

August 25 (Wed)

CMPE245

Introduction.

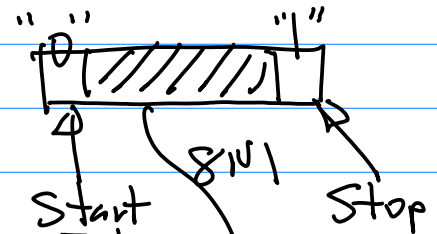
Today's Topics

Bill of Materials  
Target platform  
For Prototyping

Note: Your RF module(s)  
is to be interfaced to GPIO  
of your target Board.

Why? General Purpose  $\rightarrow$  No  
Data Framing.

for UART, Data Frame



Note: CPU Target

Option to use NVIDIA Jetson  
NANO

Guide Line for Selection  
of target Platform.

1. Register Level Control  
of GPIO, SPI Controllers

For LPC1769 ✓

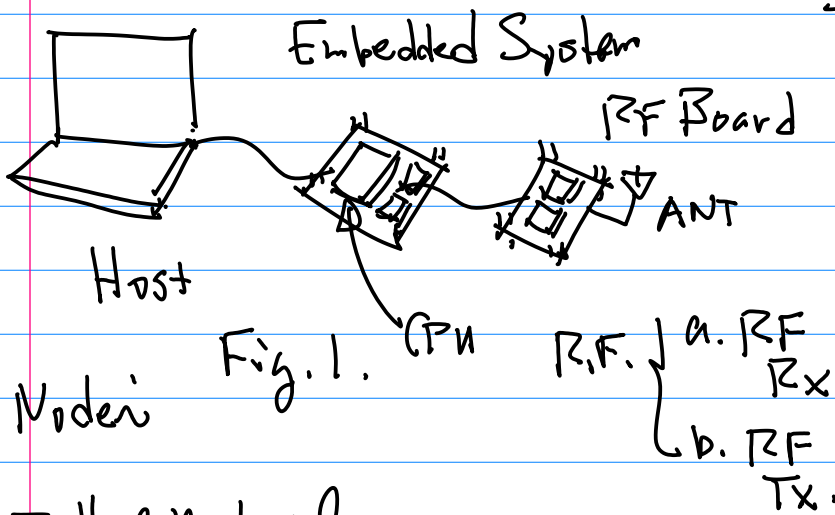
For NANO  $\rightarrow$  Devices  
Drivers.

① Datasheet + root pages

② Software Dev.  
Environment, tools.

Jetpack (OS +  
Libs)

③ O.S. Distr.



Bill of Material

1. Target CPU NXP LPC1769  
OR NANO (NVIDIA)

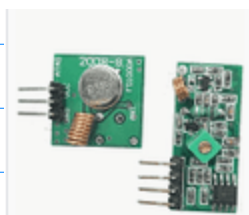
2. RF modules, physical Layer only.

② ASK R.F. module ① frequency  $\sim 433\text{MHz}$   
F.C.C. Certified

Amplitude Switching

③ Open Spectrum.  
Power  $\leq 1000\text{mW. (1W)}$   
Tx: Transmitter

④ No MAC (media Access Control)  
Needed



August 25 (Wed)

Note: ON your RF Board.

2

2° Access to CPU pins, e.g.  
GPIO pins, SPI pins.

Homework: Purchase ASK RF  
Module By Sept. 8th (Wed)  
OR ideally Sept 3rd (Fri)

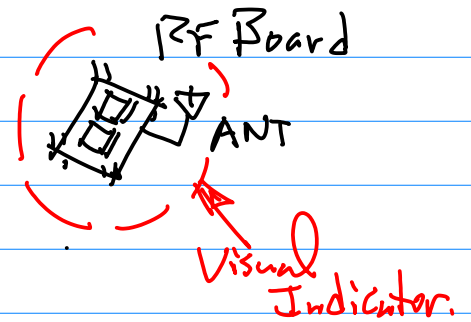


Fig. 3.

2 Blocks { Rx  
Tx  
Both Need to Be Powered.

Note: To provide Hardware Debugging CKT. on  
the RF Board

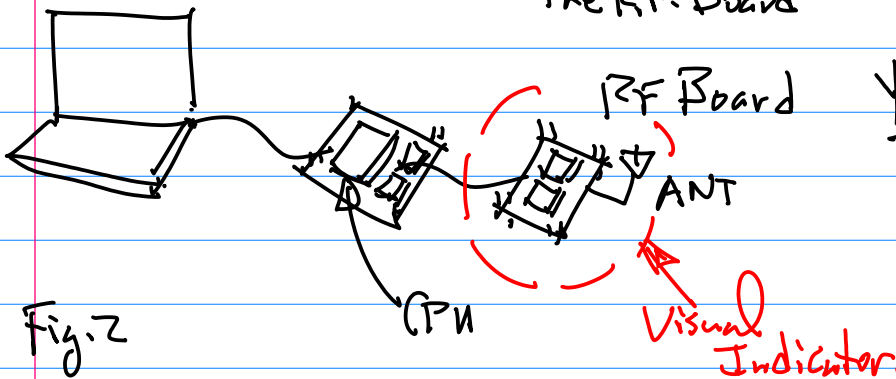


Fig. 2

You may want to have the  
DC PWR Delivered via  
CAT5 Cable from your  
Embedded Board.

RJ45 Pins (Pin): 8

Debugging Capability on the R.F.  
Board:

(1) Objective: To visualize/observe  
GPIO output.

Means: LED.

Material:

a LED (Red, Green), 4~10mA

Connectors (to cable to RF Board)

b RJ45 Right Angle Connectors

(2)

A piece of CAT5 Cable (Ethernet)

c Components { Resistors  
Caps.

(1117?)

7805, 7812 OR

August 30th.

RF module, to Build  
RF Board.

1° ASK RF.

Amplitude Shift Keying

2° Rx: Receiving

TX: Transmission

3° GPIO JF

Regulator Data Pin

Homework: Identify/Bring Your Wire Wrapping Board for RFB Design. 4x3 Inch;

To Build RFB

1° Board

2° 4 standoffs

3° Build I/O I/F Testing CKT.

To Light up LED when CPU output "1"

To Turn off LED when CPU output "0"

Output Testing

SW Toggles to Vcc, when Vcc, CPU Reads as "1"

SW " " to GND, CPU Reads as "0"

Input Testing

4° CAT-5 Ethernet Cable

RJ45 Right Angle Connectors (2)

One for Embedded;  
One for RF Board,

5. Power Distribution to RF Board:

a 5VDC is adequate  
But R.F. module can be operated with

Bigger Power, 9VDC

OR 7.5VDC may be needed during Debugg

ing;

Sept 1. (Wed)

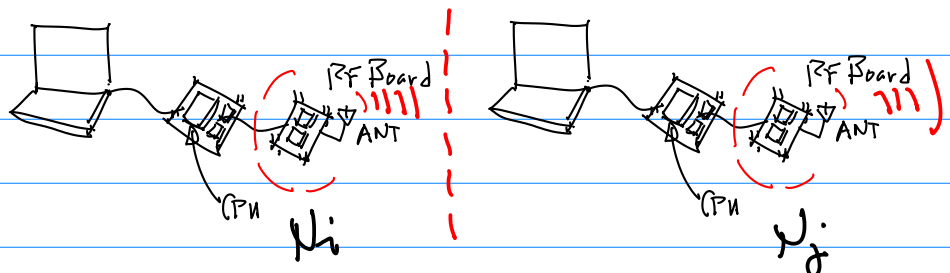
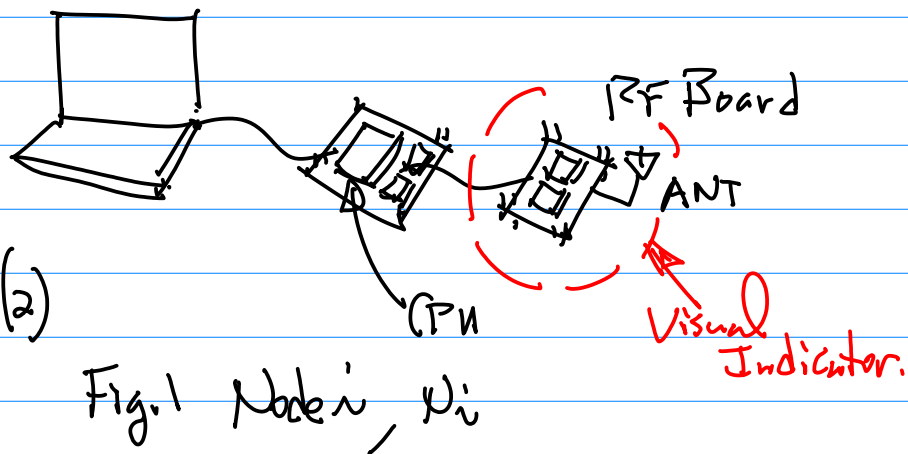
Ref: IEEE 802.6

1. github/Rualili/cmpe245/2018F

"2021F-"

2. Topics: Design R.F. Board for the 1st homework. 2pts. (Hardware)

From PP.2. System View



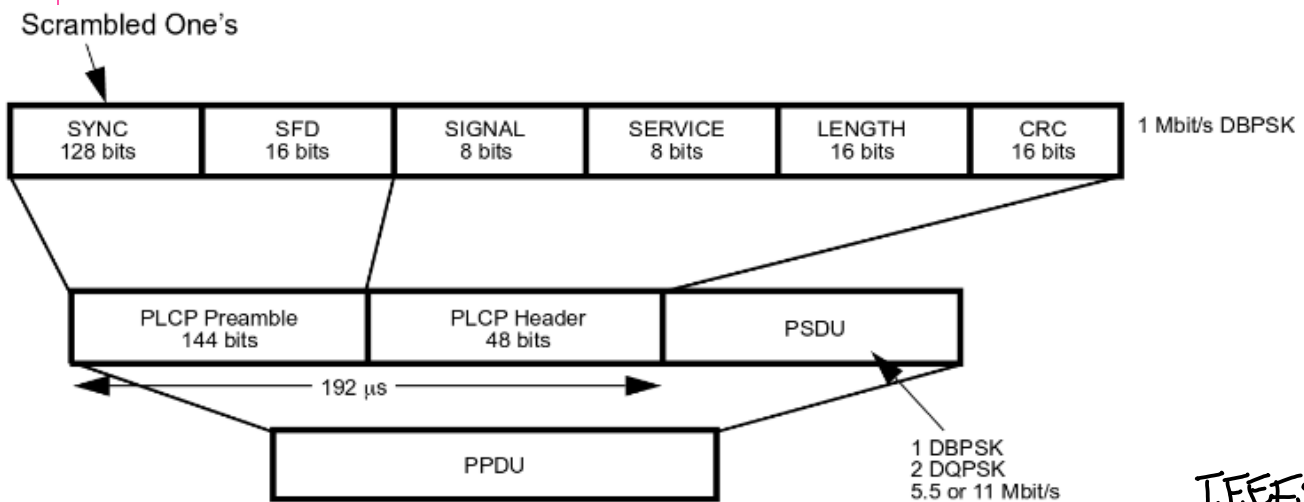
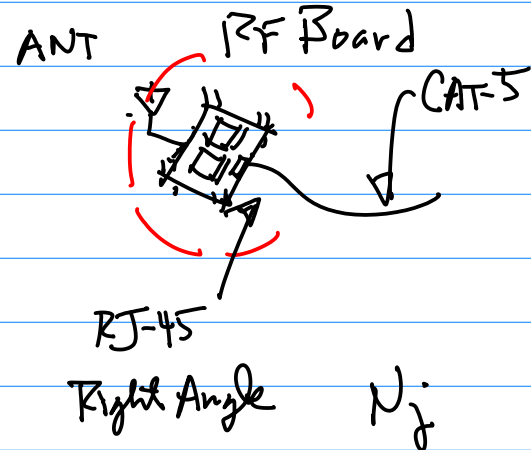
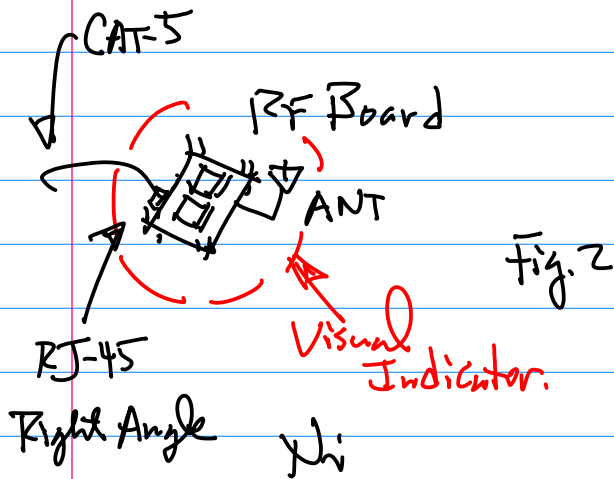


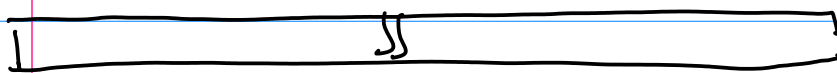
Figure 127 - Long PLCP PDU format

IEEE 802.6

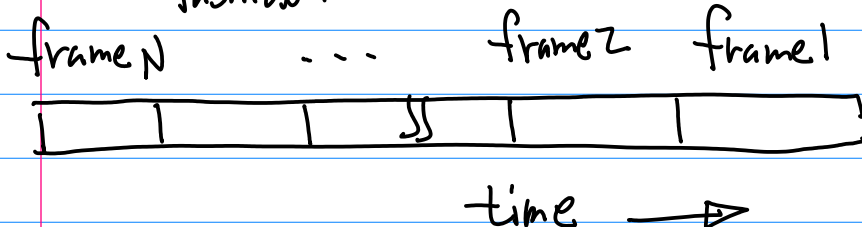
Fig. 3. TP.13

Node  $N_i$ , Node  $N_j$ , Each Node is illustrated in Fig. 1.

Sending Stream of Data from  $N_i$  to  $N_j$



String of Bits Sent in A "Framed" fashion.



Note: Consider the Construction of Each frame, frame  $i$ ,  $i=1, 2, \dots, N_j$

What are the Contents to be placed into A Frame?

like start bit(s)  $\rightarrow$  to mark the Beginning of the Communication.  
 $\rightarrow$  Timing/Sync.

C.R. (Cognitive Radio), SDR (Software Defined Radio)

multiple Bits in Sync. Design  $\rightarrow$  more Reliable  $\rightarrow$  "Robust" performance  
Maybe

Question: What is the general guideline in terms of designing "Sync" Field?

multiple Bits of what?

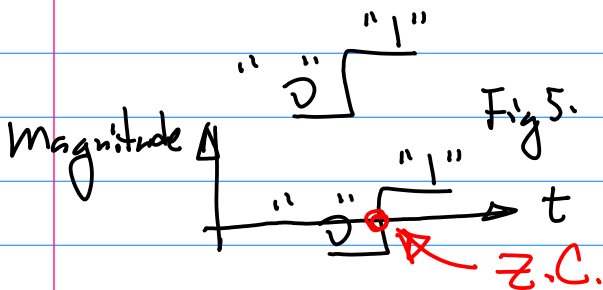
3 bits Example:  $2^3 = 8$

b2	b1	b0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Transition  $\rightarrow$  Change of State

"Zero Crossing"  $\rightarrow$

for Example "1"  $\rightarrow$  "0" OR



To provide best possible "Transitions", e.g. "Z.C."

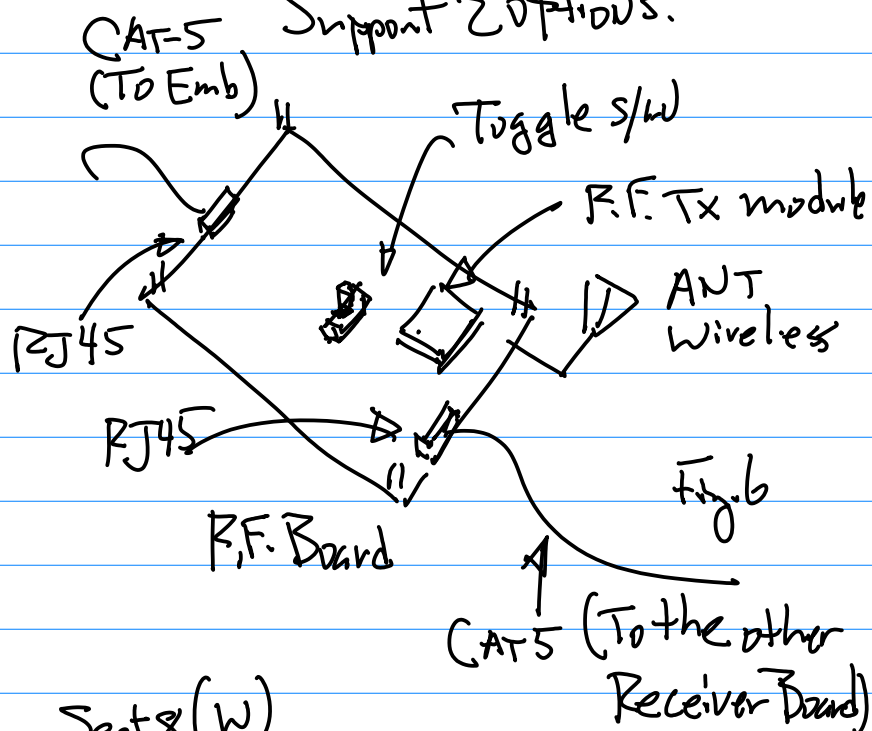
Pick 0x5 Binary: 101

The Objective for R.F. Board Framework/Design: To Allow  $N_i, M_j$  to Sync.

2-Step Approach:

Step 1. Based on Land Line;  
Step 2. then R.F. (Wireless)

Hence, R.F. Board will have to Support 2 Options.



Sept 8 (W)

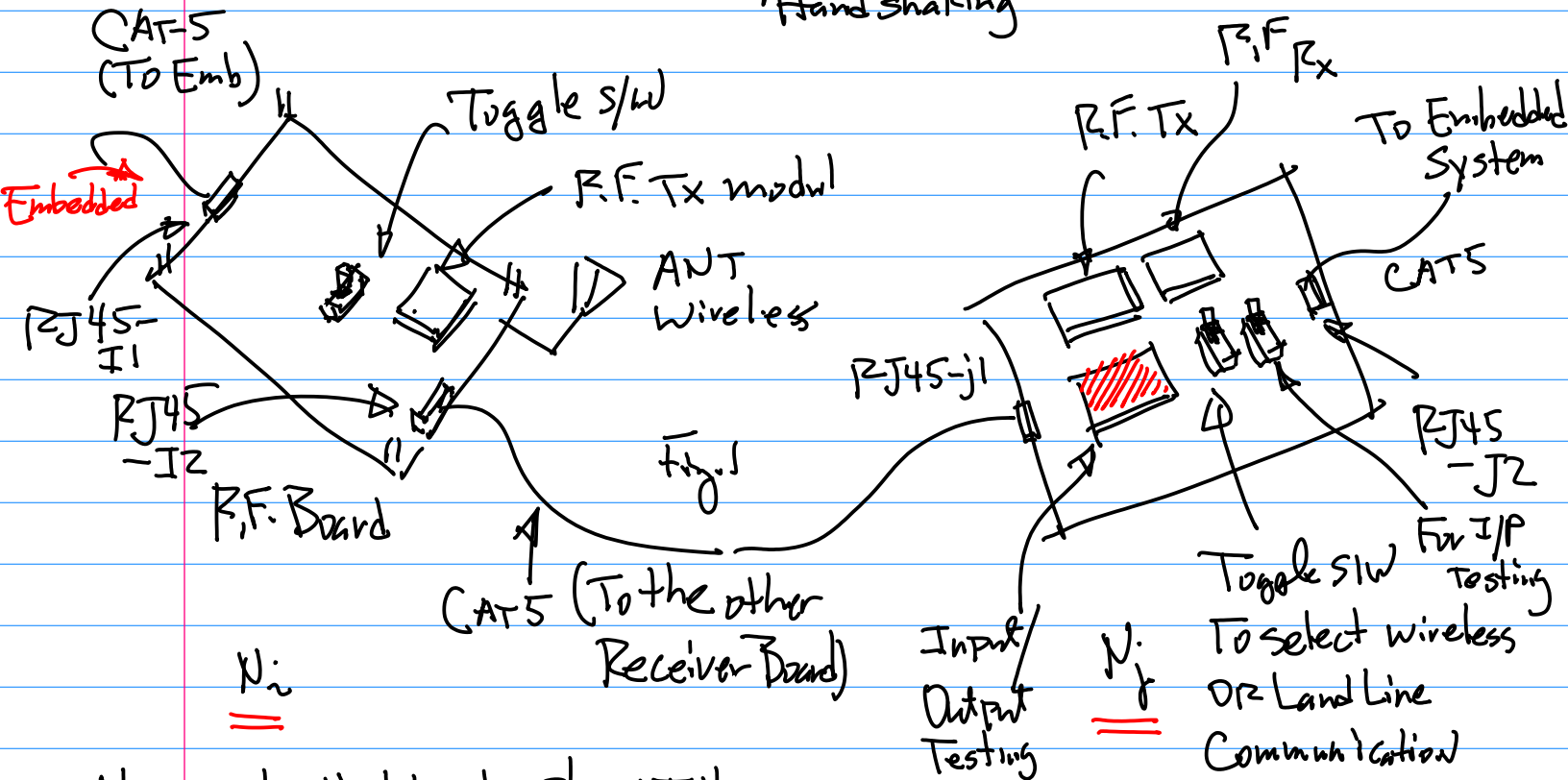
Topics: 1. Prototype for HandShaking (For LPC1769, NANO, OR your choice, such as Pie)

Note: NANO Boards are not delivered yet. So, please find your solution.

Note 2. Semester Long Project, please form

4 person team;

Example: "Land-Line" Based Design for Hand shaking



Homework: Next Monday, Show & Tell

for Each Person's R.F. Prototype Board.

Things to show for the next class: 1. 4x3" Prototype Board; Pre-Built Through-Hole; metal coating of the through-holes any; 2. Stand-offs

3. RJ45 Connectors (2x), Right Angle; Mounted on the Board;

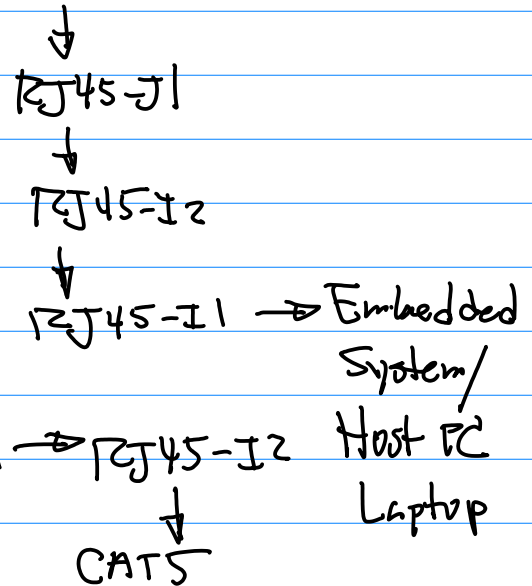
4. Testing Circuit

{	Output Testing	Output "1" from the Host/Embedd System
		Output "0" " " " " " "
	Input Testing	Input "0" originated on your R.F. Board
		Input "1" " " " " " "

↓  
RJ45-J1  
↓  
RJ45-I2

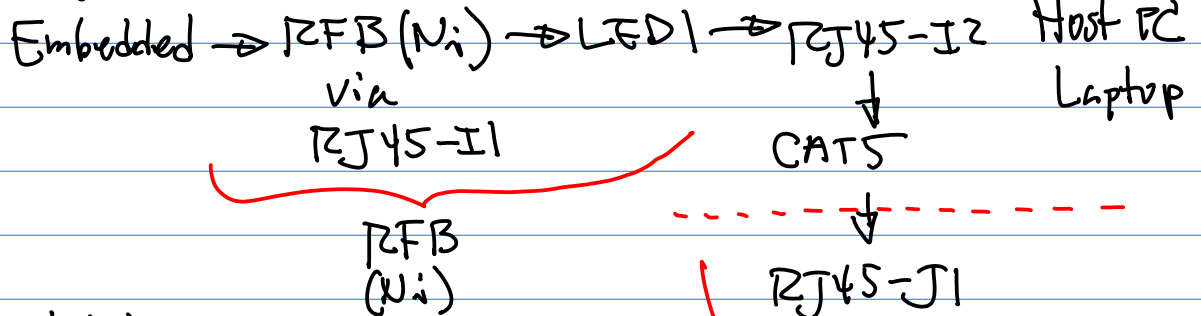
Continued,

Input Testing { Input "0", originated on your RF Board  
Input "1", " " ...  
(Similar to Input "0")



Testing CKT:

Output Testing CKT  $\rightarrow$  Red LED.



On RFB ( $N_i$  &  $N_j$ )

GPIO output from LPC/MANO

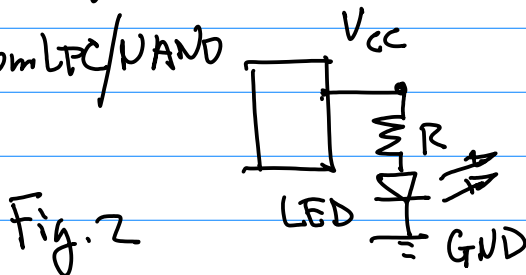


Fig. 2

$$V_{CC} = IR + V_{LED} \dots (1)$$

$$V_{CC} (CMOS) = 3.3VDC$$

$$V_{LED} = 1.8VDC; I = 10mA;$$

Solve for  $R \sim 250 \pm \Omega$  or higher;

Input Testing CKT.

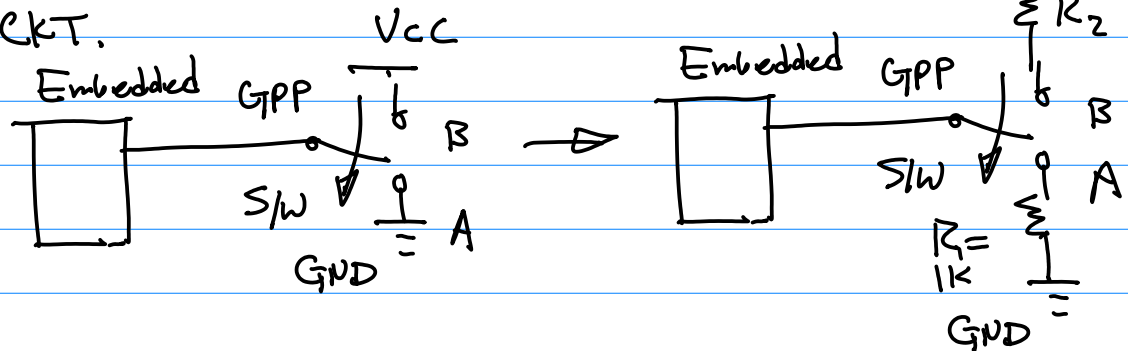


Fig. 3a



Sept 13 (Monday)

Today's Topics: 1° Hardware platform.

2° Sync Algorithm, LISA  
(Linear Invariant Sync. Algorithm)

Ref: 1° ... github, ID: ~106...

for Target RISC-V 2021F-106  
(System On Chip)

2° github/hualili/Cmpe245/...

2021F-105 ~ (NANO  
Connector)

FPGA Solution (RISC-V), Igloo2.

a Board from Future Electronics

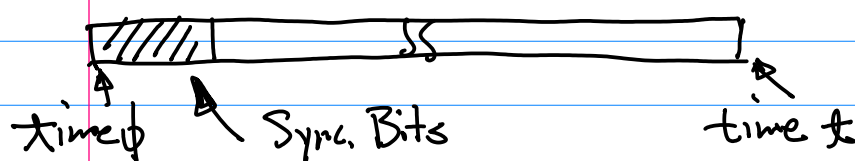
b IP-Core for CPU in Verilog  
Can be downloaded;

c TAP Plastics

NANO Board. Connector J-41  
Identify GPIO

Example: Let's consider the Design  
of SyncField in Wireless  
Communication

Consider 1° Place to define sync.  
Bits.



8  
Suppose we have a letter  
in a Hex Number

0x4f  
0100 : 1111 Binary  
Equivalent

"0" Due to noise, the  
corruption may lead  
to a failure

Correct  
0100 : 11110 ...  
missing  
Sync.  
Bit

2° Sync is established  
Based on "zc"

The Change of the State of  
a Signal, For Example  
"0" → "1" OR "1" → "0"  
Preferred change is to  
include "1".

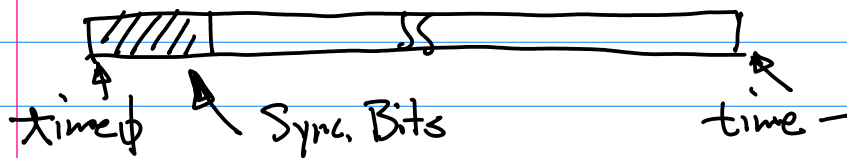
Requirements:

1° Establish Sync with  
Change of State

01010101 ...



OR  
101010...



01010101 ... OR ... (1a)  
101010... ... (1b)

Consider Random Disturbances, alters one bit in the Sync field. (1a)

of the Sync field,  
e.g. the Beginning  
of the payload.

~~0~~10101...  
1 (Random Noise)

Suppose we know the Sync. field  
Consists of 10 Bits. → Discard the  
1st 2 bits until we have 0101...

Pattern again, to satisfy the total  
Number of bits in the Sync, Definition.

Question: How to utilize the Sync  
Pattern even when this pattern  
is corrupted, And not to start  
over again?

(1) The Need to Re-use the Sync Field  
Even if it is corrupted;

(2) The Need to Know where is the end of

Sept 15 (W)

Homework: Due Next wed.  
Official (2pts) Sept. 22nd.

1° R.F Board Prototype (finish  
this Prototype for Landline  
I/F).

2° Run a simple program  
C/C++ or Python to perform  
Testing of Input/Output  
function.

Input Testing: GPIO Reads  
Input "0" AND/OR "1",  
Note Toggle the Switch on the  
RF Board to produce "0", & "1".

Output Testing: LED ON when Output = "1"  
LED Off " " = "0"

What to Submit:

1° One Description of your design implementation;

2° Photo of the System Set up

Host Laptop + Embedded Board + RF  
LPC/NANO Board

3° Source (Soft Copy) LPC1769

Export your Work as a project

NANO, Python or C/C++ Code

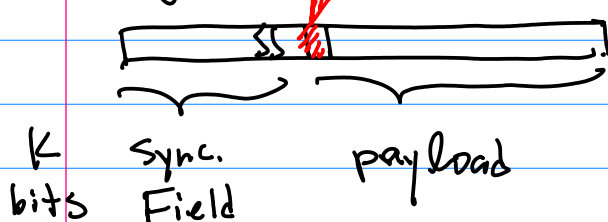
4° Video Clip 5~10 Sec.

~~ON CANVAS~~ OR Submit the Zip  
file via E-mail;

Consider Sync. Field Design:

Design Requirements:

1° Re-use the k-bit Sync  
to Be able to identify  
the start bit of the  
payload



2° To Establish Timing  
Even if Random disturbances  
Corrupted some bits in  
Sync. Field, in addition to  
pin point to where the  
Corruption occurs.

Embedding ID Index into  
Sync Field.

From 101010...

Change it to the following.

a Take 8 bits

10101010 at a time

Taking the 1st 4 bits of this  
Segment (1 byte), preserve  
its pattern

prefix: 1010

Then, Taking the 2nd 4 bits  
modify it to make it as  
an index to Refect this  
Byte position in the Sync  
Field.

e.g. 0000 ... "0"  
0001 ... "1"  
0010 ... "2"  
0011 ... "3"

0 0 0 0 ... "0"  
 0 0 0 1 ... "1"  
 0 0 1 0 ... "2"  
 0 0 1 1 ... "3"  
 ...

1 1 1 1 ... "f"

Assemble prefix and ID together

1 0 1 0 | 0 0 0 0  
 1 0 1 0 | 0 0 0 1  
 ⋮

1 0 1 0 | 1 1 1 1

a 0 | a 1 | a 2 | ... | a f ... (1) 16 Bytes

Now, Change 1010 to 0101 = 0x5  
 Therefore,

5 0 | 5 1 | 5 2 ... | 5 f ... (2)

Integrate (1) & (2) together, 16 Bytes

a 0 | a 1 | a 2 | ... | a f | 5 0 | 5 1 | ... | 5 f

e.g.

0x a 0 | 0x a 1 | ... | ... | 0x 5 f

32 Bytes (256 bits) ... (3)

Design An Algorithm for Sync.

Extraction @ Rx End

Question: What is the minimum Number of Bits in (3) do we need to establish Sync.?

Remark 1: Minimum 8 Consecutive Bits with first 4 bits from alternating Bit Pattern, e.g. 0x A, or 0x 5 is Needed to Establish

Sync. Given

Question: Sync. Based 8 bits v.s. Sync. Based on 16 bits, which one gives higher Confidence? Ans: 16 bits.

Question: What is the number bits that gives the highest confidence level? 256 bit.

Let's define Confidence level  $\eta$  as follows,

$$\eta = \left( \begin{array}{l} \text{Sync. Establish} \\ \text{with K bits} \end{array} \right) / \left( \begin{array}{l} \text{Total No. of} \\ \text{bit in Sync.} \\ \text{Field} \end{array} \right) \quad \dots (4)$$

Sept. 20 (Mon)

Topics: 1<sup>o</sup> LISA Conclusion  
2<sup>o</sup> Base Band Signals  
with Definition modulation  
Techniques, ASK, FSK, PSK.

3<sup>o</sup> Project 1. Assignment

Software Defined Radio  
Implementation of LISA.

Due Oct. 8th (Fri), 11:59pm.

Written Requirements to Be  
Posted on git & STOR CANVAS.

Official (10pts)

a. "Land Line" LISA

(Based on Homework  
of Input/Output Testing)

b. RF. "LISA".

ASK RF Rx & Tx for the  
implementation.

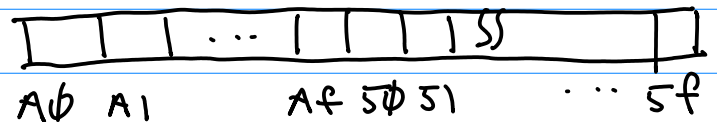
Example: Compute  $\eta$  (Confidence  
Level, CL) for the Sync  
established Based 8 bits  
data from the Sync. Field.

Sol. From Eqn (4),

$$\eta = \frac{8}{\text{Total Bits in the Sync Field}} = \frac{8}{256} = 2^{-3} = \frac{1}{8}$$

$$\text{Field} = 2^{-5} = \frac{1}{32}$$

Fig. 1



Linear: Index Arranged in a Linear  
fashion;

Invariant: Capture 8 bit sync  
pattern regardless of  
its/their position(s);

Conclusion: LISA algorithm  
Provides Robust, Versatile  
Sync. Scheme to Allow  $N_i, N_j$   
to Establish Sync. (Hand shaking).

Homework: Write C/C++ (or python)  
for your hardware platform, to Realize  
Base Line "LISA". (Due Sept. 27  
Monday) 1 pt.

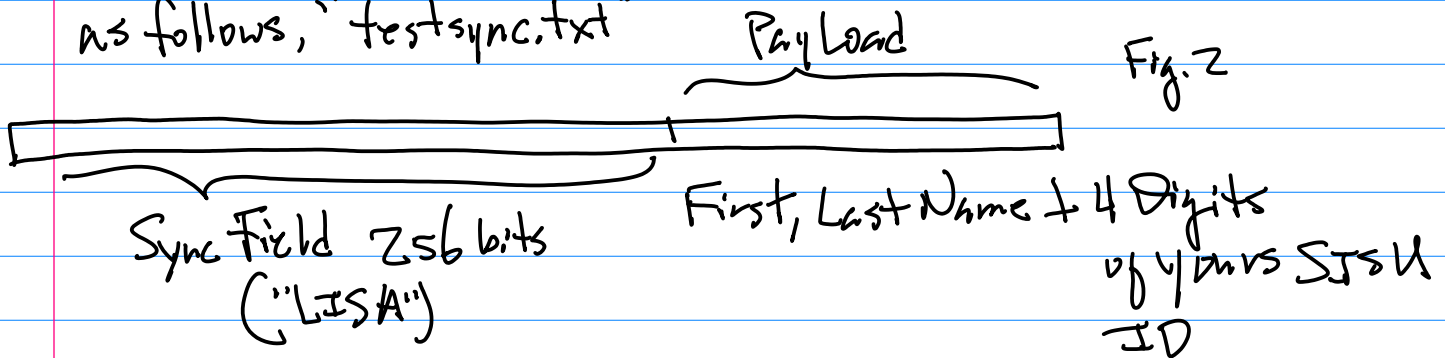
a. Prompt the user for his/her  
input for the Number of Bytes  
to establish Sync.

1 Byte, 2 Bytes, ..., 32 Bytes

b. Implementation for "the Base  
Line"

Baseline: Consecutive Bytes.

c. Creates A Test Pattern as follows, "testsync.txt"



plaintext file.

d. Run your program to process the testsync.txt, to extract Sync. By Printing the payload.

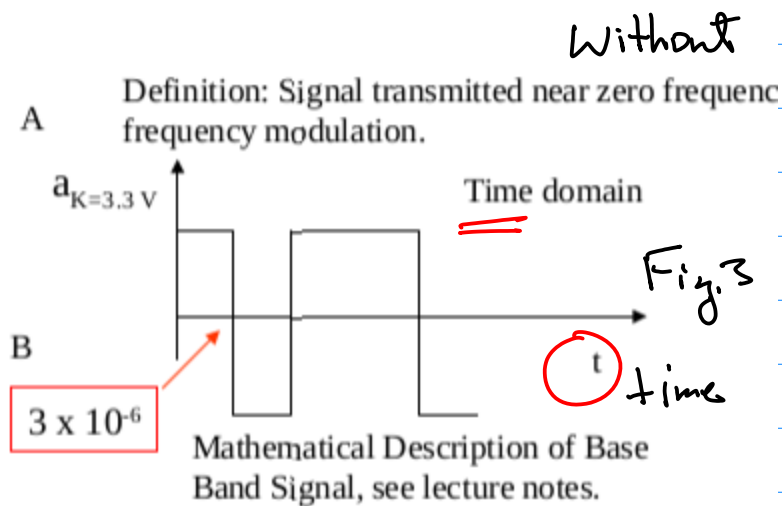
Note, then future framework will allow the Bytes from Non-Consecutive order.

Note: Optional Requirements  
Pseudo Code.

What to Submit:

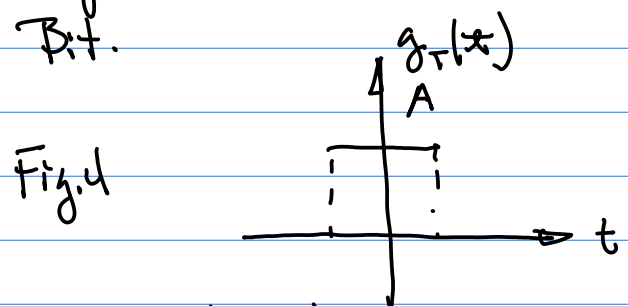
- Source Code (Indicate if on Linux, or Windows, or LPC, or NANO)
- Binary Executable;
- Optional, Pseudo Code.

Base Band Signal:



Signals from GPS, PWM, etc.  
for example.

Suppose the Signal is periodic Signal. Concentrate on a Single Bit.



Subscript "T" in  $g_T(t)$  for One bit

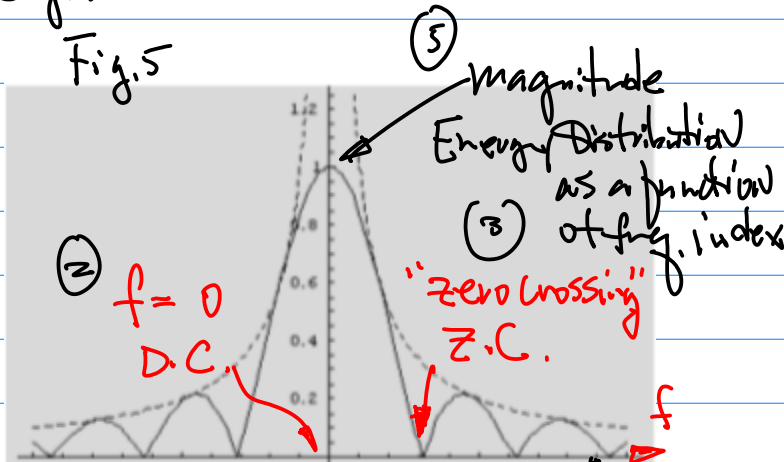
$$g_T(t) = \begin{cases} 1 & \text{for } t \in [-T/2, T/2] \\ 0 & \text{otherwise} \end{cases} \quad \dots (1)$$

(bps - Bit per second)

Sol:

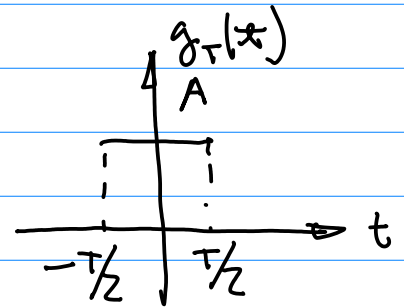
Frequency Characteristics of the B.B. Signal:

Fig. 5



(4) Base Band Defined Between 2 Z.C.s "1st Zero Crossing": Bandwidth.

(1) frequency Index  $f$



From the given bit rate,

$$f = 9600 \text{ Hz}$$

$$\text{Hence, } T = \frac{1}{f} = \frac{1}{9600}$$

$$= 0.104 \times 10^{-3} \text{ Sec}$$

$$= 104 \times 10^{-6} \text{ Sec (micro Second)}$$

Consider Frequency Characteristic in Fig. 5, Define Band width of the  $g_T(t)$

$$BW =$$

(Between the 1st pair of Z.C.s)

Eye Pattern: Characterization of Base Band Signal

IEEE

IEEE 802.11b Standard pp. 56

Sept 22 (W)

Topics Today: Base Band Signal Analysis.

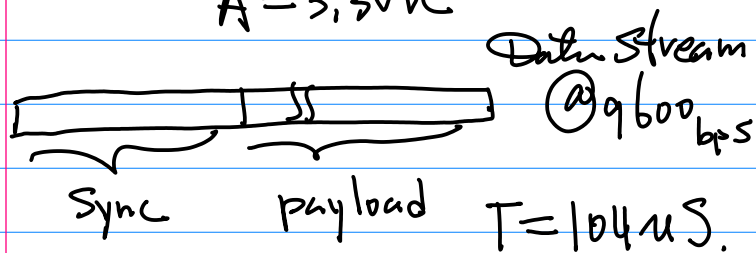
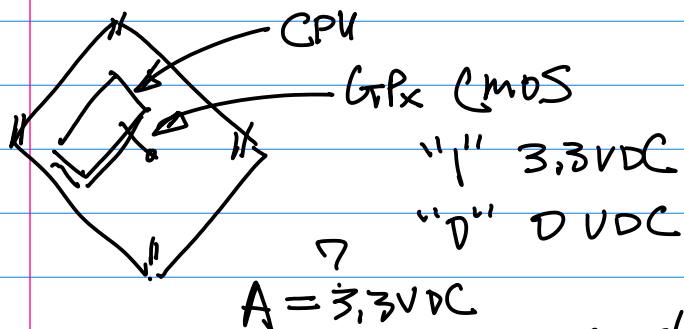
$g_T(t)$  from Eqn (1)

Example: Find  $T = ?$  for a Base Band Signal @ 9600 bps

By Fourier Transform,

$$g_T(t) \xleftrightarrow{\text{F.T.}} F[g_T(t)] \quad \dots (2a)$$

$$F[g_T(t)] = A_T \frac{\sin \pi f T}{\pi f T} \quad \dots (2b)$$



Find Band width from the 1st pair of z.c. in Eqn (2b)

$$A_T \frac{\sin \pi f T}{\pi f T} \stackrel{\text{set}}{=} 0$$

then  $\frac{\sin \pi f T}{\pi f T} = 0$ , Hence

$$\sin \pi f T = 0$$

$$\pi f T = n\pi$$

$$n = 0, 1, 2, \dots$$

$$f T = n, \text{ where } n = 1 \text{ 1st pair z.c.}$$

$$f T = 1, \therefore f = \frac{1}{T}$$

$$\text{Therefore, B.W.} = \frac{2}{T} = \frac{2}{T} \dots (3)$$

Example: Suppose a Base Band Signal is operating @ 9600 bps.

1. Find its Bandwidth?

2. If we double the Bit rate, find its new Bandwidth?

Sol: For B.W., from eqn (3)

$$B.W. = \frac{2}{T}$$

$$\text{and } T = \frac{1}{f}, \text{ or } f = 9600$$

$$\text{hence: B.W.} = 2f = 19200 \text{ Hz}$$

When the bit rate is doubled, so is the B.W.

And  $T_{\text{new}}$  is the half of its original value.

PP60 IEEE 802.11b

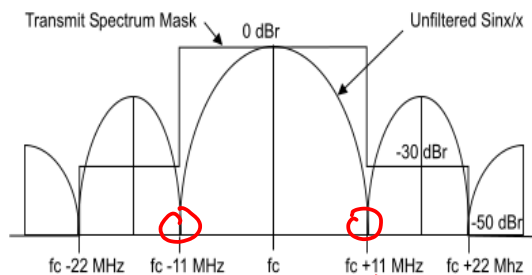


Figure 145—Transmit spectrum mask

18.4.7.4 Transmit center frequency tolerance

$$B.W. = f_c + 11 \text{ MHz} - (f_c - 11 \text{ MHz}) \\ = 11 \text{ MHz} + 11 \text{ MHz} = 22 \text{ MHz}$$

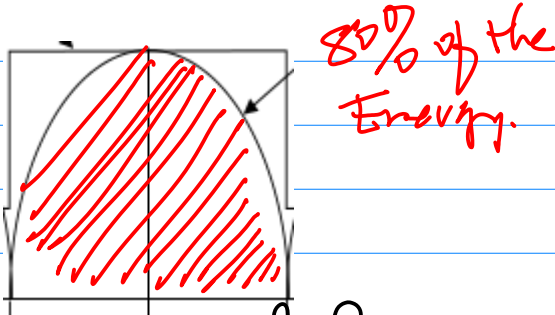
$f_c$  = Carrier frequency  $\approx 2.4$  GHz from the modulation

Carrier Signal (RF) is modulated



by a Base Band Signal.

80% of the total energy of the R.F. Signal (Base Band + Carrier) has to be Captured within the Base Band



This 80% Requirement has to be Satisfied by F.C.C.

$$\left( \frac{\text{Energy of the Base Band}}{\text{Energy of the total Signal}} \right) = 80\% \quad \text{or} \quad \dots (4)$$

higher.

Question: How to Pack more energy into a Base Band (1st Pair of the F.C.)

Discrete Fourier Analysis Tool for the Design.