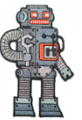# Practical Training 3

SimpliciTI protocol + Project

Practical training 3

- Peer to peer (with AP)
- Address transmission
- Non volatile objects
- Frequency agility

- Project

To keep a Boeing Dreamliner flying, reboot once every 248 days

by Edgar Alvarez | @abcdedgar | May 1st 2015 at 6:34 pm

0

The plane's electrical generators fall into a failsafe mode if kept continuously powered on for 248 days. The 787 has four such main generator-control units that, if powered on at the same time, could fail simultaneously and cause a complete electrical shutdown.
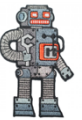
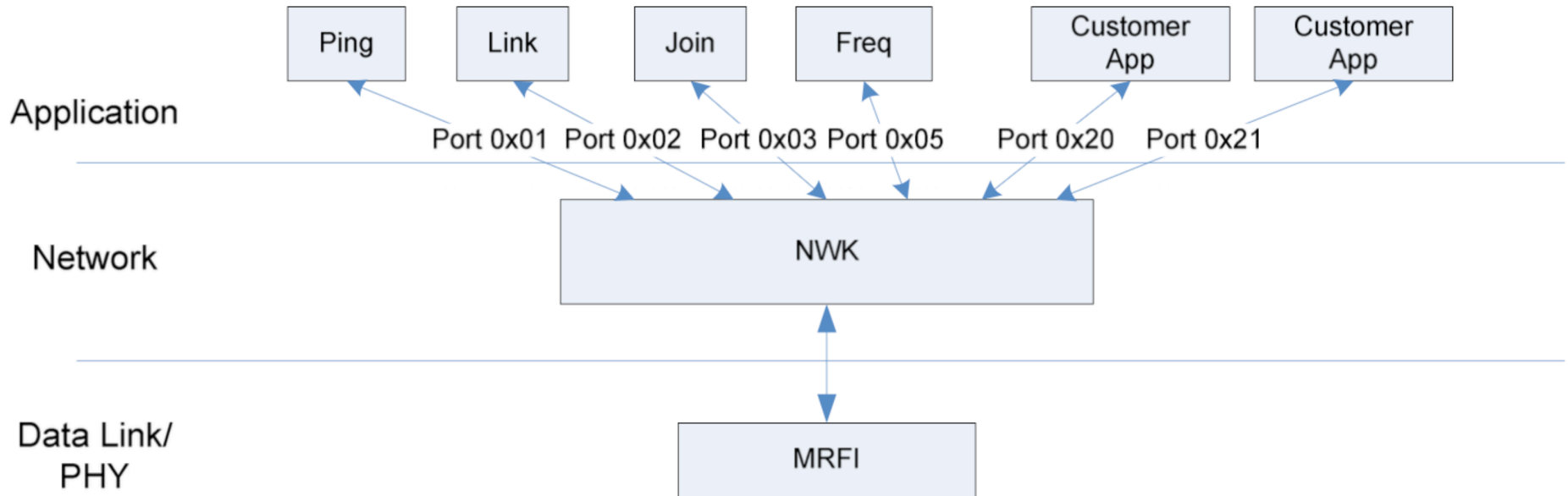LG G4 preview: Fashion and firepower collide in a flagship

SimpliciTI

RF protocol for MSP430 microcontrollers

- handling network traffic
- routing/forwarding messages
- address handling
- message acknowledge
- data encryption
- CRC
- data whitening
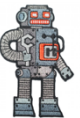
SimpliciTI Layer
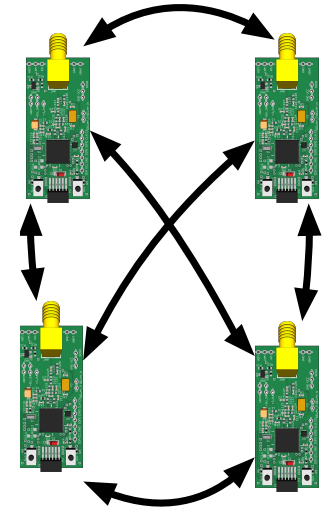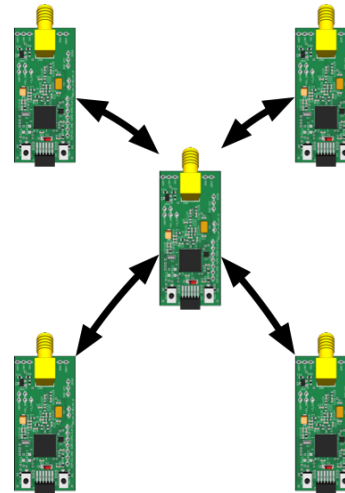
SimpliciTI

- ## Application Layer:
  - Layer for the user, no info about the actual Hardware needed

- ## Network Layer:
  - Logical Addressing (IP address), msg routing, format data in packets

- ## Data Link/PHY Layer:
  - Physical connection (electrical/RF signals) between peers (Hardware addressing MAC)
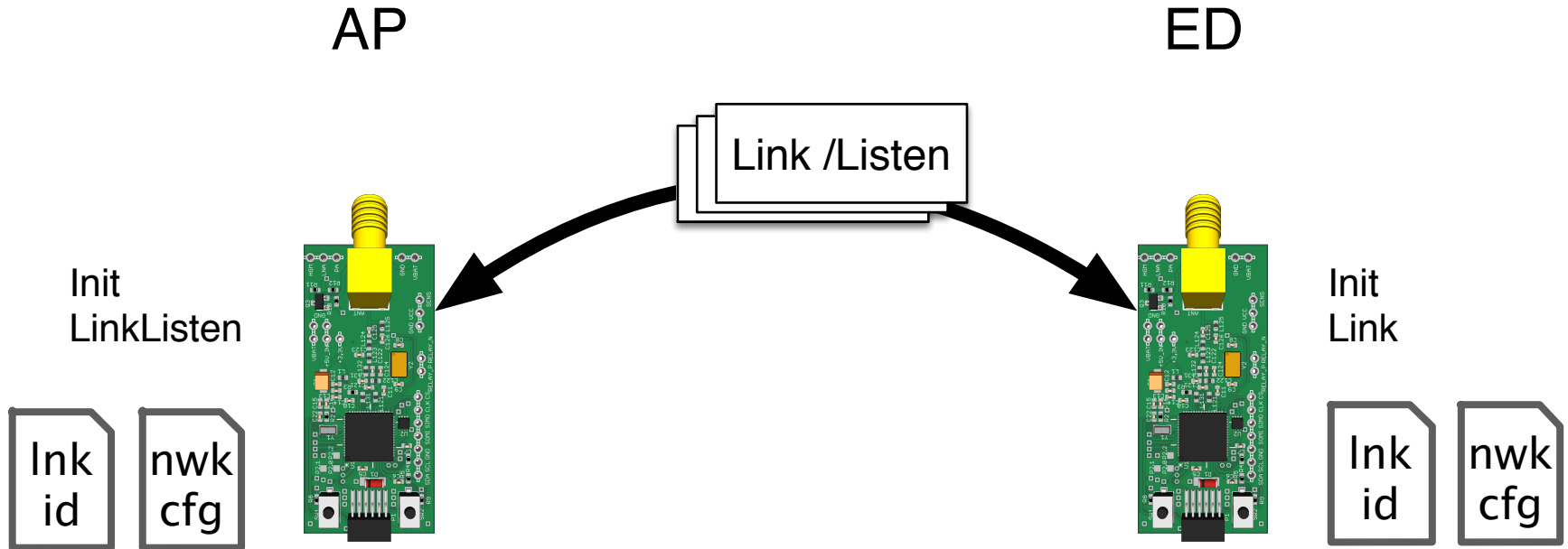
Peer to peer (with AP)

With 2 devices:

- **Access Point**
  - Manages the communication
  - Hub of the network
  - Used in star topology

- **End Device**
  - Contains the sensors and actuators
  - A strict peer to peer network only consits of EDs

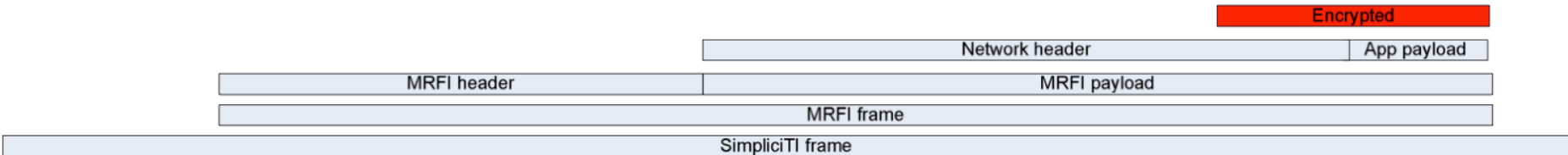- (Range extender)
  - Used in star topology as data hub/repeater

Peer to peer (with AP)

AP

ED

Link /Listen

Init
LinkListen

Init
Link

lnk
id

nwk
cfg

lnk
id

nwk
cfg

Address transmission

PREAMBLE	SYNC	LENGTH	MISC	DSTADDR	SRCADDR	PORT	DEVICE INFO	TRACTID	Security		App Payload	FCS
RD*	RD*	1	RD*	4	4	1	1	1	CTR (1)	MAC (2)	n	RD*

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">Encrypted</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="5">Network header</td><td colspan="2"></td><td>App payload</td></tr>
<tr><td></td><td colspan="4">MRFI header</td><td colspan="5">MRFI payload</td><td></td></tr>
<tr><td></td><td colspan="9">MRFI frame</td><td></td><td></td></tr>
<tr><td colspan="11">SimpliciTI frame</td><td></td></tr>
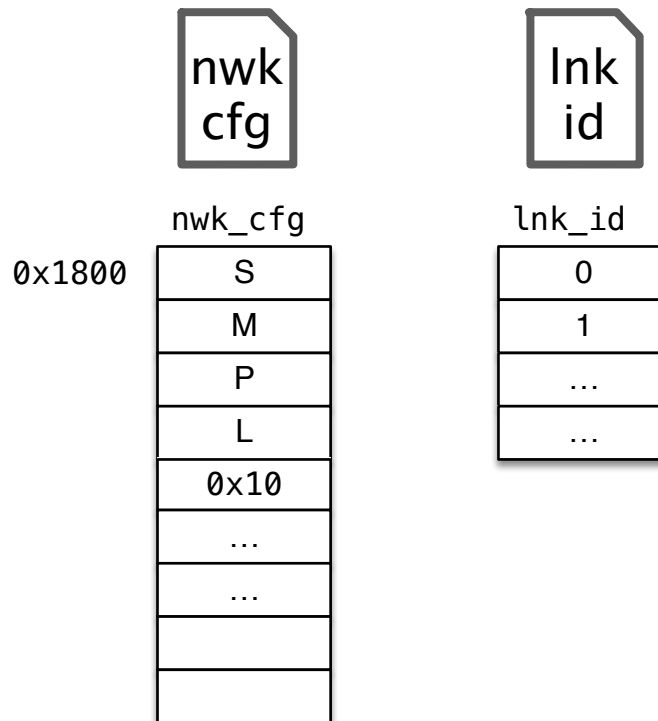</table>

*RD: Radio-dependent populated by MRFI or handled by the radio itself

Non volatile objects

Save in flash memory

- Network config
- Link ID

nwk cfg

lnk id

nwk_cfg

S
M
P
L
0x10
...
...

0x1800

lnk_id

0
1
...
...

Frequency agility

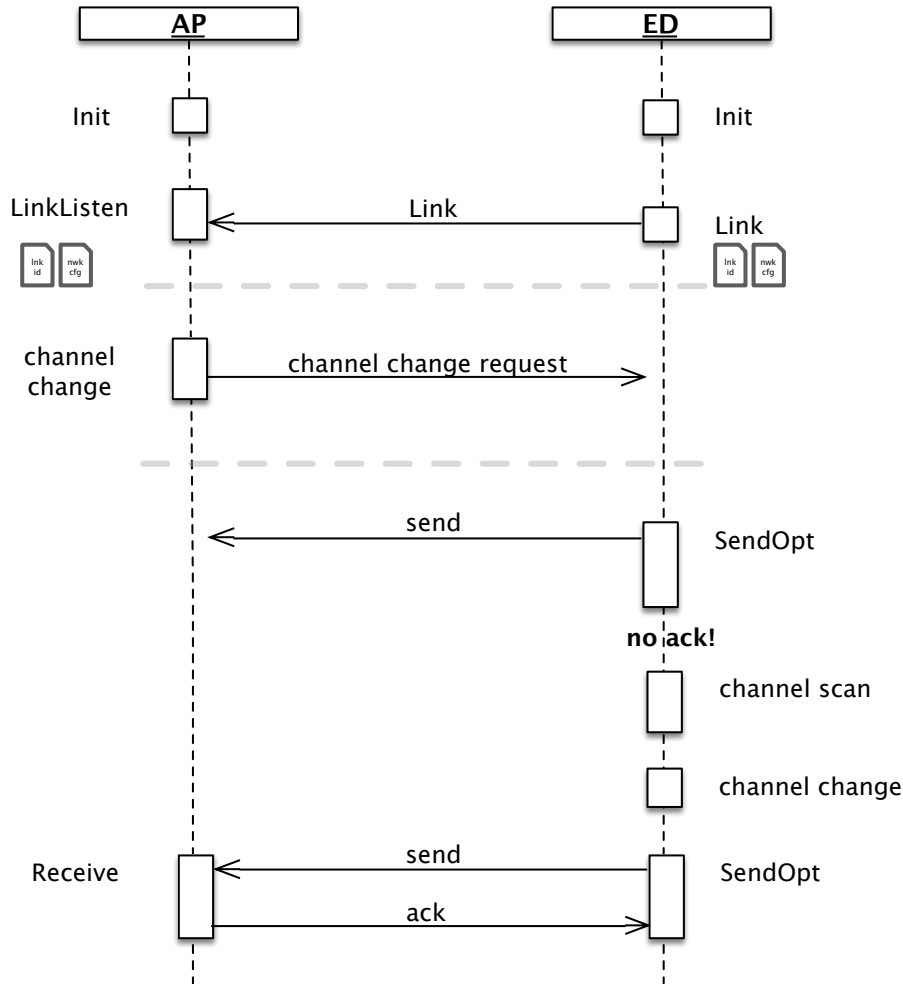Quickly change transmit/receive channel on demand

Channel change triggered by AP

ED in RX_ON

- Immediate change
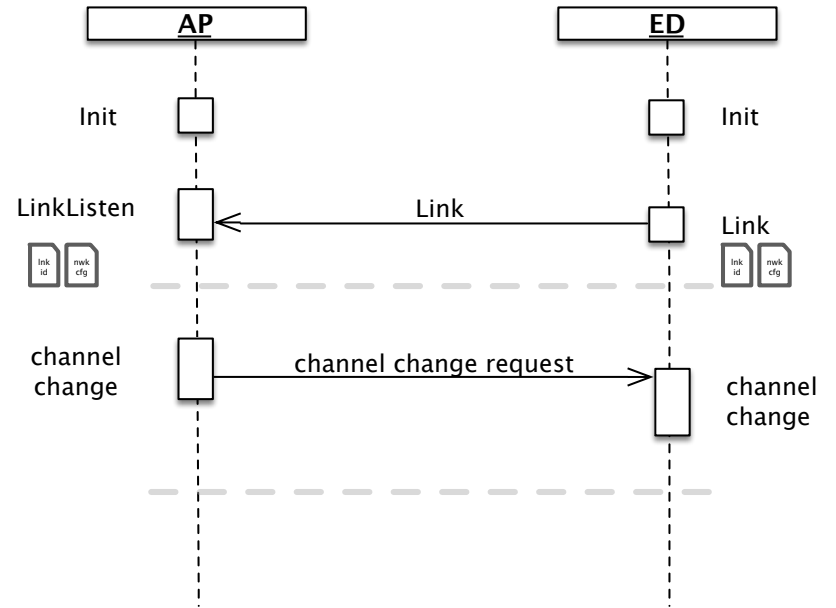- Common techniques

ED in RX_AWAKE

- No instant change
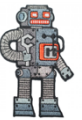- Scan for valid channel after incorrect transmission

in RX_AWAKE mode

AP	ED

Init — AP □ ‖ ED □ — Init

LinkListen — AP □ ◄——— Link ——— ED □ — Link

lnk id / nwk cfg   lnk id / nwk cfg

channel change — AP □ ——— channel change request ———► ED

AP ◄——— send ——— ED □ — SendOpt

no ack!

ED □ — channel scan

ED □ — channel change

Receive — AP □ ◄——— send ——— ED □ — SendOpt

AP □ ——— ack ———► ED □

in RX_ON mode

AP	ED

Init — AP □ ‖ ED □ — Init

LinkListen — AP □ ◄——— Link ——— ED □ — Link

lnk id / nwk cfg   lnk id / nwk cfg

channel change — AP □ ——— channel change request ———► ED □ — channel change

WSN
Practical training 3

Frequency agility

```
static const uint8_t mrfiLogicalChanTable[] =
{
  112, // GroupNR * 16      = 7 * 16 = 112
  116, // GroupNR * 16  + 4
  120, // GroupNR * 16  + 8
  124  // GroupNR * 16  + 12
};
```

SMPL functions

```
smplStatus_t SMPL_Init(uint8_t (*f)(linkID_t));

smplStatus_t SMPL_Ioctl(ioctrlObject_t object, ioctrlAction_t action, void *val);



smplStatus_t SMPL_Link(linkID_t *lid);

smplStatus_t SMPL_LinkListen(linkID_t *linkID);
```

SMPL functions

```
smplStatus_t SMPL_Send(linkID_t lid, uint8 *msg, uint8 len);

smplStatus_t SMPL_SendOpt(linkID_t lid, uint8 *msg, uint8 len, uint8 opt);

smplStatus_t SMPL_Receive(linkID_t lid, uint8 *msg, uint8 *len);
```

Project topics 1

1

IRC chat with long messages using SimpliciTI protocol

2

Use range extender and try max distance in the field (different antennas?)

3

Save values if not connected to AP, play back when connection is valid again

4

Implement your own CRC and ACK with MRFI

15   12   5   0

Exclusive OR
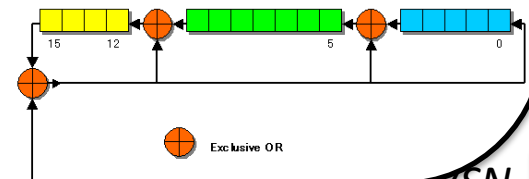
Project topics 2

5

Change channel to noise reduced one / Tell the ED/AP to change as well
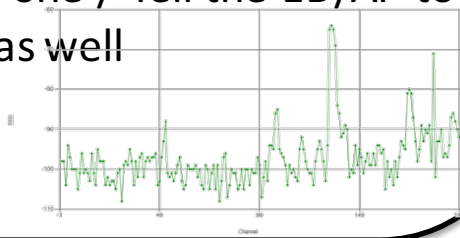
6

Write your own protocol (ID - Des - Src - Payload)
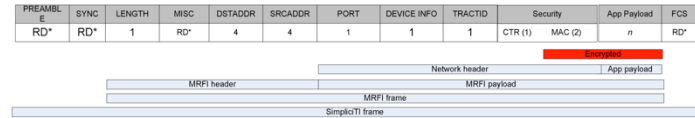
7

Triangulation of position with 3/4/5 nodes
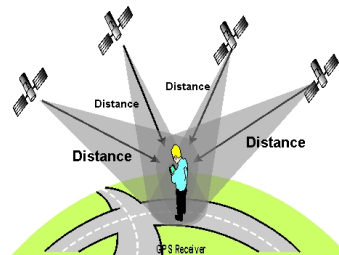
8

Secure your messages with some kind of encryption
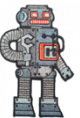
Project topics 3

9

Receive on different channels for a certain amount of time, then change the channel
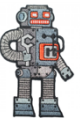
10

Transmit with different EDs on different channels

Frequency   2.40 GHz   2.4835 GHz
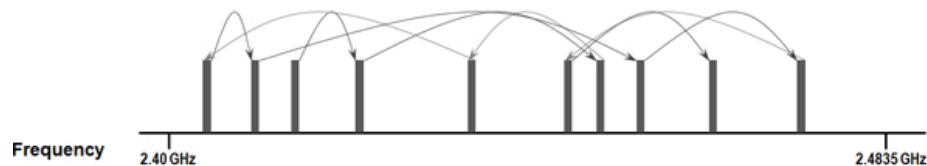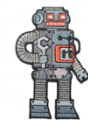
Figure 1. With Frequency Hopping Spread Spectrum, the signal is transmitted on different frequencies at intervals to spread the signal across a relatively wide operating band.
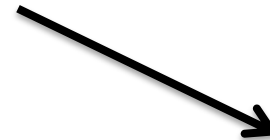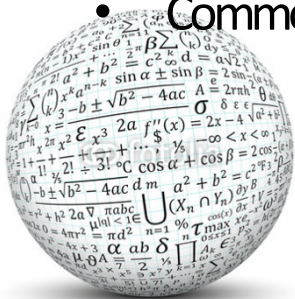
1 IRC Chat

2 Range extender

3 Storage

4 CRC, ACK
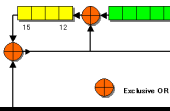
5 RSSI change
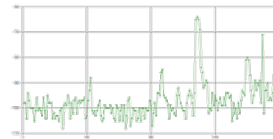
6 Protocol

7 Position
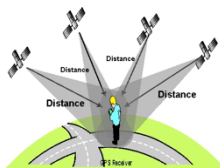
8 Encryption

9/10 Frequency hopping

WSN
Practical training 3

Projects

- Specified is only the title
- You decide the direction of your project

Theoretical background:

- How it's used
- Common techniques

Your realization:

- C Code
- Measurements / tests

Documentation

- ## Paper style

- ~ 6-10 pages
- About the general topic + project and its realization
- Code and CCS project in appendix
- Presentation as pdf in appendix
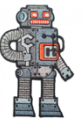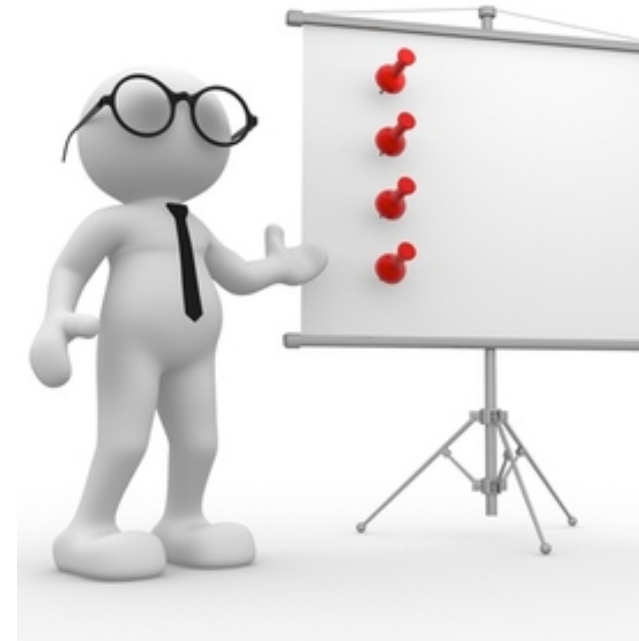- 50 % of your final grade

Presentation

- 20 min. per group
- About the general topic + your project and its realization
- 25 % of your final grade

Exam 27.05.2015   11:45   @ S159

- 90 min.
- In english language
- No aids allowed
- 25 % of your final grade

Infos

Q&A   27.05.2015   13:30   @S081

- After the exam
- About your project
- General WSN questions

my infos

location: Kampar Malaysia (GMT +8)

e-mail: matthias.wagner@st.oth-regensburg.de