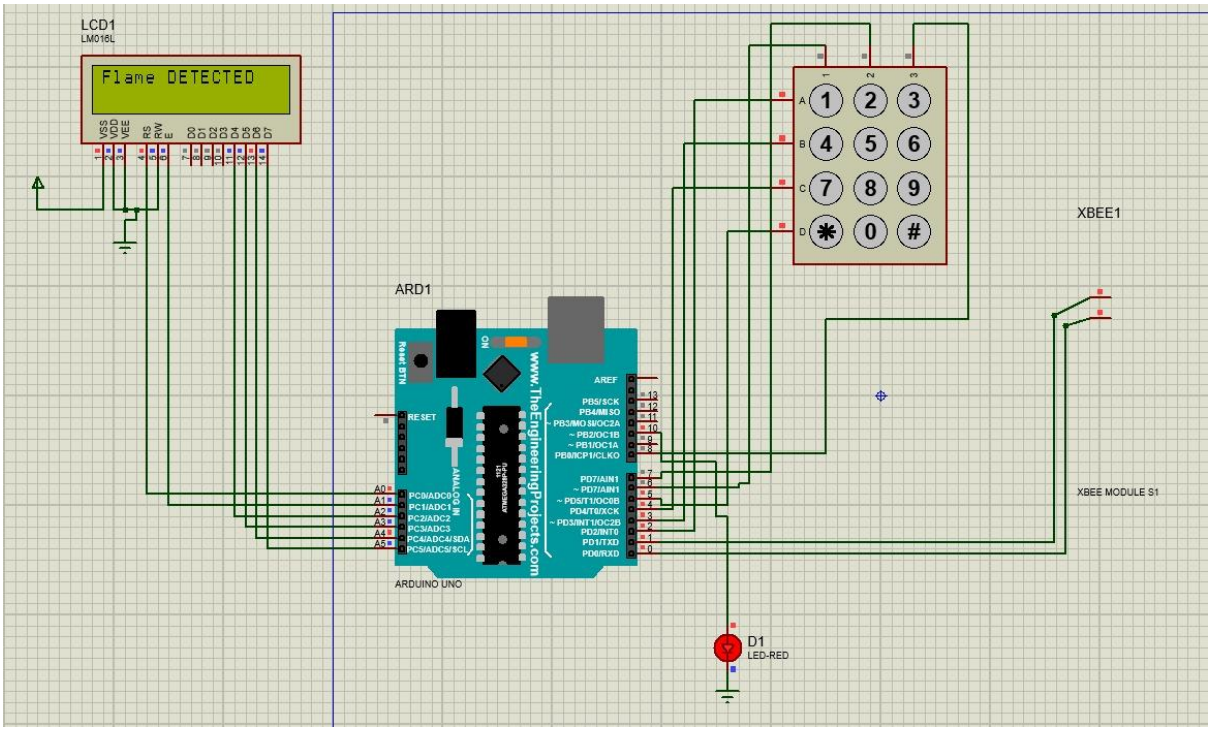
Receiver



Transmitter / Sender / Controller







How does Xbee Module work?

NS-2 code:
tcl file

```
set val(chan)          Channel/WirelessChannel    ;# Channel Type
 set val(prop)          Propagation/TwoRayGround   ;# radio-propagation model
 set val(netif)         Phy/WirelessPhy/802_15_4
 set val(mac)           Mac/802_15_4
 set val(ifq)          Queue/DropTail/PriQueue    ;# interface queue type
 set val(ll)           LL                         ;# link layer type
 set val(ant)          Antenna/OmniAntenna        ;# antenna model
 set val(ifqlen)        50                         ;# max packet in ifq
 set val(nn)           25                         ;# number of mobilenodes
 set val(rp)           AODV                       ;# routing protocol
 set val(x)                 50
 set val(y)                 50

 set val(nam)               zigbee_nam.nam
 set val(traffic)   ftp                    ;# ftp


 set appTime1           30.0          ;# in seconds
 set appTime2           30.2          ;# in seconds
 set appTime3           30.3          ;# in seconds
 set stopTime          50  ;# in seconds


 set ns_            [new Simulator]
```

```
set tracefd     [open ./zigbee_trace.tr w]

$ns_ trace-all $tracefd
if { "$val(nam)" == "zigbee_nam.nam" } {
        set namtrace     [open ./$val(nam) w]
        $ns_ namtrace-all-wireless $namtrace $val(x) $val(y)
}
$ns_ use-newtrace
$ns_ puts-nam-traceall {# nam4wpan #}
Mac/802_15_4 wpanNam namStatus on


# For model 'TwoRayGround'
set dist(5m)  7.69113e-06
set dist(9m)  2.37381e-06
set dist(10m) 1.92278e-06
set dist(11m) 1.58908e-06
set dist(12m) 1.33527e-06
set dist(13m) 1.13774e-06
set dist(14m) 9.81011e-07
set dist(15m) 8.54570e-07
set dist(16m) 7.51087e-07
set dist(20m) 4.80696e-07
set dist(25m) 3.07645e-07
set dist(30m) 2.13643e-07
set dist(35m) 1.56962e-07
set dist(40m) 1.20174e-07
Phy/WirelessPhy set CSThresh_ $dist(15m)
Phy/WirelessPhy set RXThresh_ $dist(15m)


set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)


set god_ [create-god $val(nn)]

set chan_1_ [new $val(chan)]



$ns_ node-config -adhocRouting $val(rp) \
                 -llType $val(ll) \
                 -macType $val(mac) \
                 -ifqType $val(ifq) \
                 -ifqLen $val(ifqlen) \
                 -antType $val(ant) \
                 -propType $val(prop) \
                 -phyType $val(netif) \
                 -topoInstance $topo \
                 -agentTrace ON \
                 -routerTrace ON \
                 -macTrace ON \
                 -movementTrace OFF \
```

```
      -energyModel "EnergyModel" \
      -initialEnergy 1 \
      -rxPower 0.3 \
      -txPower 0.3 \
                  -channel $chan_1_

for {set i 0} {$i < $val(nn) } {incr i} {
          set node_($i) [$ns_ node]
          $node_($i) random-motion 0                ;
}

source ./topology.scn
$ns_ at 0.0 "$node_(23) NodeLabel PAN Coor";
$ns_ at 0.0 "$node_(23) sscs startPANCoord";
set curr 0.0
for {set i 0} {$i < $val(nn) } {incr i} {
  if {$i!=23} {
   if {$i==2 || $i==7 || $i==24 || $i==6 || $i==19 || $i==12 || $i==5 || $i==21  } {
     $ns_ at $curr+0.5 "$node_($i) sscs startDevice 1";
  } else {
     $ns_ at $curr+0.5 "$node_($i) sscs startDevice 0";
  }
  }

}


proc ftptraffic { src dst starttime } {
  global ns_ node_
  set tcp($src) [new Agent/TCP]
  eval \$tcp($src) set packetSize_ 60
  set sink($dst) [new Agent/TCPSink]
  eval $ns_ attach-agent \$node_($src) \$tcp($src)
  eval $ns_ attach-agent \$node_($dst) \$sink($dst)
  eval $ns_ connect \$tcp($src) \$sink($dst)
  set ftp($src) [new Application/FTP]
  eval \$ftp($src) attach-agent \$tcp($src)
  $ns_ at $starttime "$ftp($src) start"
}

if { "$val(traffic)" == "ftp" } {
  puts "\nTraffic: ftp"
  #Mac/802_15_4 wpanCmd ack4data off
  set lowSpeed 0.2ms
  set highSpeed 0.5ms
  Mac/802_15_4 wpanNam PlaybackRate $lowSpeed
  $ns_ at [expr $appTime1+0.2] "Mac/802_15_4 wpanNam PlaybackRate $highSpeed"
  $ns_ at $appTime2 "Mac/802_15_4 wpanNam PlaybackRate $lowSpeed"
  $ns_ at [expr $appTime2+0.2] "Mac/802_15_4 wpanNam PlaybackRate $highSpeed"
  $ns_ at $appTime3 "Mac/802_15_4 wpanNam PlaybackRate $lowSpeed"
  $ns_ at [expr $appTime3+0.2] "Mac/802_15_4 wpanNam PlaybackRate 1ms"
  ftptraffic 23 15 $appTime1
  ftptraffic 11 8 $appTime2
  ftptraffic 1 23 $appTime3
```

```
    Mac/802_15_4 wpanNam FlowClr -p AODV -c tomato
    Mac/802_15_4 wpanNam FlowClr -p ARP -c green
    Mac/802_15_4 wpanNam FlowClr -p tcp -s 23 -d 15 -c blue
    Mac/802_15_4 wpanNam FlowClr -p ack -s 15 -d 23 -c blue
    Mac/802_15_4 wpanNam FlowClr -p tcp -s 11 -d 8 -c green4
    Mac/802_15_4 wpanNam FlowClr -p ack -s 8 -d 11 -c green4
    Mac/802_15_4 wpanNam FlowClr -p tcp -s 1 -d 23 -c cyan4
    Mac/802_15_4 wpanNam FlowClr -p ack -s 23 -d 1 -c cyan4
    $ns_ at $appTime1 "$node_(23) NodeClr blue"
    $ns_ at $appTime1 "$node_(15) NodeClr blue"
    $ns_ at $appTime1 "$ns_ trace-annotate \"(at $appTime1) ftp traffic from node 23 to node 15\""
    $ns_ at $appTime2 "$node_(11) NodeClr blue"
    $ns_ at $appTime2 "$node_(8) NodeClr blue"
    $ns_ at $appTime2 "$ns_ trace-annotate \"(at $appTime2) ftp traffic from node 11 to node 8\""
    $ns_ at $appTime3 "$node_(1) NodeClr blue"
    $ns_ at $appTime3 "$node_(23) NodeClr blue"
    $ns_ at $appTime3 "$ns_ trace-annotate \"(at $appTime3) ftp traffic from node 1 to node 23\""
}

 for {set i 0} {$i < $val(nn)} {incr i} {
            $ns_ initial_node_pos $node_($i) 2
 }

 for {set i 0} {$i < $val(nn) } {incr i} {
    $ns_ at $stopTime "$node_($i) reset";
 }

 $ns_ at $stopTime "stop"
 $ns_ at $stopTime "puts \"\nNS EXITING...\""
 $ns_ at $stopTime "$ns_ halt"

 proc stop {} {
    global ns_ tracefd val env
    $ns_ flush-trace
    close $tracefd
    set hasDISPLAY 0
    foreach index [array names env] {
       #puts "$index: $env($index)"
       if { ("$index" == "DISPLAY") && ("$env($index)" != "") } {
             set hasDISPLAY 1
       }
    }
    if { ("$val(nam)" == "zigbee_nam.nam") && ("$hasDISPLAY" == "1") } {
             exec nam zigbee_nam.nam &
    }
 }

 puts "\nStarting Simulation..."
 $ns_ run
```

topology.scn file

```
#
 # nodes: 25, pause: 2.00, max speed: 2.00  max x = 50.00, max y: 50.00
```

```
#
$node_(0) set X_ 26.608377307314
$node_(0) set Y_ 49.446991827566
$node_(0) set Z_ 0.000000000000
$node_(1) set X_ 49.337311778721
$node_(1) set Y_ 48.582820874924
$node_(1) set Z_ 0.000000000000
$node_(2) set X_ 27.437938215931
$node_(2) set Y_ 35.091658830425
$node_(2) set Z_ 0.000000000000
$node_(3) set X_ 10.568766842667
$node_(3) set Y_ 47.453389414657
$node_(3) set Z_ 0.000000000000
$node_(4) set X_ 29.496269526711
$node_(4) set Y_ 9.257450139045
$node_(4) set Z_ 0.000000000000
$node_(5) set X_ 26.476535726599
$node_(5) set Y_ 41.646396828920
$node_(5) set Z_ 0.000000000000
$node_(6) set X_ 36.430596917301
$node_(6) set Y_ 30.320042376073
$node_(6) set Z_ 0.000000000000
$node_(7) set X_ 20.045665842364
$node_(7) set Y_ 32.537008965783
$node_(7) set Z_ 0.000000000000
$node_(8) set X_ 8.641089465783
$node_(8) set Y_ 18.871428688349
$node_(8) set Z_ 0.000000000000
$node_(9) set X_ 50.125856848941
$node_(9) set Y_ 35.049644538277
$node_(9) set Z_ 0.000000000000


$node_(10) set X_ 50.596871605178
$node_(10) set Y_ 21.816174228621
$node_(10) set Z_ 0.000000000000
$node_(11) set X_ 8.828197591068
$node_(11) set Y_ 34.402175912553
$node_(11) set Z_ 0.000000000000
$node_(12) set X_ 18.759762507870
$node_(12) set Y_ 25.069997591097
$node_(12) set Z_ 0.000000000000
$node_(13) set X_ 13.277017351151
$node_(13) set Y_ 9.522149009306
$node_(13) set Z_ 0.000000000000
$node_(14) set X_ 35.935083056450
$node_(14) set Y_ 45.156742480258
$node_(14) set Z_ 0.000000000000
$node_(15) set X_ 23.507724673396
$node_(15) set Y_ 9.566772846479
$node_(15) set Z_ 0.000000000000
$node_(16) set X_ 20.236128396148
$node_(16) set Y_ 45.484419961171
$node_(16) set Z_ 0.000000000000
```

```
$node_(17) set X_ 45.709636471608
$node_(17) set Y_ 48.353113573382
$node_(17) set Z_ 0.000000000000
$node_(18) set X_ 45.610349971557
$node_(18) set Y_ 49.787732692400
$node_(18) set Z_ 0.000000000000
$node_(19) set X_ 24.345058881159
$node_(19) set Y_ 19.614188426582
$node_(19) set Z_ 0.000000000000
$node_(20) set X_ 35.972132273531
$node_(20) set Y_ 10.620805728046
$node_(20) set Z_ 0.000000000000
$node_(21) set X_ 41.626765649851
$node_(21) set Y_ 40.028905860151
$node_(21) set Z_ 0.000000000000
$node_(22) set X_ 8.914806804685
$node_(22) set Y_ 24.094071765795
$node_(22) set Z_ 0.000000000000
$node_(23) set X_ 28.443020411336
$node_(23) set Y_ 28.367299319338
$node_(23) set Z_ 0.000000000000
$node_(24) set X_ 35.102836431793
$node_(24) set Y_ 22.394995544608
$node_(24) set Z_ 0.000000000000
```

statistics.awk file

```
BEGIN {
print("\n\n******** Network Statistics ********\n");
        energy_left[25];
        packet_sent[25];
        packet_recvd[25];
for(i=0;i<25;i++)
{
        energy_left[i] = 1.000000;
        packet_sent[i] = 0;
        packet_recvd[i] = 0;
}

total_pkt_sent=0;
total_pkt_recvd=0;
pkt_delivery_ratio = 0;
total_hop_count = 0;
avg_hop_count = 0;
start = 0.000000000;
end = 0.000000000;
packet_duration = 0.0000000000;
recvnum = 0;
delay = 0.000000000;
sum = 0.000000000;
i=0;
total_energy_consumed = 0.000000;
}
```

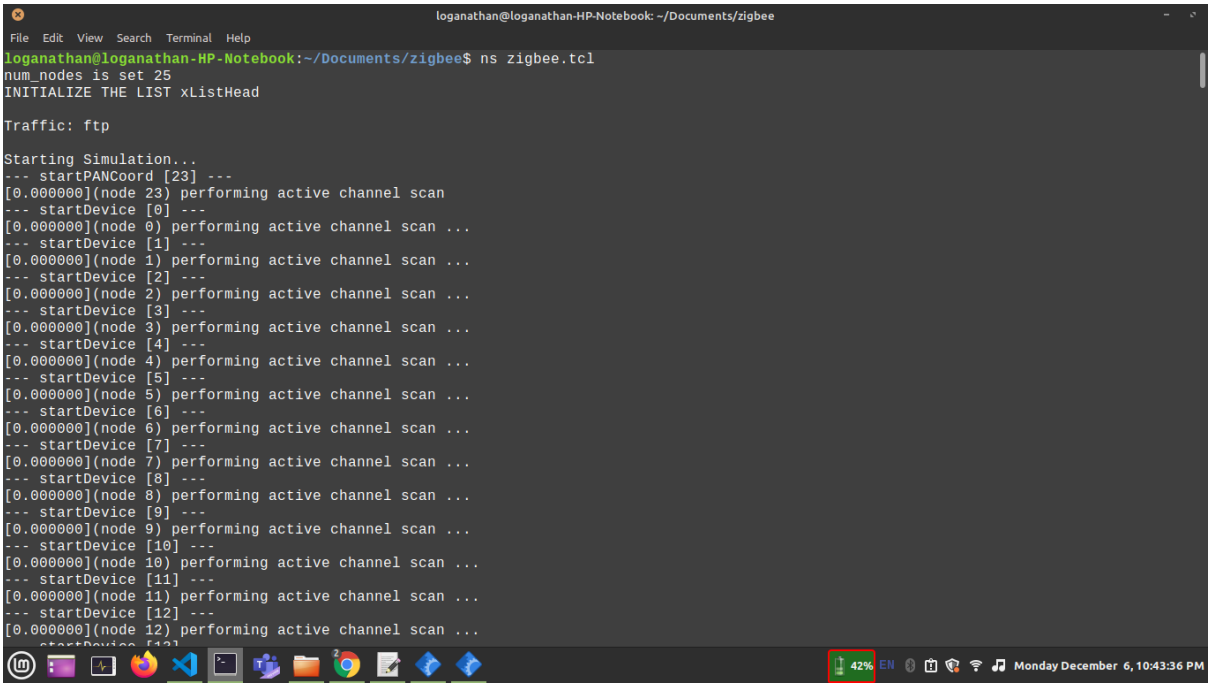
```
{
state           =       $1;
time            =       $3;

node_num        =       $5;
energy_level    =       $7;


node_id         =       $9;
level           =       $19;
pkt_type        =       $35;
packet_id       =       $41;
no_of_forwards          =       $49;


if((state == "s") && (level=="AGT")) {
        for(i=0;i<25;i++) {
                if(i == node_id) {
                packet_sent[i] = packet_sent[i] + 1; }
}
}else if((state == "r") && (level=="AGT")) {
        for(i=0;i<25;i++) {
                if(i == node_id) {
                packet_recvd[i] = packet_recvd[i] + 1; }
}
}

# total hop counts
if ((state == "r") && (level == "RTR") ) { total_hop_count = total_hop_count + no_of_forwards; }

#Average End to End Delay


if (( state == "s") && ( level == "AGT" ))  { start_time[packet_id] = time; }

 if (( state == "r") && ( level == "AGT" )) {  end_time[packet_id] = time;  }
 else {  end_time[packet_id] = -1;  }

#Average Energy Consumption
if(state == "N") {
        for(i=0;i<25;i++) {
                if(i == node_num) {
                                        energy_left[i] = energy_level;
                                }

                        }
}
}


END {
for(i=0;i<25;i++) {
printf("%d %d \n",i, packet_sent[i]) > "pktsent.txt";
printf("%d %d \n",i, packet_recvd[i]) > "pktrecvd.txt";
```

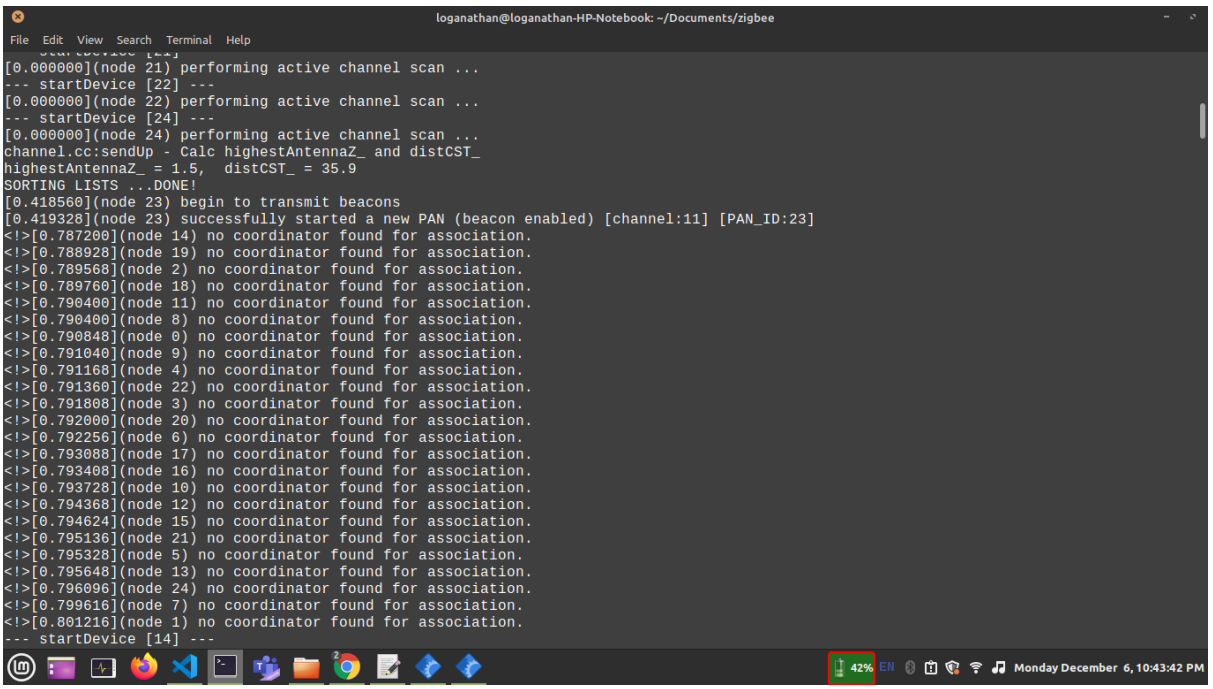
```
printf("%d %.6f \n",i, energy_left[i]) > "energyleft.txt";

total_pkt_sent = total_pkt_sent + packet_sent[i];
total_pkt_recvd = total_pkt_recvd + packet_recvd[i];
total_energy_consumed = total_energy_consumed +(1.000000-energy_left[i]);

}
printf("Total Packets Sent                :        %d\n",total_pkt_sent);
printf("Total Packets Received            :        %d\n",total_pkt_recvd);

pkt_delivery_ratio = (total_pkt_recvd/total_pkt_sent)*100;

printf("Packet Delivery Ratio             :        %.2f%\n",pkt_delivery_ratio);

printf("The total hop counts are  :       %d\n", total_hop_count);

avg_hop_count = total_hop_count/total_pkt_recvd;
printf("Average Hop Count                 :        %d hops\n", avg_hop_count);


#End to End Delay

for ( i in end_time ) {
 start = start_time[i];
 end = end_time[i];
 packet_duration = end - start;
 if ( packet_duration > 0 )  { sum += packet_duration; recvnum++; }
}

delay=sum/recvnum;

printf("Average End to End Delay          :      %.9f ms\n", delay);

printf("Total Energy Consumed                     :      %.6f\n", total_energy_consumed);

printf("Energy Consumption is          :      %.2f%\n",
((total_energy_consumed/(25*1.000000))*100.000000));

}
```

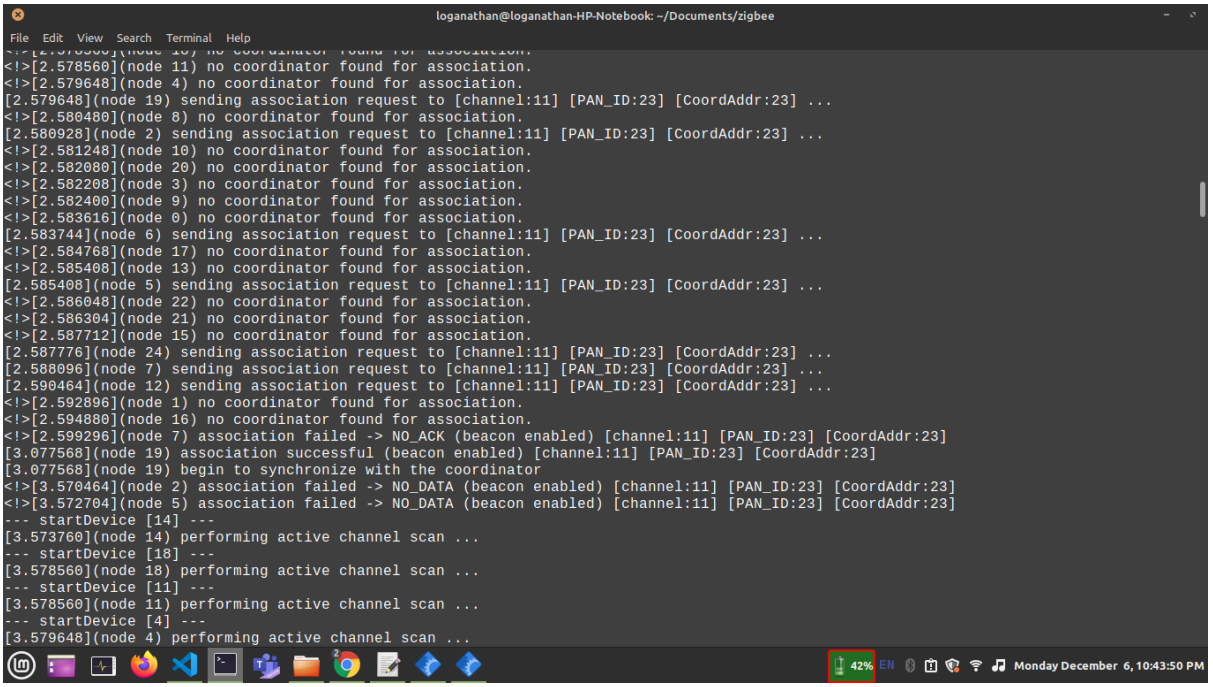
Github Link of our Project :

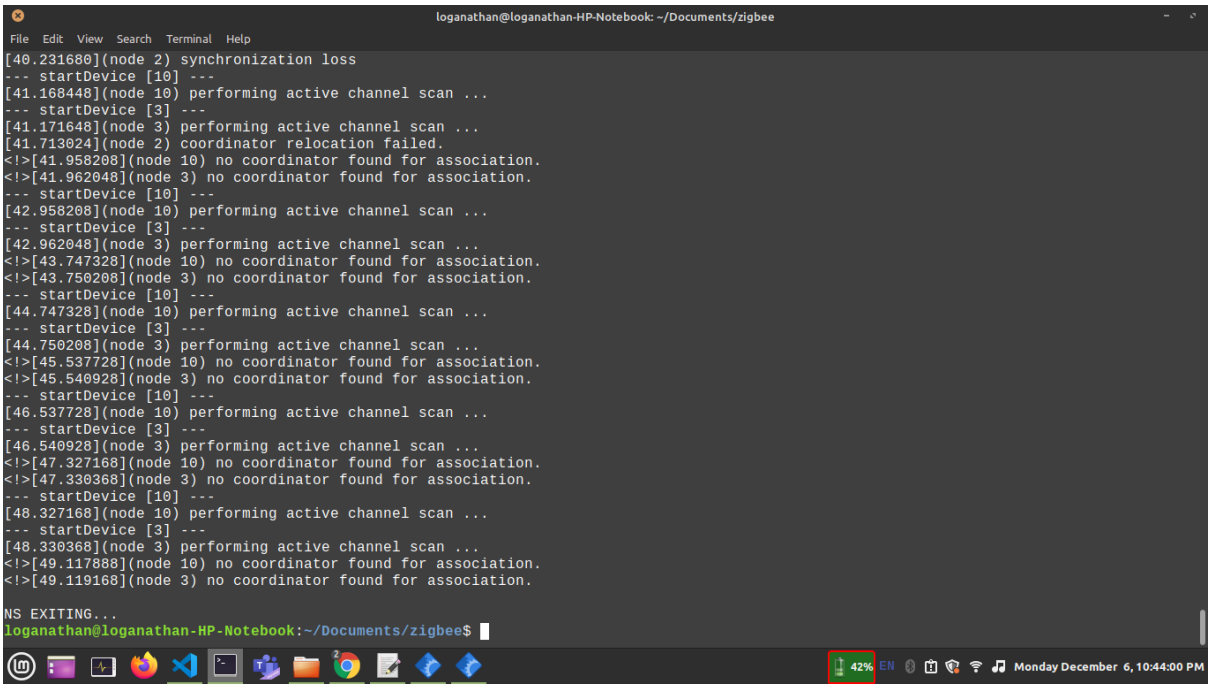
https://github.com/LOGANATHANAN/zigbee-network-simulation

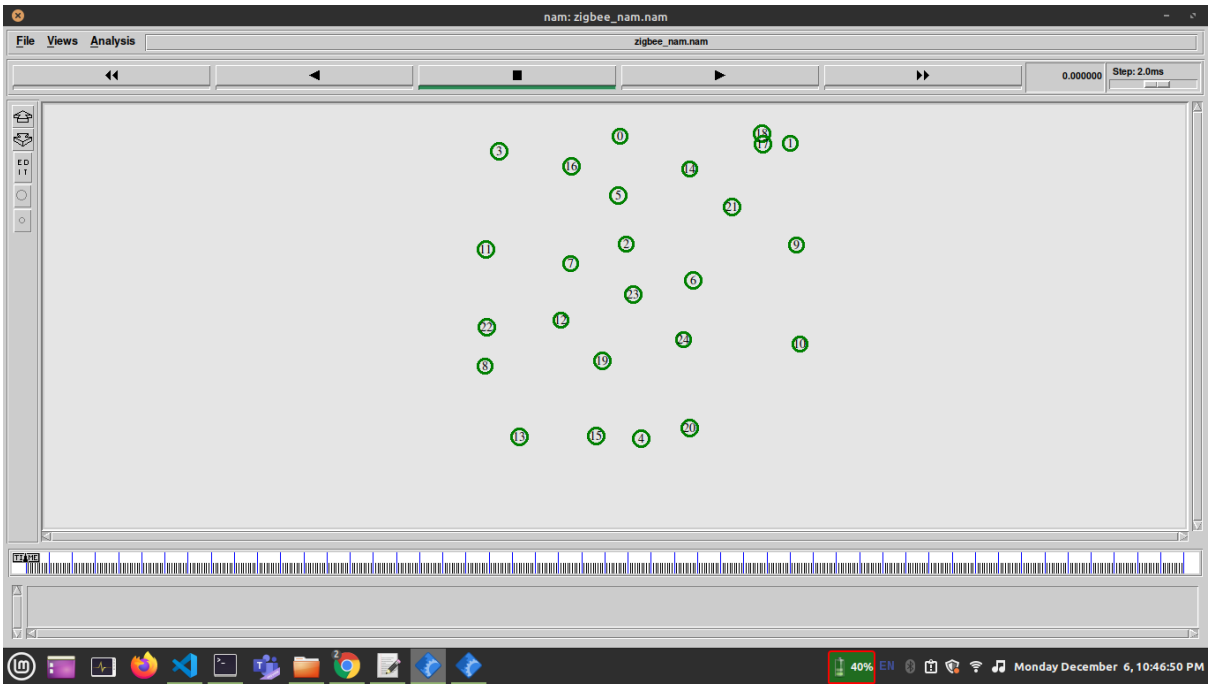
NS-2 simulation and results:

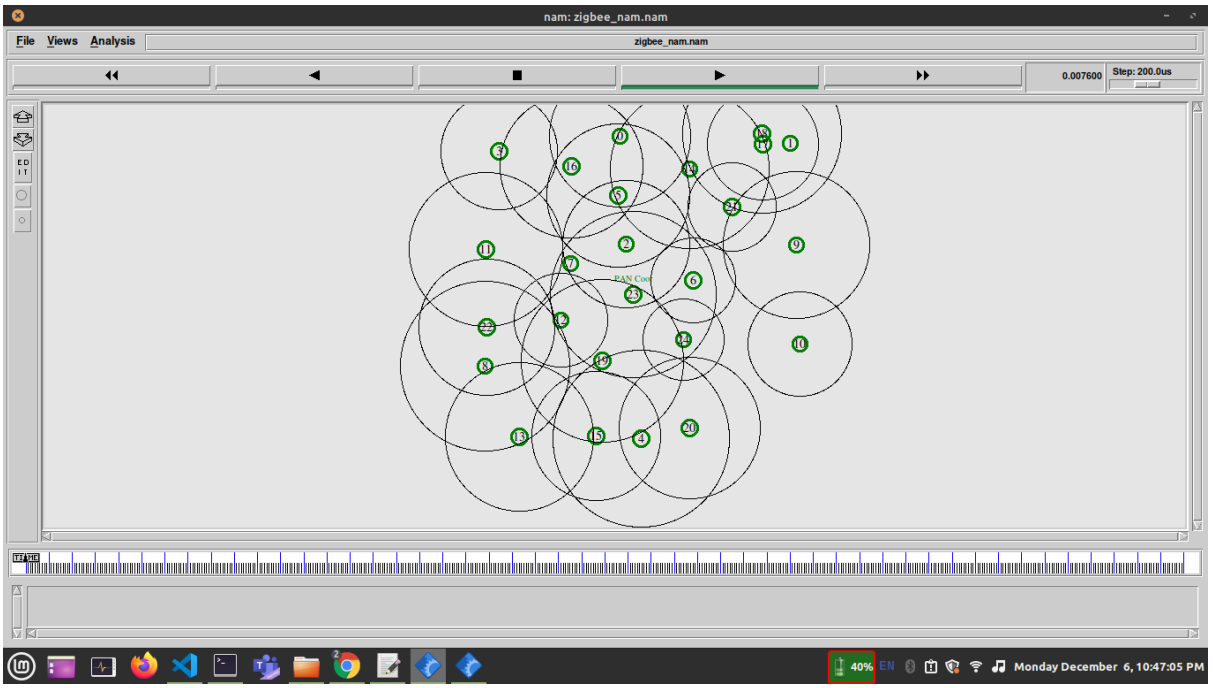


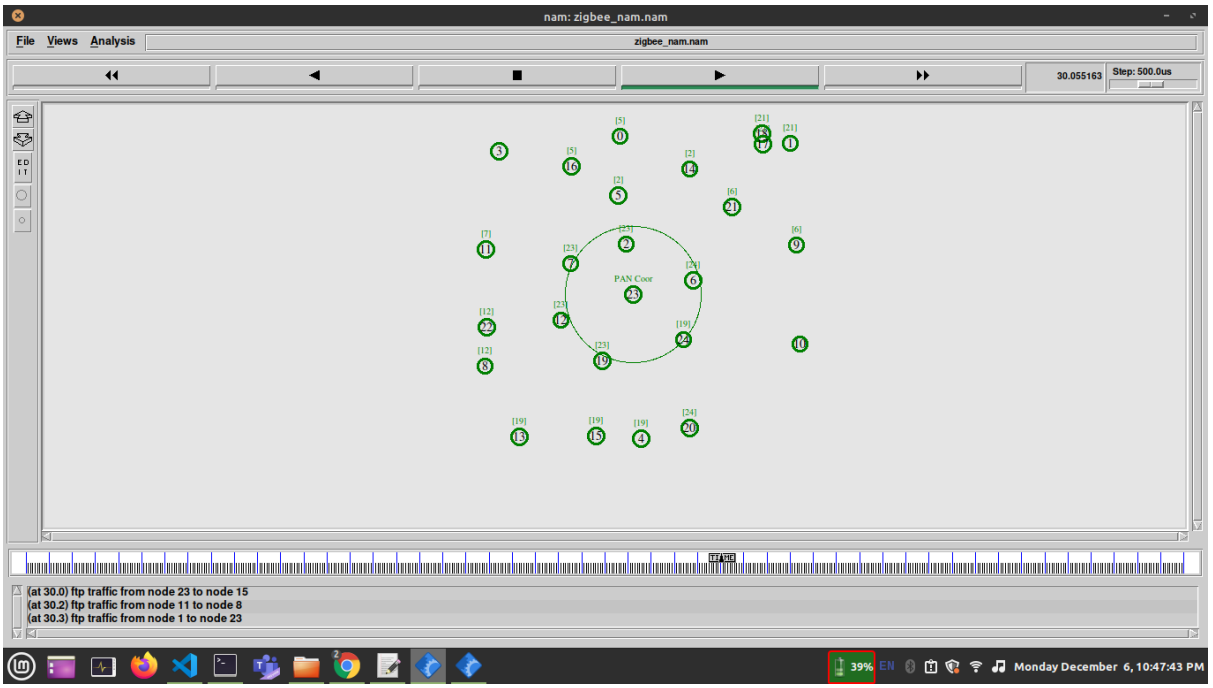



loganathan@loganathan-HP-Notebook: ~/Documents/zigbee
File Edit View Search Terminal Help
<!>[2.578560](node 11) no coordinator found for association.
<!>[2.579648](node 4) no coordinator found for association.
[2.579648](node 19) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
<!>[2.580480](node 8) no coordinator found for association.
[2.580928](node 2) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
<!>[2.581248](node 10) no coordinator found for association.
<!>[2.582080](node 20) no coordinator found for association.
<!>[2.582208](node 3) no coordinator found for association.
<!>[2.582400](node 9) no coordinator found for association.
<!>[2.583616](node 0) no coordinator found for association.
[2.583744](node 6) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
<!>[2.584768](node 17) no coordinator found for association.
<!>[2.585408](node 13) no coordinator found for association.
[2.585408](node 5) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
<!>[2.586048](node 22) no coordinator found for association.
<!>[2.586304](node 21) no coordinator found for association.
<!>[2.587712](node 15) no coordinator found for association.
[2.587776](node 24) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
[2.588096](node 7) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
[2.590464](node 12) sending association request to [channel:11] [PAN_ID:23] [CoordAddr:23] ...
<!>[2.592896](node 1) no coordinator found for association.
<!>[2.594880](node 16) no coordinator found for association.
<!>[2.599296](node 7) association failed -> NO_ACK (beacon enabled) [channel:11] [PAN_ID:23] [CoordAddr:23]
[3.077568](node 19) association successful (beacon enabled) [channel:11] [PAN_ID:23] [CoordAddr:23]
[3.077568](node 19) begin to synchronize with the coordinator
<!>[3.570464](node 2) association failed -> NO_DATA (beacon enabled) [channel:11] [PAN_ID:23] [CoordAddr:23]
<!>[3.572704](node 5) association failed -> NO_DATA (beacon enabled) [channel:11] [PAN_ID:23] [CoordAddr:23]
--- startDevice [14] ---
[3.573760](node 14) performing active channel scan ...
--- startDevice [18] ---
[3.578560](node 18) performing active channel scan ...
--- startDevice [11] ---
[3.578560](node 11) performing active channel scan ...
--- startDevice [4] ---
[3.579648](node 4) performing active channel scan ...
42%
Monday December 6, 10:43:50 PM


loganathan@loganathan-HP-Notebook: ~/Documents/zigbee
File Edit View Search Terminal Help
[40.231680](node 2) synchronization loss
--- startDevice [10] ---
[41.168448](node 10) performing active channel scan ...
--- startDevice [3] ---
[41.171648](node 3) performing active channel scan ...
[41.713024](node 2) coordinator relocation failed.
<!>[41.958208](node 10) no coordinator found for association.
<!>[41.962048](node 3) no coordinator found for association.
--- startDevice [10] ---
[42.958208](node 10) performing active channel scan ...
--- startDevice [3] ---
[42.962048](node 3) performing active channel scan ...
<!>[43.747328](node 10) no coordinator found for association.
<!>[43.750208](node 3) no coordinator found for association.
--- startDevice [10] ---
[44.747328](node 10) performing active channel scan ...
--- startDevice [3] ---
[44.750208](node 3) performing active channel scan ...
<!>[45.537728](node 10) no coordinator found for association.
<!>[45.540928](node 3) no coordinator found for association.
--- startDevice [10] ---
[46.537728](node 10) performing active channel scan ...
--- startDevice [3] ---
[46.540928](node 3) performing active channel scan ...
<!>[47.327168](node 10) no coordinator found for association.
<!>[47.330368](node 3) no coordinator found for association.
--- startDevice [10] ---
[48.327168](node 10) performing active channel scan ...
--- startDevice [3] ---
[48.330368](node 3) performing active channel scan ...
<!>[49.117888](node 10) no coordinator found for association.
<!>[49.119168](node 3) no coordinator found for association.
NS EXITING...
loganathan@loganathan-HP-Notebook:~/Documents/zigbee$
42%
Monday December 6, 10:44:00 PM


nam: zigbee_nam.nam
File Views Analysis
zigbee_nam.nam
0.000000
Step: 2.0ms
Monday December 6, 10:46:50 PM


nam: zigbee_nam.nam
File Views Analysis
zigbee_nam.nam
0.007600
Step: 200.0us
Monday December 6, 10:47:05 PM




```
loganathan@loganathan-HP-Notebook: ~/Documents/zigbee
File  Edit  View  Search  Terminal  Help
<!>[47.330368](node 3) no coordinator found for association.
--- startDevice [10] ---
[48.327168](node 10) performing active channel scan ...
--- startDevice [3] ---
[48.330368](node 3) performing active channel scan ...
<!>[49.117888](node 10) no coordinator found for association.
<!>[49.119168](node 3) no coordinator found for association.

NS EXITING...
loganathan@loganathan-HP-Notebook:~/Documents/zigbee$ gawk -f statistics.awk zigbee_trace.tr


******** Network Statistics ********

Total Packets Sent              :       470
Total Packets Received          :       372
Packet Delivery Ratio           :       79.15%
The total hop counts are        :       7954
Average Hop Count               :       21 hops
Average End to End Delay        :       0.034995701 ms
Total Energy Consumed           :       12.057685
Energy Consumption is           :       48.23%
loganathan@loganathan-HP-Notebook:~/Documents/zigbee$
```

Results & Discussion:

We have simulated the home control and monitoring system containing essential devices in our home in the Proteus software in which all devices communicate with each other using the Xbee module which uses zigbee technology behind the scenes. And to prove that using zigbee technology may increase performance and decrease energy consumption, we simulated 802.15.4 protocol which is the basis for the zigbee tech in ns2. In ns2 we created virtual nodes

each representing a device in our home or office and made them communicate using 802.15.4 stack rather than traditional wifi or bluetooth which we are using at present.

From the statistics from the generated trace logs we can analyze that 802.15.4 protocol is more efficient for personal area networks than traditional wifi and bluetooth technologies. Its features like low energy usage,low data rate, better network connectivity with multi hop routing makes it best suited for IOT and home automation applications.It is also being used as a standard protocol for connecting devices from different manufacturers.

References:

https://www.nsnam.com/2021/10/wireless-networks-inthis-post.html

http://vlabs.iitkgp.ac.in/ant/10/theory/

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.294.7727&rep=rep1&type=pdf