CMPE245
Sept. 7

Sept 7.

Note: 1° Homework (RF module & RF Work-in-Progress) Due today Inspection in Class.

Homework (1st) Due A week from Today. Write C/C++, OR Python to Implement LISA Algorithm (Phase I) Such that:

1° Console Input from user to Select No. of Bytes for Synchronization.

Print the payload message.

Note: Python Implementation on Jetson NANO, OR R-pie 3 B+ or 4, you can do the same.

Note: This homework is for Laptop Based Implementation.

Based on the Homework (Today. RF. Board) we will Continue with "Landline" Testing Capability.

For Example: 16 Bytes or 32 Bytes.

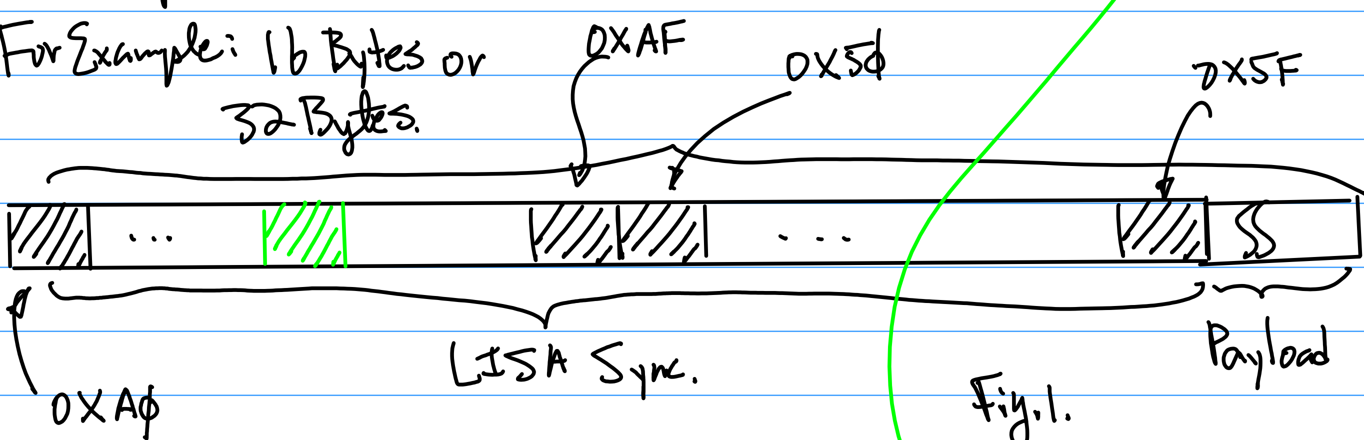
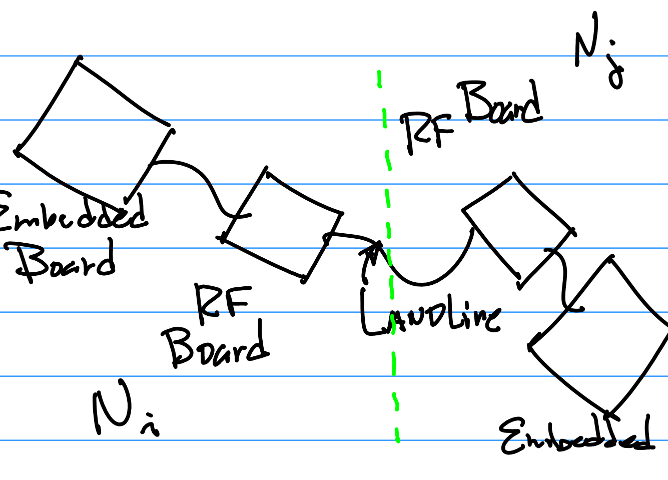


Fig. 1.

2° Note in the future (phase II) We would like to Extend this Implementation to Allow a Single Byte (as "Green" in Fig. 1. Matching to the LISA Sync Field to Establish Synchronization.



3° Payload: First Name + Last Name + 4 Digits ID + CMPE245 + SJSU

CmpE245
Sept 7

Example: Ref from the Class github, ID: 2018F-104~

Observation 1: The Minimum Number of Bytes to establish Synchronization is 1. Therefore 1/32 Bytes for the Sync. → Confidence Level / Index η

$$\eta = \frac{\text{No. of Bytes to Establish Sync.}}{\text{Total Number of Bytes (32)}} \quad \cdots (1).$$

Note: In Software Defined Radio, We can Change η (Confidence Level) to trade the quality for Speed if it is allowed.

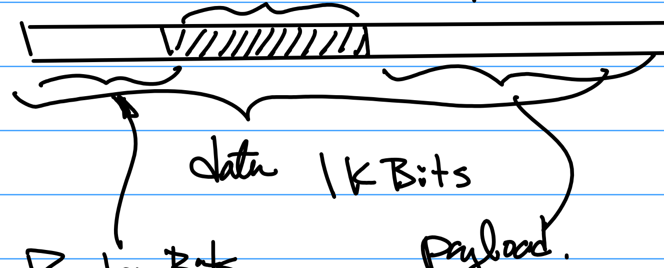
In Cognitive Radio Design, We would like to have this Ability.

Observation 2: "LINEAR" Characteristics is from the fact LISA Index is defined from 0 to F with Linear increment. And "Invariant" Characteristic is due to the fact the ID Index, e.g. Ranging from 0 to F will allow the Algorithm to pin point to the Begining of the payload.

2018F-105~

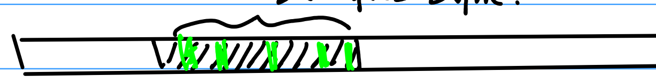
Example: Homework ON LISA from the Class github.

1. 32 Bytes Sync.



2. Sync Field is Corrupted.

32 Bytes Sync.



256 Bits (=32 Byte in Sync Field)

↓

10% Random Bits Corruption.

↓

256×10% = 26 Bits Random to alter Sync Field.

Generate Random Bits. (26 bits).

Use "XOR" Bitwise at Any Arbitrary location within the Sync Field.

3. User Input for the No. of Bytes (as Confidence Level), then the Code will parse the input file with the Confidence Level to Establish Synchronization.

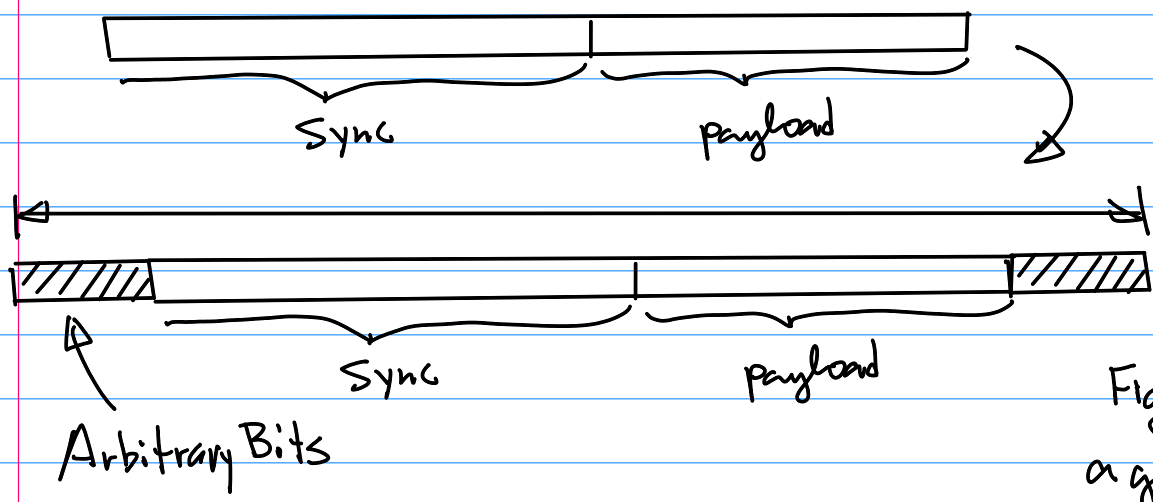
Sept. 12 (monday)

Today's Topics: 1° LISA Homework Implementation. 2° BaseBand Signal

Cmpe245

Sept.12

Example: LISA Implementation.

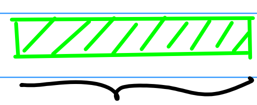


1K or 2K Buffer

... (1)

Fig.1. As a given condition.

Step2. Create a "mask" Template to Reflect the matching size that you like



... (2)

Step3. Add Random Noise to Fig1. then move (2) mask at the beginning bit (1st Bit) of (1). Such as

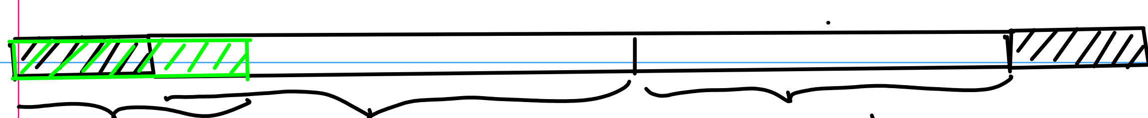


Fig.2

Check the matching result Between (1) (Black) and (2) (Green) if matched, then use the matching index to jump to the 1st Bit of the payload; O/W, Continue By shifting the mask 1 bit to a newer location.

CmpE245
Sept. 12, 22

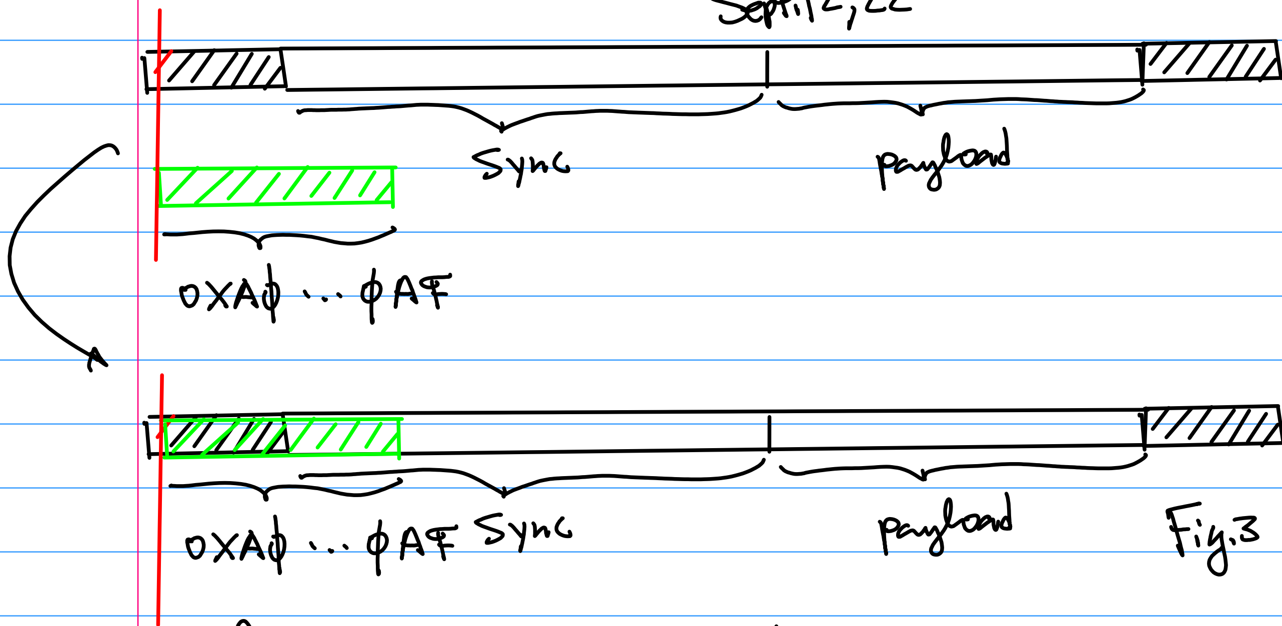


Fig.3

Perform matching operation similar as the one the previous step. if matching is confirmed, then jump to the 1st Bit of the payload Based on the matching index.
O/W. Continue this process (similar as the previous step) by shifting 1 additional Bit to the newer position, repeat the matching process.

This process Continues till the matching is found or the input Buffer (Data) is exhausted.

Note: In terms of the Implementation,
we have an inner "for-Loop," for the "green Region" in Fig3

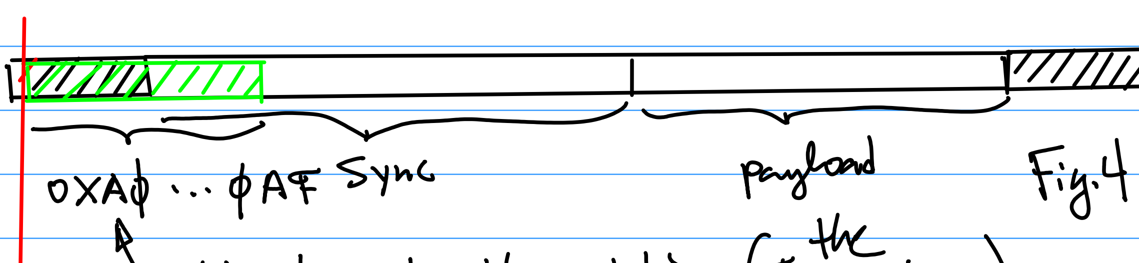


Fig.4

For-Loop for the matching (the Inner Loop).

For the "Red Line" (Newer Position), we can have an outter "for-Loop"

CmpE245

Sept. 12, 22.

3/

The outter loop will continue till the matching is found or the entire Buffer is exhausted;

Note: Reference to Convolution (1D)

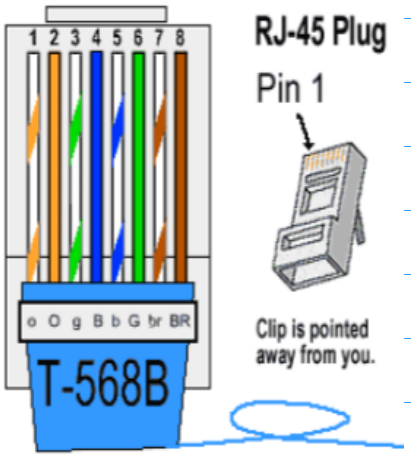
$$\sum_{k=0}^{N-1} h(k)\, g(n-k)$$

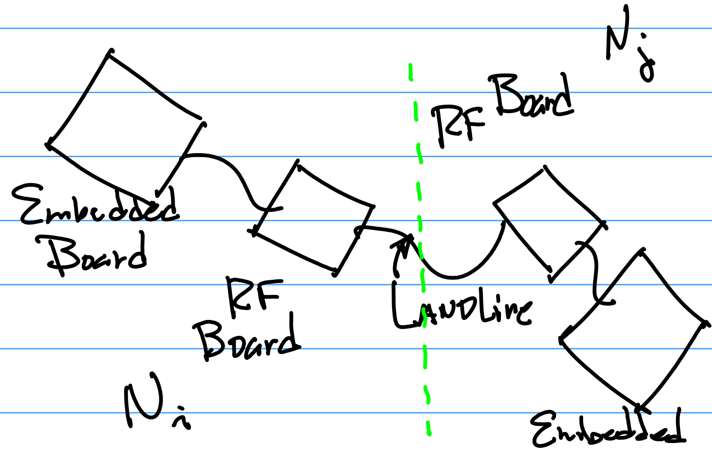
Kernel (e.g. mask)

Its implementation is similar.

Example: Land Line Testing.

Note: 8 pos





Select Pos 1. for Tx (P0.2) N_i ; N_j

Select Pos 3. for Rx (P0.3) N_i ; N_j

Pos 5. GND.

Build One Slide PPT. With this Photo, And Connectivity table

Choose RJ-45 Connector, and CAT5 or CAT6 Cable for Landline.

GPIO is the protocol for the implementation (Not Ethernet).

Hardware Design —— For LPC1769 { P0.2 Output, P0.3 Input }

Software Design —— For NVDA NANO

Physical pin @ the J2 Connector

P1.30	AD0.4	P1.30	J2-19
P1.31	AD0.5	P1.31	J2-20
P0.2		P0.2	J2-21
P0.3		P0.3	J2-22
P0.21		P0.21	J2-23
P0.22		P0.22-RED LED	J2-24

2021F-109-II-note-2021-11-10.pdf

Pin 12
Pin 40

Demo Example Pin-32

Note: J

UI

Sept.14

Homework Implementation.

Software Side { NXP LPC1769, or LPC11xx
NVDA Jetson NANO

Prerequisit:

Hardware Side: Embedded System Prototyping.

{ NXP LPC { 1769 { Prototype Board. 2017F-102-lecLayout 2017-2-7.pdf
11C24 CTIONE Board-B from ebay.
NVDA NAND 2021F-114-gpio-nano-v2-hl-2021-10-20.pdf

Step 1. GPIO Sample Code

{ NXP LPC1769
NVDA NANO

① Sample code from the github
CMPE240-Adv-Microprocessors / 2018S-11-GPIO-2015-1-30.zip

② NxpDeveloper. www.nxp.com
Sign up as a developer.
Down Load, Install MCUXpresso.

③ Config the IDE (MCUXpresso).

④ Down Load LPC1769 Patch.
Buildit.

C/C++ project Semihost CMPE240-Adv-Microprocessors / 1769 patch.zip

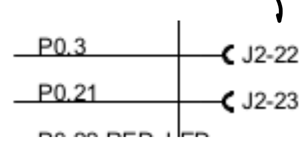
⑤ GPIO Sample code.

CMPE240-Adv-Microprocessors / 2018S-11-GPIO-2015-1-30.zip

To Run the test code, Be sure to have hardware Ready.

CMPE240-Adv-Microprocessors / 2018F / 2022F-101-notes-cmpe240-2022-09-12.pdf

LPCXpresso1769_CD_revD(1).pdf — LPCXpresso LPC1769 CMSIS-DAP rev D.sch



CmpE245
Sept. 21, 22

Sept. 21. Due Oct. 3rd (Monday).

Note: 1° Homework for LISA on the target platform.

Objective:

1. To Implement LISA algorithm on The target platform, e.g. LPC1769, or NVDA NANO.
2. To Establish Communication Between Node i & Node j. By Transmitting the following message:

"SJSU_CmpE245_FirstName_LastName_SID(4 Digits)"

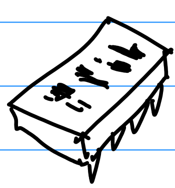
3. The Testing is Land line Testing.

Requirements:

1° To have 2 Nodes Sync'd on the Bit Rate, for Example 1Kbps or lower;

2°. Land line Communication.

3°. Provide LED for Debugging Purpose, tied to GPIO Output Port. using "toogle" or DIP S/W.



Submission:

1° Source Code.

2° Export Project Code for LPC1769 Board;

3° No more than One page Readme Document;

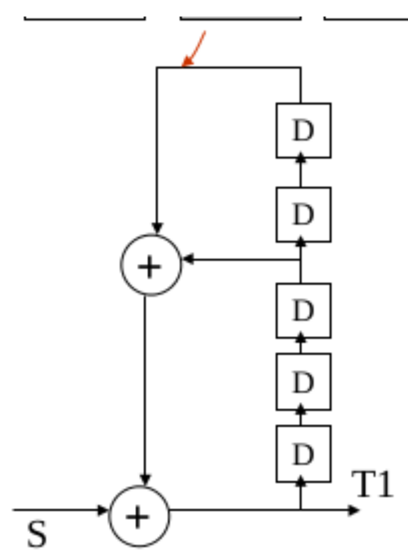
4° photo of Your Testing Environment (Testing System Setup).

5° 10~15 Seconds Video Clips that Shows the program is working.

Note: please Bring Your Board to the Class on the 3rd for quick Show-and-tell;

Submission is on CANVAS.

Example: Scrambler/De-scrambler Design.



Scrambler

Requirements: 1. Design of the Scrambler/Descrambler is required. for Order N, odd

$N = 3, 5, 7, \cdots, 13$, etc.

Design Steps for the Scrambler:

1. Divide the Delay Units (n) into 2 Banks. (By half)

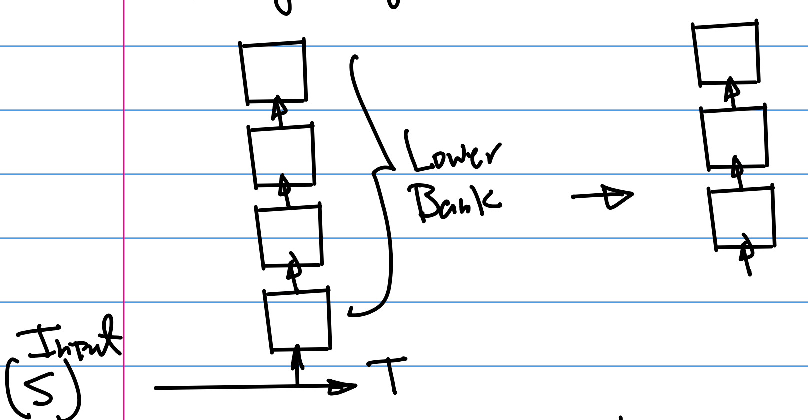
   $\left.\frac{N}{2}\right|_{N \text{ is odd Number}}$ Rounded up

   to define the Lower Bank; Such that it is always 1 order higher than the Upper Bank.

   Example: for $N = 7$.

   $7/2 = 3.5$ Rounded up, $3.5 \to \underline{\underline{4}}$.

   Delay Unit for the Low Bank.



2. Define 2 XOR operators / Processing Units to form feedback Loop.

3. Testing the operation of the Scrambler By Sending:

"SJSU_CmpE245_FirstName_LastName_SID(4 Digits)"

ASCII Coding for Letters & Numerals.

Suppose $S \to$ 0X8F

0X8F

1000 1111 Send Data Out, MSB first

Test By generating the test Table, Similar to Table 5.3 below.

Table 5.3. Input and
of the sc
Figure 5.2

Input S	1 0 1 0 1
D^3T_1	0 0 0 1 0
D^5T_1	0 0 0 0 0
Output T_1	1 0 1 1 1

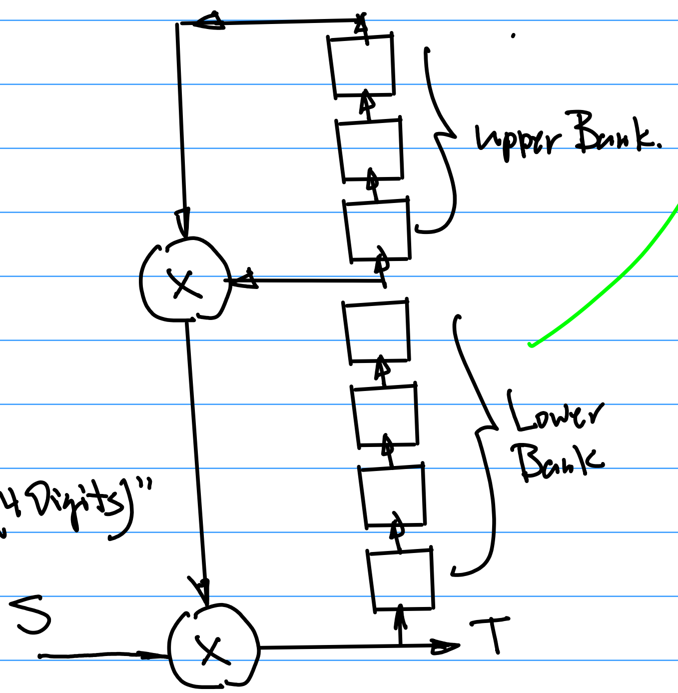
Reference: Di

Input S

Lower Bank. D^4T.

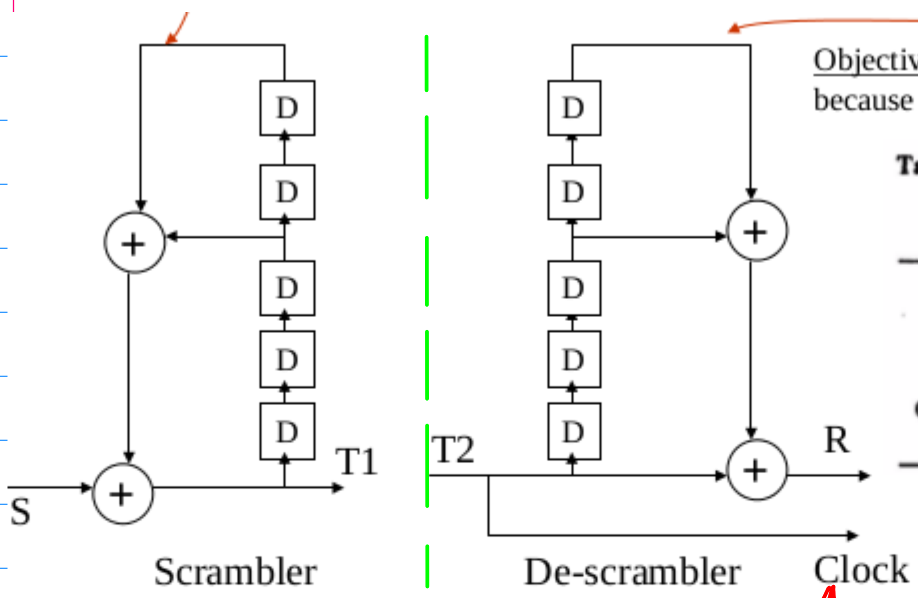
Lower + Upper Bank D^7T

Output T



CmpE245

Sept.21,22



Note: The Theoretical Analysis and proof of the scrambling/Descrambling Technique Can be Accomplished By using Formal math. Formulation from The Switching Theory.

Note: a. Descrambler has "Clock" (Sync Extraction Output)

b. Mirror type of Architecture of Both System, allows Design of De-scrambler to be constructed Accordingly.

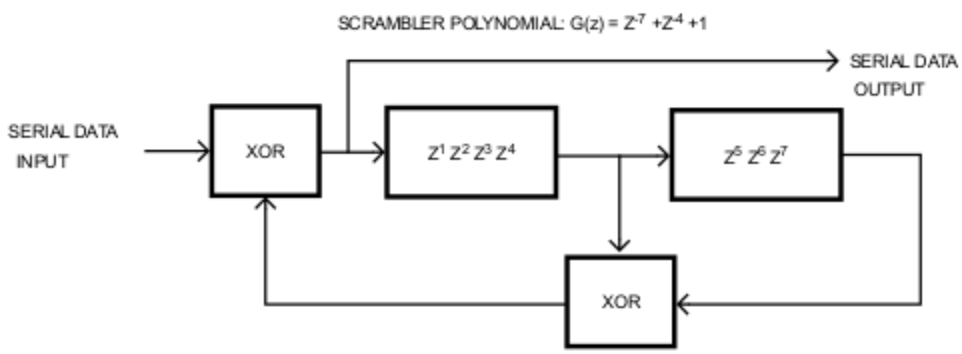


Figure 131—Data scrambler

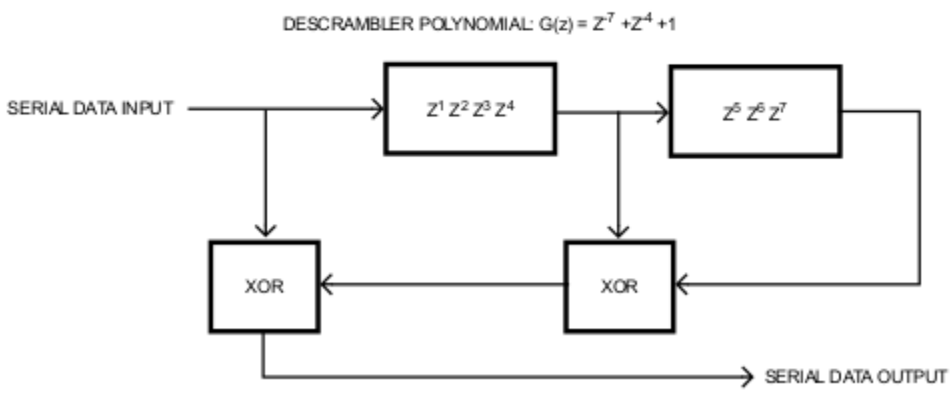


Figure 132—Data descrambler

CmpE258
Sept.21,22

Consider BaseBand Signal Analysis & Formulation.

Motivation: 1. Analyzing the Wifi Communication, Channel Availability & Allocation. a. Power Spectrum of A modulated Base Band Signal is shown in Both Fig.141 & Fig.142.

Figure 141—North American channel selection—non-overlapping

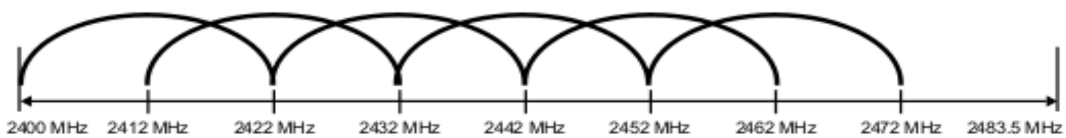


Figure 142—North American channel selection—overlapping

2. Fig145. a. f_c : Carrier frequency for IEEE Wifi. $f_c \sim 2.4$ GHz → 11 Channels. then 11 f_c

b. Base Band Spectrum. Defines the Bandwidth: $f_c + 11$ MHz $- (f_c - 11$ MHz$) = 22$ MHz.

c. 80% of the Energy of WiFi has to be kept Within the Bandwidth.

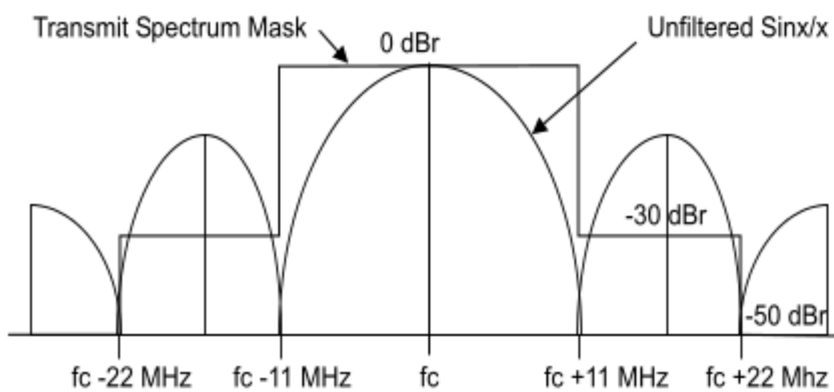
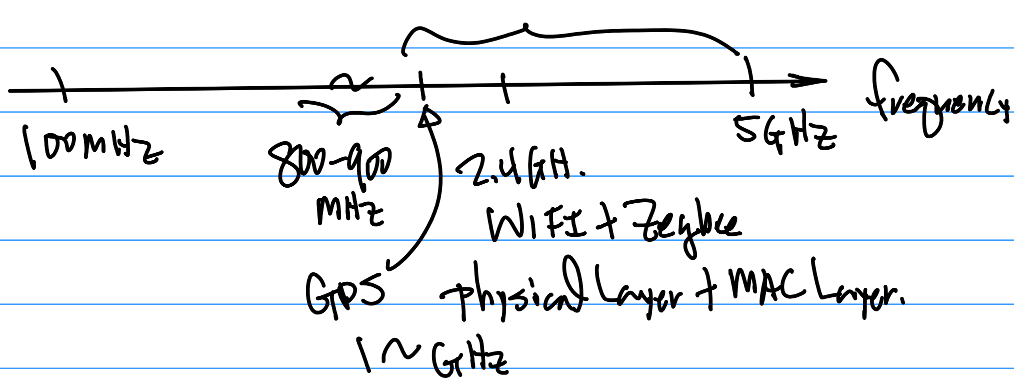


Figure 145—Transmit spectrum mask pp.60



CmpE245
Sept.21,22

Sept.26. Monday.

Note: 1° Check Homework on CANVAS. LISA on the target platform with R.F. Board.

2° Optional Target Platform, NVDA Jetson NANO, 5-6 pieces [ppt]
Sample code have been posted on github.

- 2022F-104-#2021F-114-gpio-connector-sys…
- 2022F-104b-python-gpio-jetson-nano-#202…
- 2022F-105-sd-card-bring-up-nano-2021-10…
- 2022F-106-nomachine-remote-nano-2021-…
- 2022F-107-config#2021F-114b-pwm-nano-…

CMPE245
Sept. 26

Step1. Background, Pin Assignment. J41. Two Pins for GPIO. (Input, Output Each) 10

Pin12	gpio79	input
pin40	gpio78	Output

NVIDIA Jetson Nano J41 Header Pino

https://www.jetsonhacks.com/nvidia-jetson-nano-j41-header-pinout/

Note: I2C and UART pins are connected to hardware and should not be reassigned. By default, all other pins (except power) are assigned as GPIO. Pins labeled with other functions are recommended functions if using a different device tree.

1. take Pin 1 Vcc (3.3V) and Pin 39 GND to test out LED, make sure you can light up a LED with 220 Ohm resistor in series.

Sysfs GPIO	Name	Pin	Pin	Name
	3.3 VDC Power	1	2	5.0 VDC Power
	I2C_2_SDA I2C Bus 1	3	4	5.0 VDC Power
	I2C_2_SCL I2C Bus 1	5	6	GND
gpio216	AUDIO_MCLK	7	8	UART_2_TX /dev/ttyTHS1
	GND	9	10	UART_2_RX /dev/ttyTHS1
gpio50	UART_2_RTS	11	12	I2S_4_SCLK
gpio14	SPI_2_SCK	13	14	GND
gpio194	LCD_TE	15	16	SPI_2_CS1
	3.3 VDC Power	17	18	SPI_2_CS0
gpio16	SPI_1_MOSI	19	20	GND
gpio17	SPI_1_MISO	21	22	SPI_2_MISO
gpio18	SPI_1_SCK	23	24	SPI_1_CS0
	GND	25	26	SPI_1_CS1

a. Pin12

	GND	25	26	SPI_1_CS1	gpio20
	I2C_1_SDA I2C Bus 0	27	28	I2C_1_SCL I2C Bus 0	
gpio149	CAM_AF_EN	29	30	GND	
gpio200	GPIO_PZ0	31	32	LCD_BL_PWM	gpio168
gpio38	GPIO_PE6	33	34	GND	
gpio76	I2S_4_LRCK	35	36	UART_2_CTS	gpio51
gpio12	SPI_2_MOSI	37	38	I2S_4_SDIN	gpio77
	GND	39	40	I2S_4_SDOUT	gpio78

b. Pin40.

Step2. Bring-up the System. Note: Power (Wall mount Adaptor, ~4000mW) 4W

SD Card. 32GB.

Down Load A software → Put/Copy the Pre-Built.
"Flasher" Kernel Image

/ 2022F-105-sd-card-bring-up-nano-2021-10-28.pdf

Write Image to MicroSD Card

NVIDIA

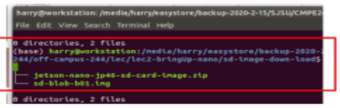
https://developer.nvidia.com/embedded/community

Prerequisite:
1. A micro-SD card (minimum 16GB) and SD card reader with USB interface;
2. A 5V 3A MicroUSB power supply;
3. An Ethernet cable;

Step 1. Down load SD card OS image from Nvidia to your host machine, laptop, the zipped file size is 6.1G, unzip it to get OS image, e.g., *.img file, ref:

https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#write

Harry Li, Ph.D.



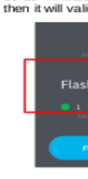
Step 2. Write the image to your microSD card by following the instructions from Nvidia, first you will need to down load the writer software "ethcher" to your host machine from this site:

(2.1) for Linux host, Download, install, and launch Etcher.

https://www.balena.io/etcher/

Use USB card r
your host mach
start it to write

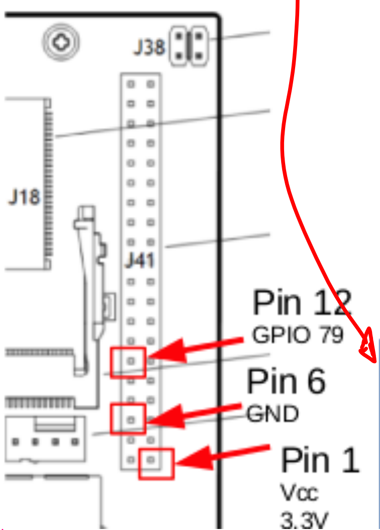
The program '
10-15 minutes
then it will vali

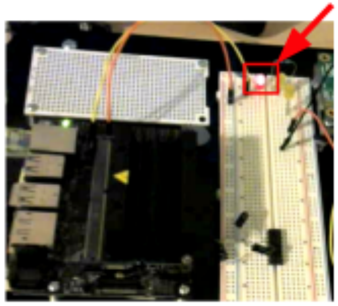


Step3. Test GPIO Input/Output
Using By Command Line
Instruction.
See Ref on the
github, 2022F_104~

Command Line Information.

Testing J41 40 Pin Connecto





We can control our LED

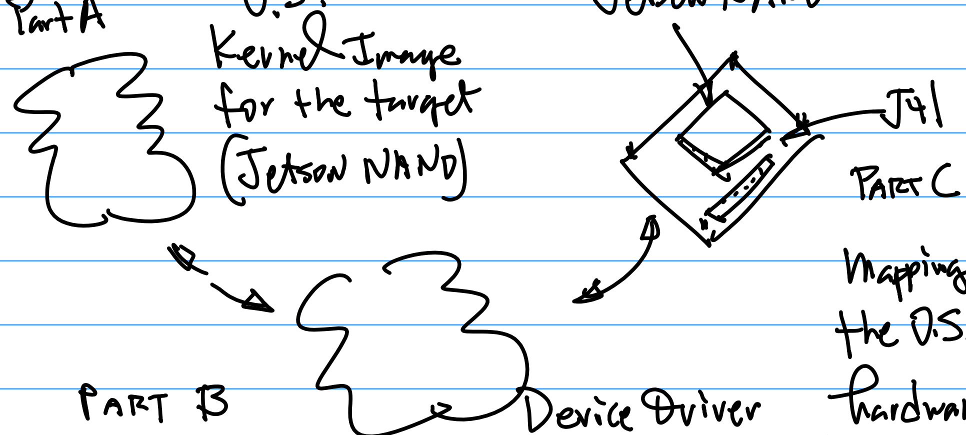
```
# Map GPIO Pin
# gpio79 is pin 12
$ echo 79 > /sys/c
# Set Direction
$ echo out > /sys/
# Bit Bangin'!
$ echo 1 > /sys/cl
$ echo 0 > /sys/cl
# Unmap GPIO Pin
$ echo 79 > /sys/c
# Query Status
$ cat /sys/kernel/
```

In the above code, the
look at the Jetson Nano
header.

```
$echo 79 > /sys/class/gpio/export
$ echo out > /sys/class/gpio/gpio79/direction
$echo 1 > /sys/class/gpio/gpio79/value
$echo 0 > /sys/class/gpio/gpio79/value
```

Step 5. In order to Program GPIO Port. First Chose High Level Programming Language, Such as Python or C++. Python is recommended. Then, to Be Able to program device drivers.

Note: please choose Ubuntu 18.04 for the target platform.



Mapping of Device Driver(s) from the O.S. Kernel to the target hardware is done by "Configuration" process.

CMPE245-Embedded-Wireless / 2018F / 2022F-107-config#2021F-114b-pwm-nano-v3-hl-2021-11-9.pdf

CMPE245

Sept. 26, 22

13

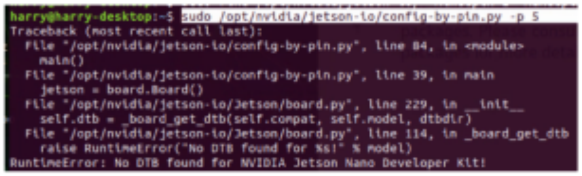
Note: Run Configuration Python Code if Pins Added Note By Factory Default.
Recommended To use Configuration Mapping.

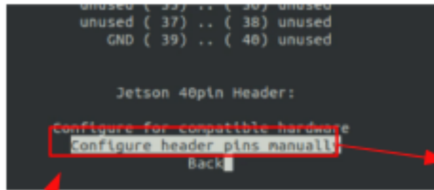
Step 1. Fix bugs from the distribution

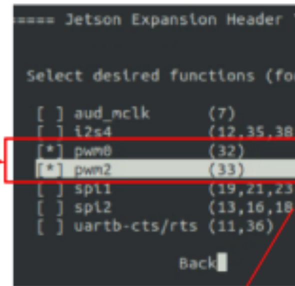
Configuration of Pins with jetson-io.py

$sudo find /opt/nvidia/jetson-io/ -mindepth 1 -maxdepth 1 -type d -exec touch {}/__init__.py \;

$sudo /opt/nvidia/jetson-io/config-by-pin.py -p 5

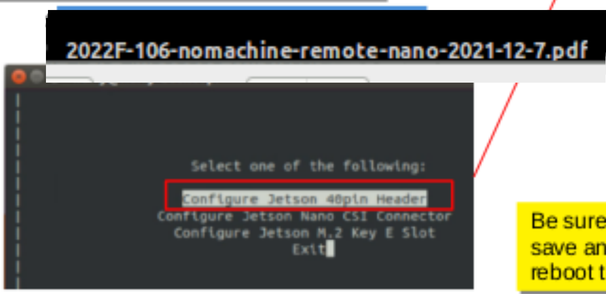


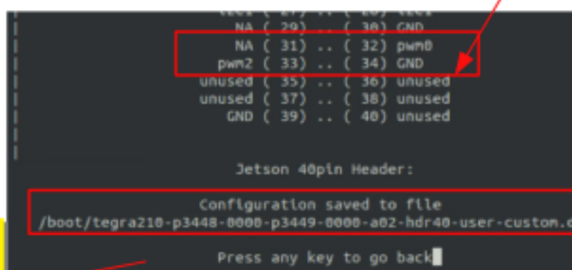




$sudo mkdir -p /boot/dtb
$ ls /boot/*.dtb | xargs -I{} sudo ln -s {} /boot/dtb/

Step 2. Run jetson-io.py to configure pins





Be sure to choose save and reboot to reboot the system

Harry Li, Ph.D.

Step 6. Setup Remote Access for the purpose of using your laptop KeyBoard, Mouse, And Display.

2022F-106-nomachine-remote-nano-2021-12-7.pdf

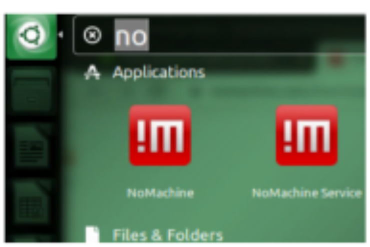
Nomachine for Jetson NANO

https://www.nomachine.com/download/download&id=116&s=ARM

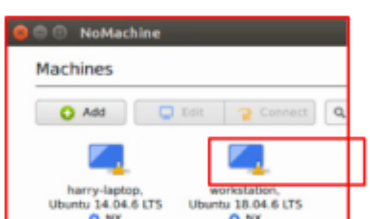
Aarch64 version 7.74_1; size: 42.29 MB, type: TAR.GZ

Follow nomachine website info for installation, I have installed it in my \Document\NX folder, you could installed it in \usr\NX folder

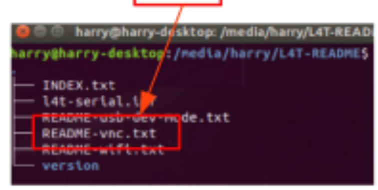
On nano after the installation, you can see this



Once you start nomachine on your laptop, and enter the right user name and password of the nano, you can see it now



Note: From NVDA L4T installation on NANO, you can see VNC installation readme, below

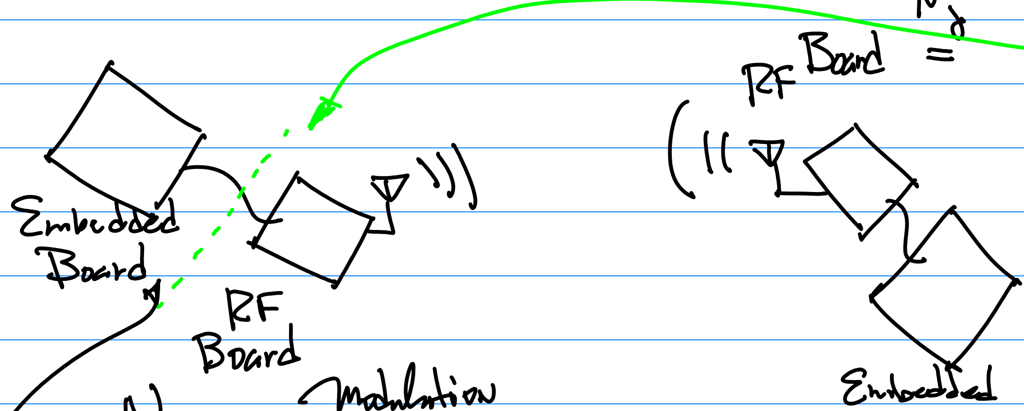


CMPE245
Sept. 26, 22

Continued for the Base Band Signal Discussion.

Example: Communication B/W N_i & N_j

" SJSU_CMPE245" To Be Sent. → ASCII → "0's & "1", Base Band Signal



N_i

Modulation
"Transport" the Base Band Signal to a proper frequency Range for Desired Characteristics.

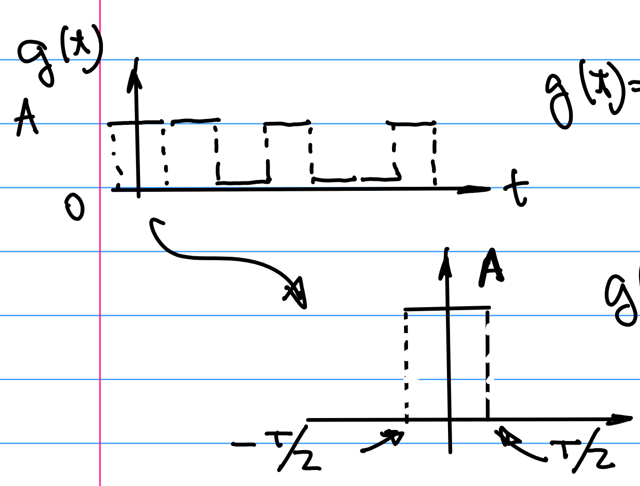
This Process (modulation) will → Satisfies 80% Energy distribution Requirements. By IEEE/FCC.

a. The waveform "Shape" of the Base Band Signal is well designed to meet the 80% energy Requirement.

b. The "Transporting" (e.g. modulation) is most effective and provide "Robustness" (e.g. Resisting to Random Noise)

Example: Base Band in Time Domain.

Given a sequence of A B.B. (Base Band) Signal, in Figure 1.



$$g(t) = \begin{cases} A & t \in [t_0 + T/2,\ t_0 - T/2] \\ 0 & \text{o/w} \end{cases} \quad \ldots (1)$$

$g(t - kT)$, for $k = 0, 1, 2, \ldots$

For A sequence of BB:

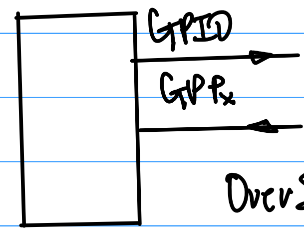
$$\sum_{k=0}^{N} g(t - kT)$$

Define Fourier Transform

$$F[g(t)] \triangleq \frac{1}{T}\int_T g(t)\, e^{j2\pi f t}\, dt \quad \ldots (2)$$

$g(t) \longleftrightarrow \mathcal{F}[g(t)]$

$$\mathcal{F}[g(t)] = A_T \frac{\sin \pi f_T}{\pi f_T} \quad \cdots (3)$$



1024 bit/Sec.

OverSampling.

LPC1769 GPP Reading is 5X1024
OR
NANO

Sept 28 Wed.

Note 1°. Regarding the Homework on Landline testing of LISA.

Buffer of 1K or 2K bits for the Homework Can be implemented.

Step 1. Read from GPP_x @ Clock Rate 5X Bit Rate → Step 2 for Every 5 Consecutive Read, Voting Mechanism is employed to get one Bit

Random Data

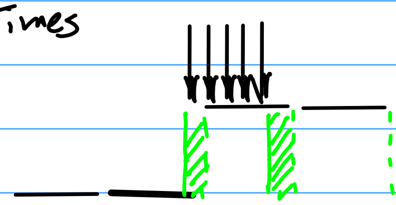
Sync

payload

Step 3. fill in the 1K Buffer, Continue till the Buffer is filled.

Step 4. Start "LISA" Parsing to establish Sync.

2° Consider Using Over Sampling. e.g. for each bit, Read 5 Times

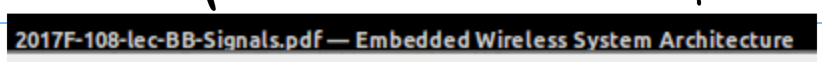


"Voting", majority Samples will be Considered as the actual Bit Value.

↓

GPIO Reading, 5x faster than the Bit Rate. for bit Rate = 1 KbPS (= 1024 bits/Sec.) → Need Clock with Interrupt to define the GPIO Reading.

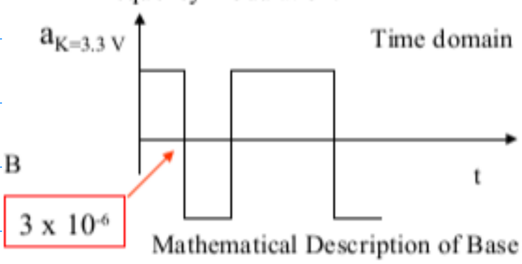
Example: BaseBand Signal Analysis



CmpE245
Sept. 28, 22

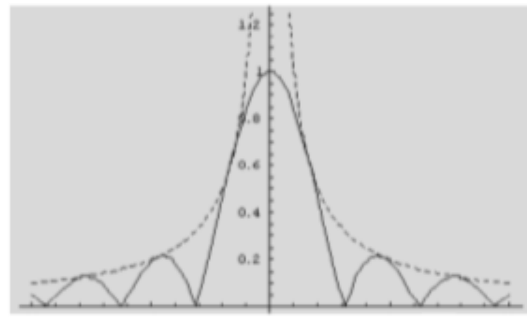
Base Band Signal

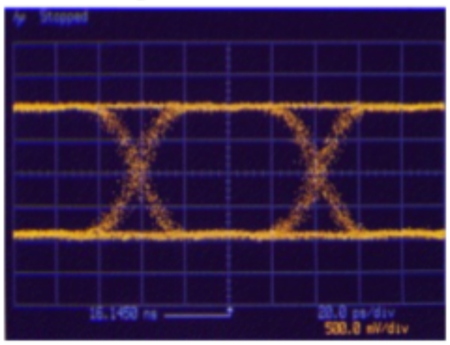
Definition: Signal transmitted near zero frequency range without frequency modulation.



Mathematical Description of Base Band Signal, see lecture notes.

Frequency domain





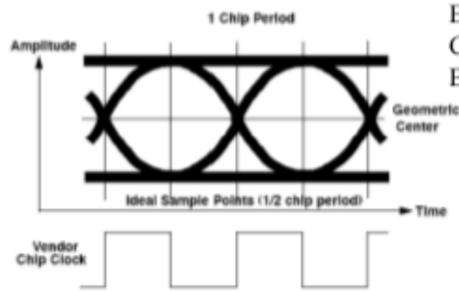


Figure 149—Chip clock alignment with baseband eye pattern

Eye Pattern: Characterization of Base Band Signal

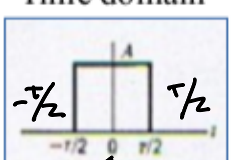
IEEE 802.11b Standard pp. 56

Harry Li, Ph.D. SJSU, CMPE 242/296AA

Base Band Signal Formulation

a. T one period, one Bit

Time domain



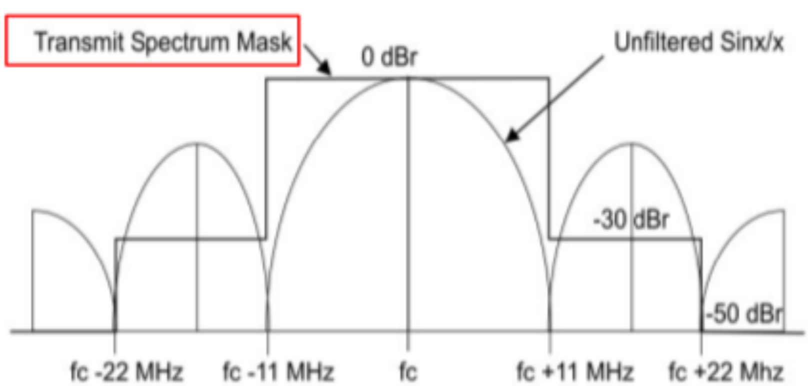
$$g(t) = \begin{cases} A & \text{for } [-T/2, T/2] \\ 0 & \text{otherwise} \end{cases} \quad ...(1)$$

$$\boxed{A\tau \frac{\sin \pi f \tau}{\pi f \tau}} \quad ...(2)$$

Example: Calculation of bandwidth

Let equation (2) = 0, then we have
Pi * f * T = k * Pi
Let k=1 for the first pair of zero crossings, we have

f = 1/T



Frequency domain

Figure 145—Transmit spectrum mask from IEEE 802.11 b, pp. 60

Counting both sides of the spectrum, we have

$$\boxed{BW = 2/T} \quad ...(3)$$

b. $f_b = \frac{1}{T_b}$ frequency

$BW = 2 f_b$

Consider the Definition / Derivation of the Bandwidth (BW).

From BB Signal in Eqn (1), pp 3.
We perform F.T. in Eqn (2).

$$A_T \frac{\sin \pi f T}{\pi f T}$$

We can define a Power Spectrum of the Signal By using its F.T.

$$P(f) = \sqrt{Re^2[F(f)] + Im^2[F(f)]} \quad \cdots (3)$$

Where $F(f) \triangleq \mathcal{F}[g(t)]$
Fourier Transform of $g(t)$.

Real Part of the F.T.
$Re[F(g(t))]$
Imaginary Part:
$Im[F(g(t))]$

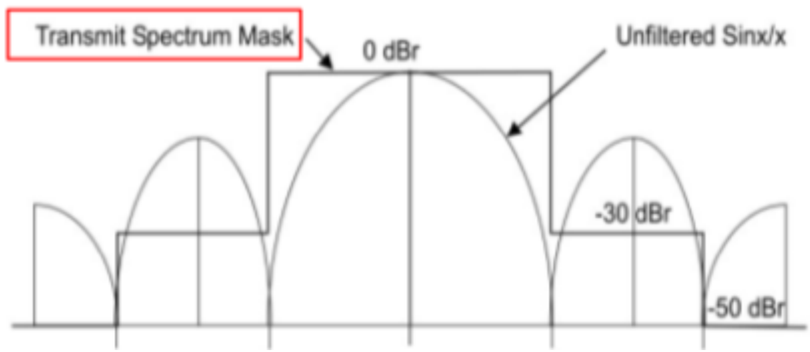
Note: $g(t) \longleftrightarrow \mathcal{F}[g(t)] = F(f)$

Note for the F.T. of the BaseBand Signal $g(t)$, it is the following,

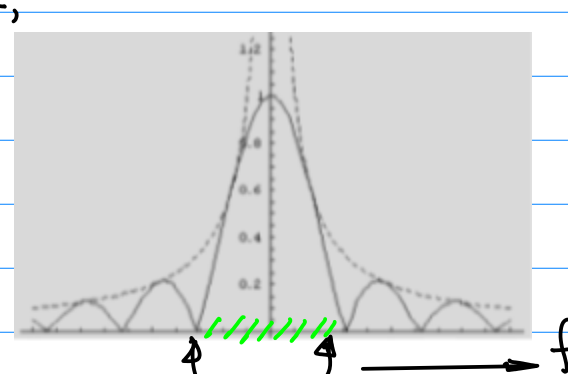
$$A_T \frac{\sin \pi f T}{\pi f T}$$

which only has the Real Part, therefor its F.T and Power Spectrum is the Same.

Plot the power Spectrum.



OR,



Define the Bandwidth of the Base Band

Note:

$$\frac{1}{T}\int_T g(t) e^{-j2\pi f t}\, dt$$

where

$$e^{-j2\pi f t} = \cos 2\pi f t - j \sin 2\pi f t$$

(Euler Formula)

Let $A_T \dfrac{\sin \pi f T}{\pi f T} = 0$

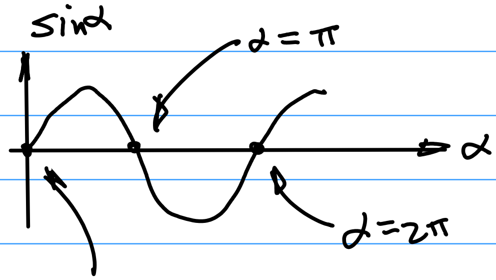
$\downarrow$

$\cancel{A_T} \dfrac{\sin \pi f T}{\pi f T} = 0$

$\downarrow$

Multiplying $\pi f T$ By Both Sides of the Equation

$\sin \pi f T = 0$



$\sin \alpha = 0 \Rightarrow \alpha = 0$

Hence,

$\pi f T = K\pi$

for $K = 0, 1, 2, \dots$

$\cancel{\pi} f T = K \cancel{\pi}$

$f = K/T$

↑

frequency Index, where $K=1$ for the 1st Z.C.

$f = 1/T$

Now, the other side in the Frequency or Range, we have $f = -1/T$ for the 1st Z.C.

To find the Bandwidth, we have

$$B.W. = 1/T - (-1/T) = 2/T \quad \dots (4)$$

Example: Find the Bandwidth of the Signal in Our Homework.

Sol:

$B.W. = 2/T$

$T = 1/f$, f is the Bit Rate which is 1Kbps.

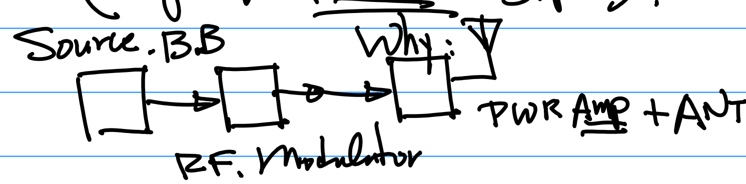
$f = 1024$ Hz

$T = 1/1024$

$\therefore B.W. = 2/T = 2f = 2048$ Hz

Observation 1: B.W and Bit Rate are Connected by Equation (4). To Double the Bit Rate, By Eqn (4), we have to Double the Bandwidth, which is NOT desirable or practical. From Now on, we will try to Develop Technology to increase the Bit Rate while Keep the Same Bandwidth.

Consider The Tool for Analyzing the Energy Distribution of A given R.F. (e.g. of A Base Band Signal).



CmpE245
Sept. 28. 22

Define D.F.T. for this purpose:

$$X(m) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi \frac{mn}{N}} \quad \ldots (5)$$

Oct 5.

Q & A Session.

Timer/Timing

Background On Time Design.

System Clock → PCLK → Counter Design. Count By N_1

PCLK: Peripheral Clock

↓

Another Counter, Count By N_1

Match Register. ⇐ $f = \frac{f_{PCLK}}{N_1 N_2}$

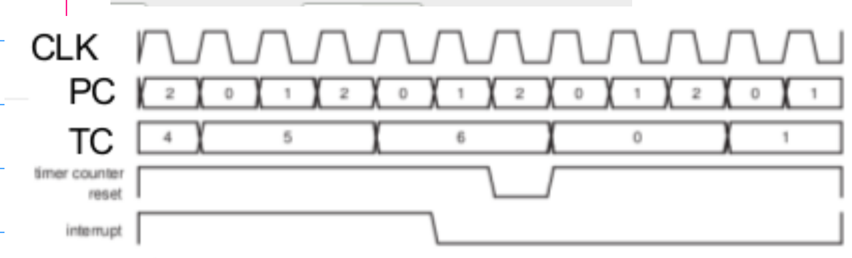
More Granulation/Resolution.

$Reg_m = N_1 N_2$

↓

Trigger An Interrupt. (Waveform, pp. 3)

2017F-120-INT-TIMER-Part2-v3-HL-2017-12-4.pdf



1. Note: SPRs.

PR: Prescaler Register. TC: Timer Counter.
PC: Prescale Counter. MR: Match Register

(3) Set Prescaler

Note: 4 important registers, PR, PC, TC, and MR

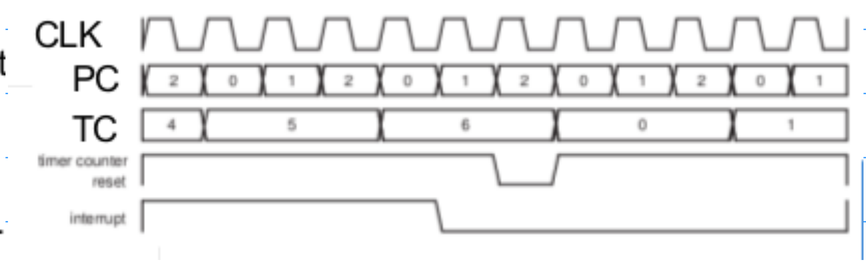
Two steps: first, PCLKSEL register, Peripheral Clock Selection register, Timer0 clock; Section 4.7.3, pp. 57;

Bit	Symbol	Description
1:0	PCLK_WDT	Peripheral clock select
3:2	PCLK_TIMER0	Peripheral clock select
5:4	PCLK_TIMER1	Peripheral clock select

2nd, LPC_TIM0->PR Prescaler register, When the Prescale Counter (PC) is equal to PR, Timer Counter (TC) is incremented by 1;
e.g,
TC is incremented every PR+1 cycles of PCLK.

2. Algorithm.

Example:
Given PR = 2, MR = 6; An interrupt generated on match.
(1) at every clock, PC -> PC+1;
(2) when PC = PR, TC -> TC+1 ;
(3) when TC = MR, INT generated.



PR: Prescaler Register. TC: Timer Counter.
PC: Prescale Counter. MR: Match Register

Suppose $f_{PCLK} = 20\,MHz = 20 \times 10^6\,Hz.$

$f_{INT} = 1 \times 10^3\,Hz.$

$$f_{PCLK} = K \cdot f_{INT} \quad \cdots (1)$$

where $K = N_1 \cdot N_2 = N_{PR} \cdot N_{MR} \quad \cdots (2)$

Hence,

$$f_{PCLK} = N_{PR} \cdot N_{MR} \cdot f_{INT} \quad \cdots (3)$$

$$f_{PCLK} / f_{INT} = N_{PR} \cdot N_{MR}$$

$$\frac{20 \times 10^6}{1 \times 10^3} = N_{PR} \cdot N_{MR}$$

$$2 \times 10^4 = N_{PR} \cdot N_{MR}$$

32 Bit Register

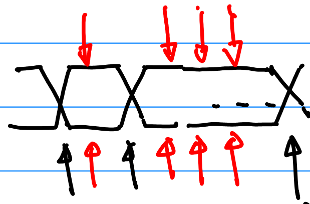
Assume 8-bit minimum usable [0, 255]

Let $N_{PR} = 255$

$$N_{MR} = 2 \times 10^4 / 255 \simeq 78$$

Breadboard

B.B. ~5 MHz | Wire Wrapping ~20 MHz | Soldering ~50 MHz



— Uncertainty.

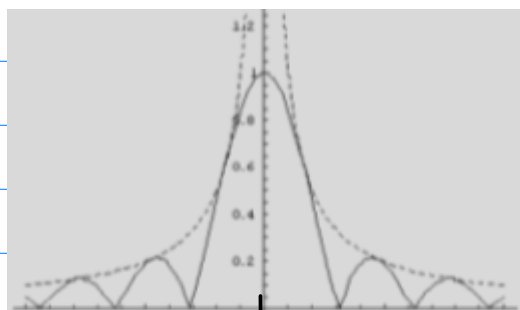
— Good Timing

Starting pt for
Capturing the Data
@ Rx End

N_i (Tx) $\qquad$ N_j (Rx)

f_{CLK_i} $\qquad$ $f_{CLK_j} = 5 f_{CLK_i}$

Example: Power Distribution / Power Spectrum



Continuous F.T.

Sampling Theorem.

$$f_{Sampling} \geq 2 f_{Max}$$

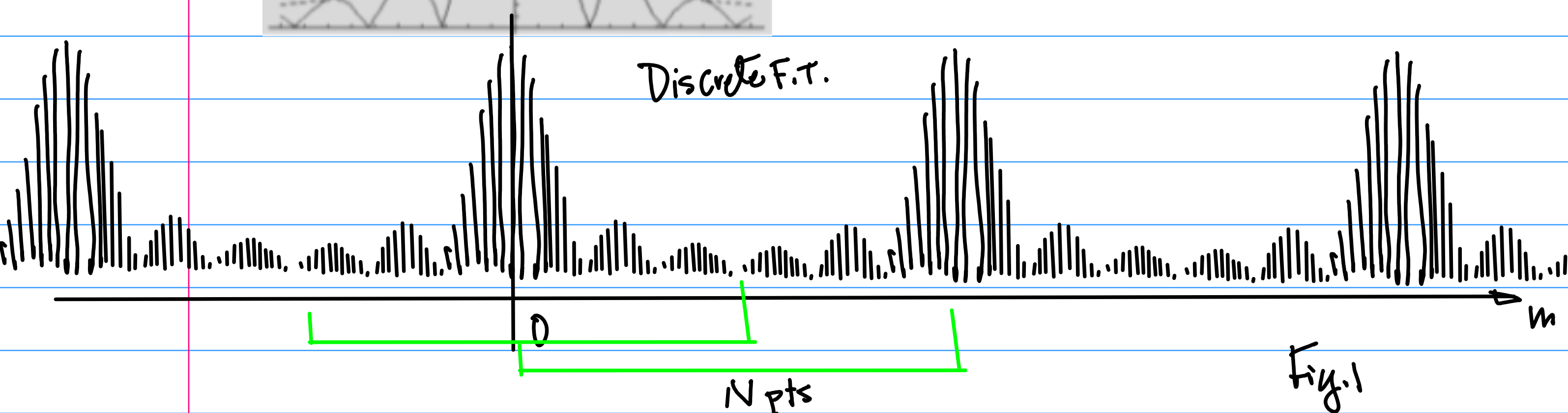


Fig. 1

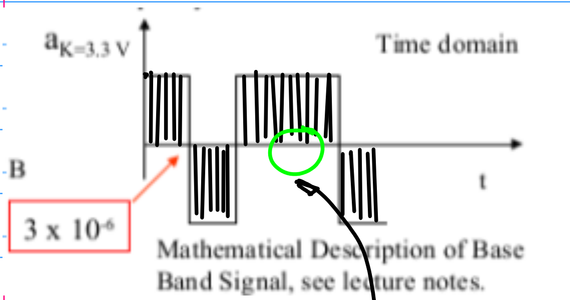
Oct. 10 (Monday).
Note: 1. In-Class Homework
Show & Tell, Demo, Debugging.

Theoretical Background on Power
Spectrum Interpretation
1. Nyquest Sampling Theorem.

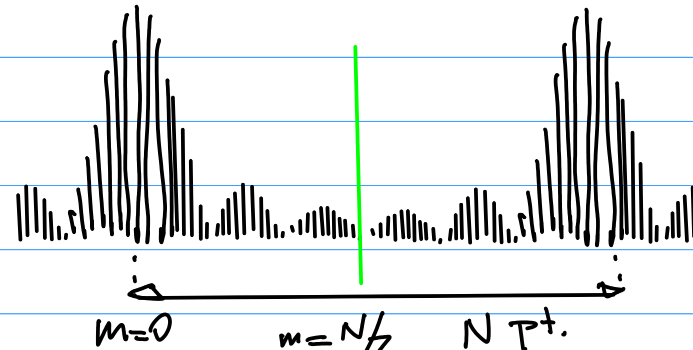
$$f_{Sampling} \geq 2 f_{max} \quad \cdots (1)$$

The Sampling Speed $f_{Sampling}$ has to greater than or equal to Twice the Highest Frequency of the signal.

Note: 1.



Samplings By a Microprocessor.



Sampling Time $\Delta t_{Sampling} = 1/f_{Sampling}$ ...(2)

$f_{Sampling} = 1/\Delta t_{Sampling}$

2. from the Power Spectrum Illustration in Fig.1, pp20, it is a Periodic function, the period is equal to N.

$P(m) = P(m+KN)$, ... (3)

for $K = 0, 1, 2, \ldots$

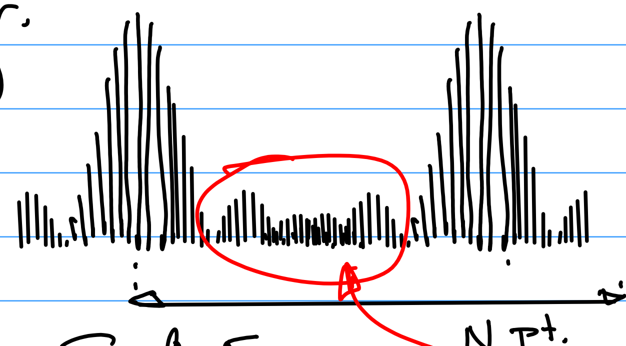
N is the Number of the pts. in One Period;

3. Focus our Discussion on One period, that $m = 0, 1, 2, \ldots, N-1$;

4. $m=0$, $P(m)|_{m=0}$ is D.C. Component. The Highest Frequency Index $m = N/2$

For FFT Computation $N = 2^x$

5.



Sampling Frequency $f_{Sampling}$ is too low.

∴ Distortion. "Alising"

Oct.12 (Wed). Due Oct.30 (Sunday)

Note: 1° Project on Wireless (RF) Communication on Sync. (LISA Implementation). 10pts

1.1 Two Nodes, N_i and N_j for Rx, Tx Communication. Sending/Decoding the message (Same as the Landline Homework). Make Sure to have adequate Distance Between Rx & Tx.

1.2 Capture up to 30 Second Video to Demonstrate the Success of the Communication of the 2 Nodes.

1.3. Photo of the Entire System Setup. Laptop, Prototype Board, RF Board with

CmpE245

Oct.12,22

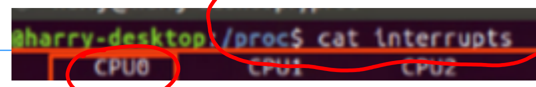
22

Rx and/or Tx.

1.4. In the RF Board Implementation, Be sure to Keep/Have the Landline Testing Circuit;

1.5 One screen Capture of your IDE or Software Execution Environment, And show program executed Successfully, with your Personal Identifier, such as a file folder with your Name.

1.6 Update the status ON the Semester Long project (Using LORA module). Option to Consider: WIFI is acceptable, if using WIFI module, please Visit me during office Hours.



e. Sample code C++ Reading, Build A Test Circuit LED on/off

f. Communication Between C++ and Python, Based on TCP/IP.

Note: Need to Download Kernel Source, J.P. 4.5.1 (For Example, Ubuntu 18.04 O.S. Packages for DCNN, T.F.) 2019.

Run $ menuconfig

Note: Midterm is Scheduled on the 31st, Monday.

Example: ON NANO a. Ref; b. Datasheet, TRM (Technical Requirements pdf 4000 Pages)

Timer Interrupt Service Routine on Na

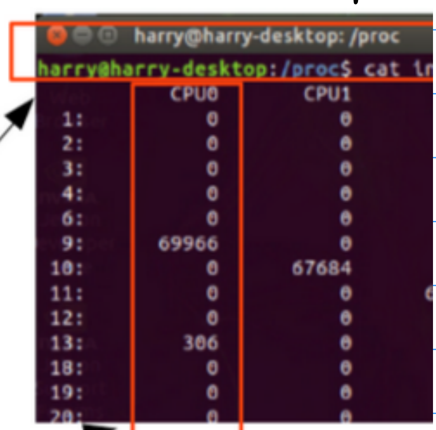
https://forums.developer.nvidia.com/t/timer-interrupt-service-routine-using-in-jet

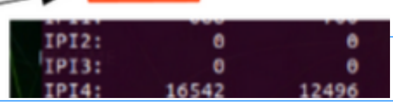
c. Chapter 8 Timer

d. NANO 2G

1. Jetson Nano already uses TMR10, TMR11, TMR12 and TMR13.
2. So I want to use TMR0 for ISR.
3. What is the IRQ number for TMR0 and how to change TMR0 frequency.
4. If I want to use ISR in kernel space, I should call "request_irq" function. In "request_irq" function, first argument is IRQ number as I know, but the datasheet does not seem to have this information.
5. How do you modify the TMR0 frequency?

1. The default ISR rate is 1000 Hz. A long time ago the default in Linux (and most operating systems) was 100 Hz. The method for changing it was (and probably still is) via a kernel feature (selecting 100 Hz or 1000 Hz and then build a new kernel and install it). I don't know if this has evolved to allow changing this dynamically.





2. Many IRQ numbers are listed in "/proc/interrupts". Note that many hardware devices can only be routed to the first core, CPU0 (software IRQ can be serviced by any core, but the hardware must have access to

Cmpe245
Oct.12,22

Example: Frequency in D.F.T.

Given D.F.T as Follows.

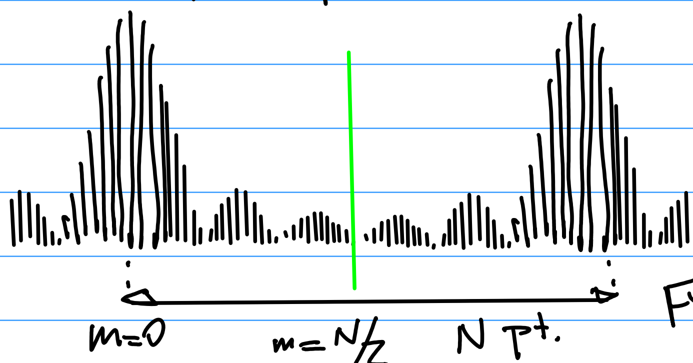
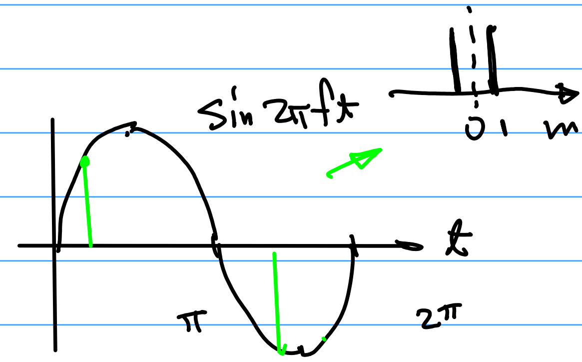
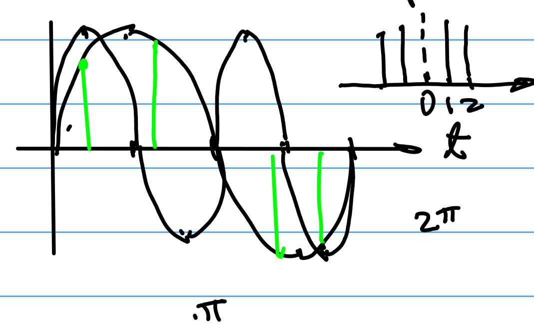


Fig.1

$f_{Sampling} = 2\,KHz$



100% Information of the Signal is Captured by the Sampling result.



Note:
1° Frequency Range of the Base Band Signal → Sampling Freq. $f_{Sampling}$

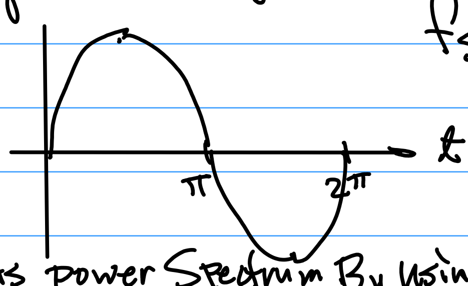
One period N → Sampling Frequency.

2° Highest Frequency of the Signal Being Sampled is $f_{max} = f_{sampling}/2$

The Signal may have the freq Beyond what was Sampled (if Nyquest Sampling was not Implemented).

3° To Test out the Concept Above,

a. generate a Signal



Compute its power Spectrum By using FFT Code on the Class github.

New Sampling Frequency.

$f'_{Sampling} = 2(f_{Sampling})$

Homework ON Power Spectrum Computation. (Due One Week, Oct. 19)

1° Laptop as a platform;

2° Use the Sample C Code Posted on the Class github.

$f_{Sampling} \geq 2 f_{max}$

f_{max} foundamental Frequency.

CMPE245

Oct. 12, '22

Sample Code

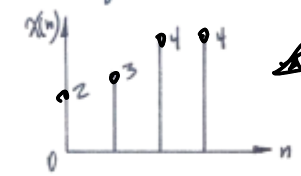
a. Add "0" for Porting from FORTRAN to C.

b. $N = 2^x \big|_{x=4} = 16$ pts for Sampling Observation of frequency & Visualization.

```
int main(void)
{
        float arr[16] = {0.0, 2.0, 3.0, 4.0,
                         4.0, 0.0, 0.0, 0.0,
                         0.0, 0.0, 0.0, 0.0,
                         0.0, 0.0, 0.0, 0.0};
```

CMPE245-Embedded-Wireless/2018F/2019F-101-#2pwr-fft.c

1) Given A Discrete Signal $x(n)$ as follows, find its D.F.T.



Sol

By Definition 1D D.F.T. is

Hint: The P(m) of the Signal in 6, should looks like the following.

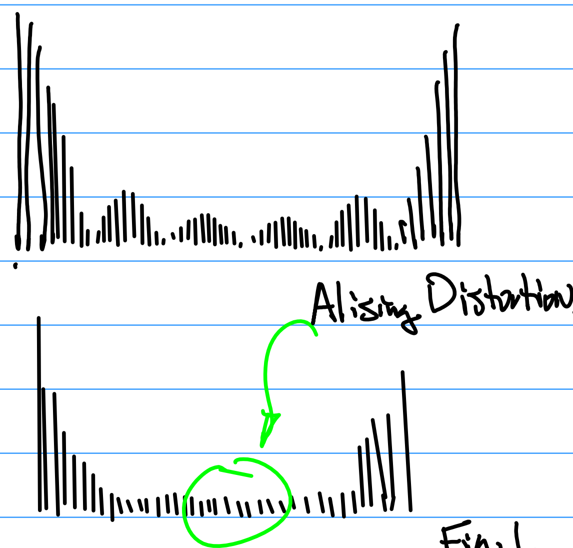


Fig. 1.

Continued from the Homework Requirements.

3. $N = 2^x$, for Example $N = 256$ OR 1024.

4. Compute FFT, then the power Spectrum $P(m)$, Plot your Result.

5. The Signal should be:

a). $\sin 2\pi f t \rightarrow$ Discrete Signal

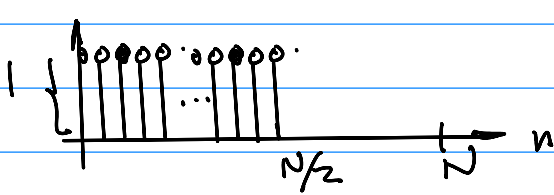
$\sin 2\pi \frac{mn}{N}$

b) $\sin 2\pi \frac{mn}{N} + \sin 2\left(2\pi \frac{mn}{N}\right)$

c) $\sin 2\pi \frac{mn}{N} + \frac{1}{2} \sin 2\left(2\pi \frac{mn}{N}\right)$

6. Then, Base Band Signal with

$$x(n) = \begin{cases} 1 & 0 \le 1 \le N/2 \\ 0 & O/W. \end{cases}$$

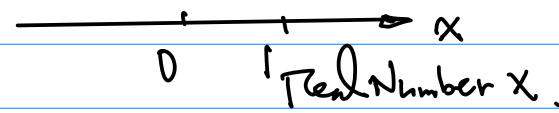


Oct. 17 (monday)

Note: 1° midterm, Scheduled on Nov. 7th (monday), In-person. Submission onLine to CANVAS; Please Bring (1) your Laptop; (2) Your Prototype System; Execution of the code is required; (3) Blank Printer Papers are Needed, take photos, convert to pdf, then Submission.

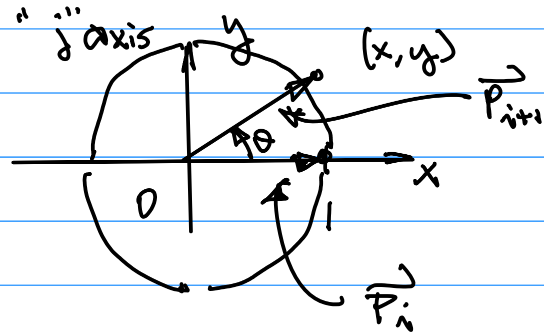
2° Homework on FFT (One week, Oct. 24), On CANVAS.

3° Project 1 (LISA), 10 pt.
Oct 31 (Monday), Submission to CANVAS.



Example: Energy Distribution Index

$$\eta = \frac{\left(\text{Energy of the Signal up to the frequency } m'\right)}{\left(\text{Total Energy of the Signal}\right)} \quad \ldots (1)$$



For $2+j4 \rightarrow$ Mag $= \sqrt{2^2+4^2} = \sqrt{4+16} = \sqrt{20}$

$$\theta = \tan^{-1}\frac{4}{2}\left(\frac{Im}{Re}\right) = \tan^{-1} 2$$

gives the total Energy of the Signal. ... (2)

$$\sum_{m=0}^{N-1} P(m) = \underset{\text{D.C}}{P(0)} + P(1) + P(2) + \cdots + P(N-1)$$

where $P(m) = \sqrt{Re^2[X(m)] + Im^2[X(m)]} \quad \ldots (3)$

And

$$X(m) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi\frac{mn}{N}} \quad \ldots (4)$$

Euler Equation <u>EULAR</u>

$$e^{-j2\pi\frac{mn}{N}} = \cos 2\pi\frac{mn}{N} - j\sin 2\pi\frac{mn}{N} \quad \ldots (5)$$

Energy of the Signal up to the frequency Index m'.

$$\sum_{m=0}^{m'} P(m) \quad \ldots (6)$$

Note:

Re : Real Part of A Complex function, or A Complex Number.

Example: Given 2+j4, then

$Re[2+j4] = 2$, And
$Im[2+j4] = 4$.

Hence, Eqn(2) & (6) together,

$$\eta = \frac{\sum_{m=0}^{m'} P(m)}{\sum_{m=0}^{N-1} P(m)} \quad \ldots (7)$$

where m' as a frequency index for the Base Band Signal.

CmpE245

Oct. 17, 22

Note: To find m' for the Base Band.

1° N period $\rightarrow f_{sampling}$

2. $f_{sampling} \geq 2 f_{max}$ Nyquest Theorm.

3. $f_{sampling}$ is given;

4 Base Band Bandwidth Is given; $f_{BB} \rightarrow m'$

Example: Suppose we have $f_{sampling} = 1$ KHz; Suppose B.W. = 500Hz (Assume $f_{sampling} \geq 2 f_{max}$). And N = 1024

Calculate energy index By Eqn (?)

Sol

Total Energy: $\sum_{n=0}^{1023} P(n)$;

$$m' = 1024 / (f_{sampling} / BW)$$

where

$$f_{sampling} / B.W = \frac{1000}{500} = 2$$

hence $m' = 1024/2 = 512$

So

$$\sum_{m=0}^{511} P(m).$$

Therefore

$$\eta = \frac{\sum_{m=0}^{511} P(m)}{\sum_{n=0}^{1023} P(n)}$$

"

Oct. 24. (Monday)

1. Nov. 2nd Midterm Exam. 30%.

(1) Printer Paper (2) Prototype System (3) Smart Phone for photo taking purpose to upload Answer Sheets. ; (4) One page Formula is permitted ;

~3 Questions.

Subject Area: Energy Distribution index η; Power Spectrum ;

Today's Topics:

1° Semester Long Project Planning. LORA Module for Industrial IoT Applications.

Homework (0 pts) Update the plan and Status of LORA project preparation. Due 1 week from Today. Oct 31.

Cmpe 245

Oct. 24, 22

Example: a. SX127x Family; b. Wider Frequency Range, 137MHz ~ 1 GHz (1.02 GHz)

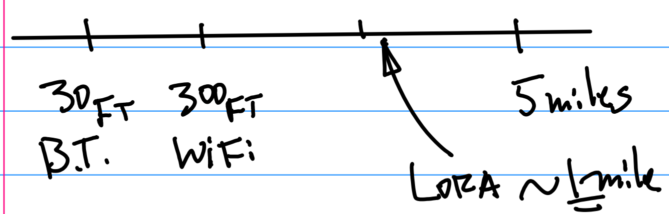
https://www.semtech.com › wireless-rf › lora-connect

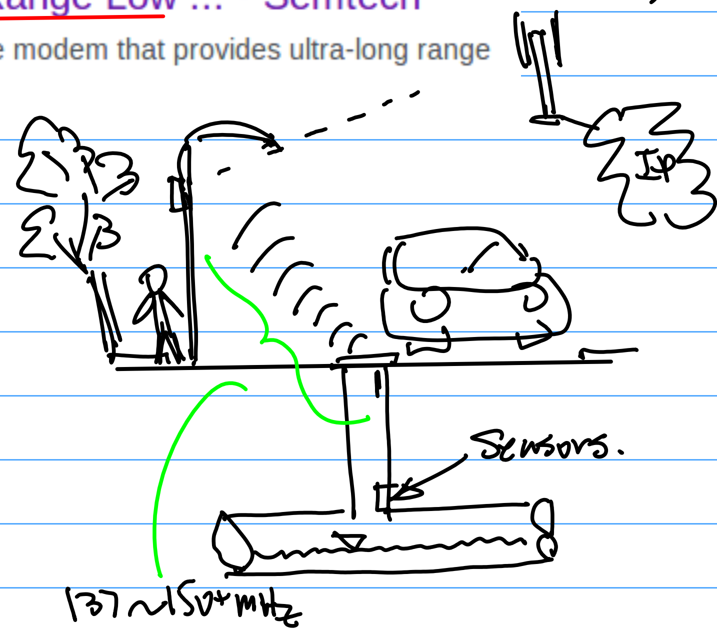
SX1276 | 137MHz to 1020MHz Long Range Low ... - Semtech

The SX1276 transceiver features the **LoRa** long range modem that provides ultra-long range

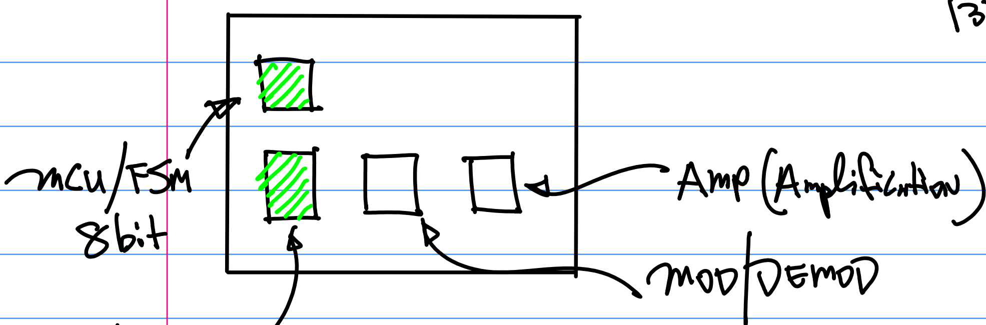
C. Long Range, Reference: Bluetooth ~30FT;

Wifi ~300FT; Cellphone ~5 miles





Comparison Between LoRA & ASK Module



C.R. (Cognitive Radio)
Communication Capability.

Selection/Evaluation Matrix:

1° Freq Range: 137 MHz ~ 1.02 GHz;

2° SPI or I2C

Existing Sample Code ON CmpE245 Class Github.

3° Cost/Performance: ~$10-15

4° Scalability into Real Applications.

1. Architecture Aspects:

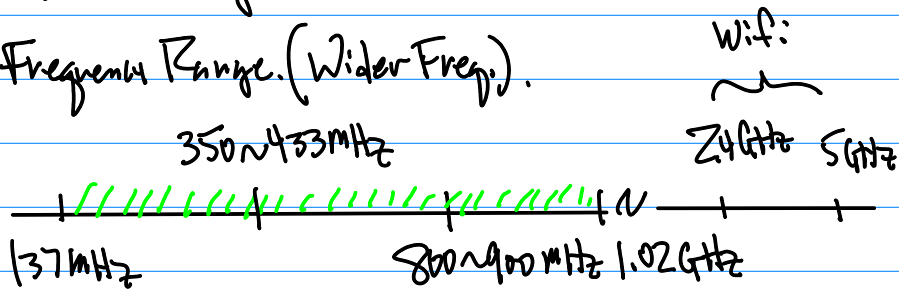
MAC (Media Access Layer)

Such as: Sync., Scrambling/Descrambling, Error Detection/Correction; Freq Hoping;

V.S. ASK module.

2. Distance. Longer distance.

3. Frequency Range. (Wider Freq.).



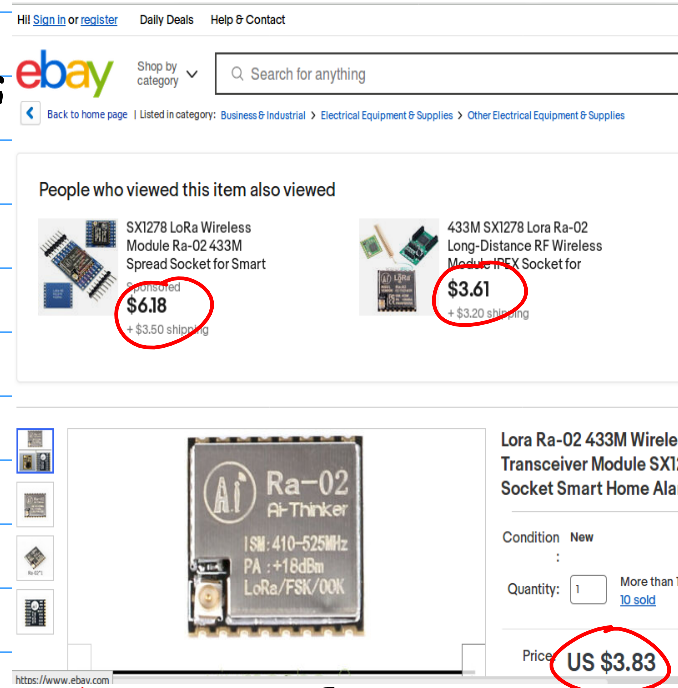
CmpE245

Oct. 24, 22

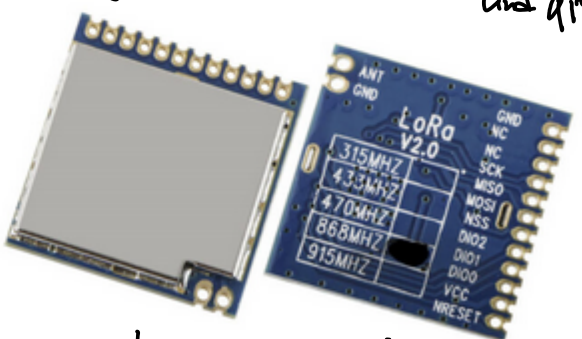
https://ozrobotics.com/shop/lora1276-c1-868-ce-certified-spi-interface-100mw-sx1276-868mhz-4km-wireless-transceiver-modules/?gclid=EAIaIQobChMI9qzmrPH5-gIVVQytBh3QoQzXEAQYBiABEgIuZ_D_BwE

Example: Design for the semester Long Project with LoRA.

Ref: 2018F-116-lec8-lora-RF.pdf



Promissing: a. Freq Range. 315, 433, 470, 868, and 915 MHz;



b. SPI I/F MISO, MOSI, SCK

c. GPP pins for the Test I/F.

d. Sx1276 Chip. Sample Code Compatible. e. Cost

Lora1276-C1-868 CE Certified SPI Interface 100mW SX1276 868MHz 4KM Wireless Transceiver Modules

$16.50

- 2PCs LoRa1276-C1-868 868MHz Modules
- 2PCs 868MHz Spring Antennas

Note: for LPC1769 { SPI, GPP

Sx1276 SCH		LPC1769 PIN	
RFI_LF(P.1)	SCK	SCK1 ✓	J6-7
VR_ANA(P.2)		VIO3V3	
VBAT1(P.3)	SI	MOSI1 ✓	J6-5
VR_DIG(P.4)		GND	
NRESET(P.7)		SSEL1 ✓	J6-8
DIO0(P.8)	SO	MISO1 ✓	J6-6
DIO2(P.10)		GPIO (P0.2)	J6-21
RFO_HF(P.22)		VIO3V3	
VBAT2(P.24)		GND	
GND(CON_P.32)		GND	
VDD_FEM(CON_P.34)		VIO3V3	

Harry Li, Ph.D.

2018F-116-lec8-lora-RF.pdf

https://www.ebay.com/itm/153541209648?chn=ps&norover=1&mkevt=1&mkrid=711-117182-37290-0&mkcid=2&mkscid=101&itemid=153541209648&targetid=1262779894729&device=c&mktype=&googleloc=9032168&poi=&campaignid=14859008593&mkgroupid=130497710760&rlsatarget=pla-1262779894729&abcId=9300678&merchantid=108817104&gclid=EAIaIQobChMI9qzmrPH5-gIVVQytBh3QoQzXEAQYCSABEgImNPD_BwE

For Connector J6, See Next Page from the SCH.

Note: Pin Connectivity: { SPI (4 pins), GPP (1 pin), PWR/GND (6 pins)

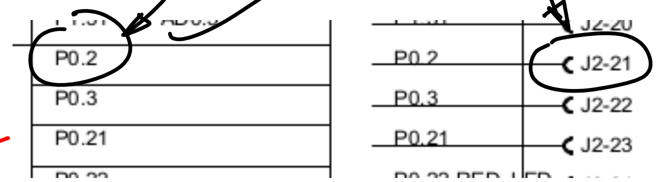
CMPE245

Oct.24,22

https://github.com/hualili/CMPE240-Adv-Microprocessors/blob/master/2018F/LPCXpresso1769_CD_revD(1).pdf

	LPCXpresso			Dual row holes (2x27), 1(
	GND		GND	J2-1
	VIN (4.5-5.5V)		EXT_VIN	J2-2
	VB (battery supply)		VBAT	J2-3
	RESET_N		TARGET_RESET	J2-4
	P0.9	MOSI1	P0.9	J2-5 J6-5
	P0.8	MISO1	P0.8	J2-6 J6-6
	P0.7	SCK1	P0.7	J2-7 J6-7
	P0.6	SSEL1	P0.6	J2-8 J6-8
DA	P0.0	TXD3/SDA1	P0.0	J2-9
CL	P0.1	RXD3/SCL1	P0.1	J2-10
	P0.18	MOSI0	P0.18	J2-11
	P0.17	MISO0	P0.17	J2-12
TX	P0.15	TXD1/SCK0	P0.15	J2-13
	P0.16	RXD1/SSEL0	P0.16	J2-14
	P0.23	AD0.0	P0.23	J2-15
	P0.24	AD0.1	P0.24	J2-16
	P0.25	AD0.2	P0.25	J2-17
	P0.26	AD0.3/AOUT	P0.26	J2-18
	P1.30	AD0.4	P1.30	J2-19
	P1.31	AD0.5	P1.31	J2-20

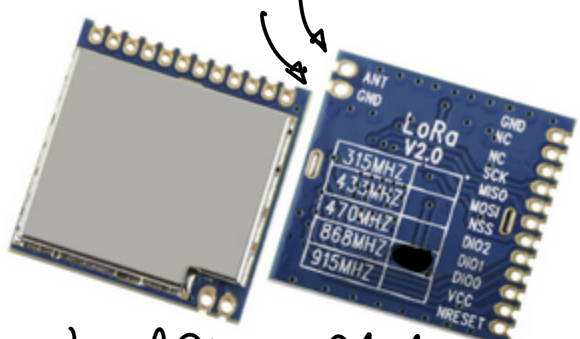
Note: P0.2



Suppose Design with OZRobotics Module (Sx1276)

ANT: 23Cm

GND

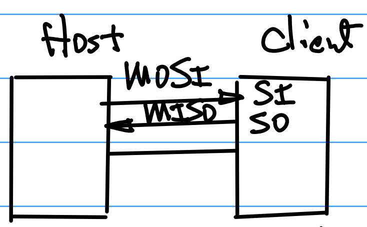


ANT physical Dimension Calculation is Based ON 433 MHz.

Note: Document Your Design with PPT. (1 slide for Pin Connectivity Table).

Oct.26(W).

Example: Design Guideline for LORA R.F. Communication.



Target Platform. LORA R.F.

https://www.nicerf.com/articles/detail/antenna-length-frequency-and-wavelength.html

$L_{ANT} \triangleq \lambda/4$ ... (1)

where λ is the Wave Length.

$\lambda = C/f$ ... (2)

C Speed of the Light 3×10^5 m/s; f: frequency in MHz

Wavelength = 300,000 kilometers/435MHz

= 300000000/435000000

= 300/435

= 0.69 meters

$L_{ANT} = \lambda/4 = \frac{0.69m}{4}$

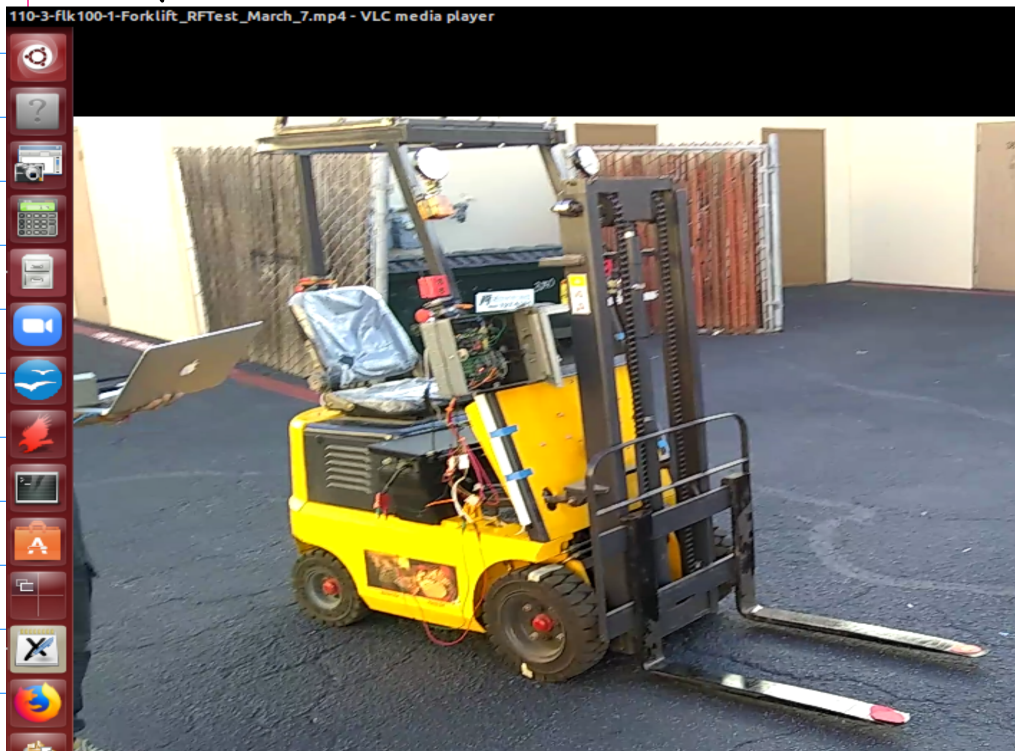
$= 17.25$ Cm

CmpE245

Oct 24, 22

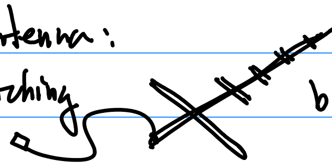
Example with R.F. Communication.

30

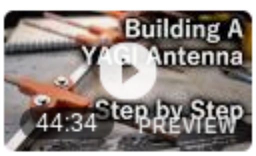


Note: Yagi Directional Antenna:

a. Input Impedance Matching



b. Physical Dimension



Building A Directional Yagi Antenna for 144 VHF or 433 UHF

YouTube · Concentric Machining

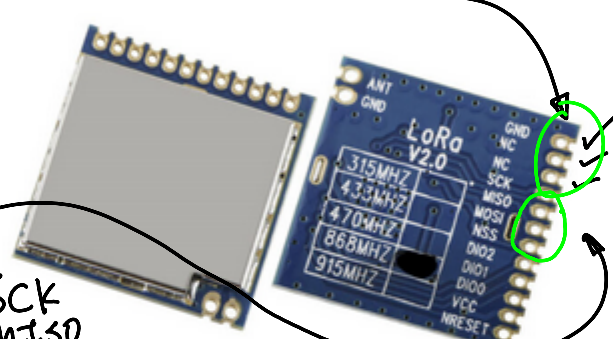
Dec 14, 2020

c. Gain.

d. Connector SMA, OR SMB Type for Better Impedeme Matching.

Now, GND : Build Common GND.

NC : No Connection, Leave it float;



SCK
MISO
MOSI
NSS (? "N", Negation, Active Low.
"SS" Serial (SPI) Select ??)

CmpE245

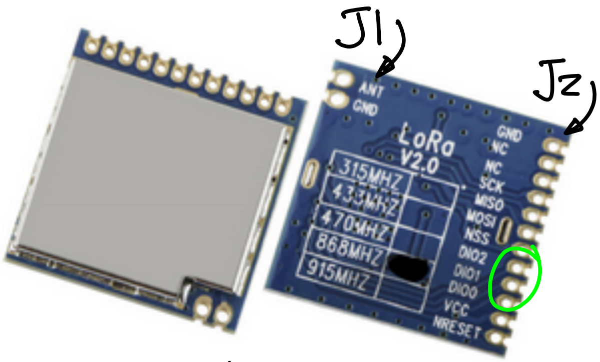
Oct. 26, 22

31

Note: DIO0, DIO1, DIO2 : 3 GPP GPIO.

Now, for Vcc 5VDC ??
Check Datasheet

NRESET: Reset. P2.13 (J6-27)
N ← Negation Active Low.
Additional GPIO pin



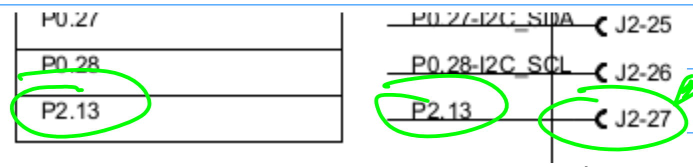
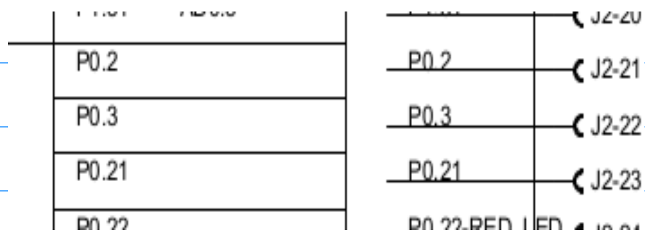


Table 1. Connectivity Table for LoRA SX1276

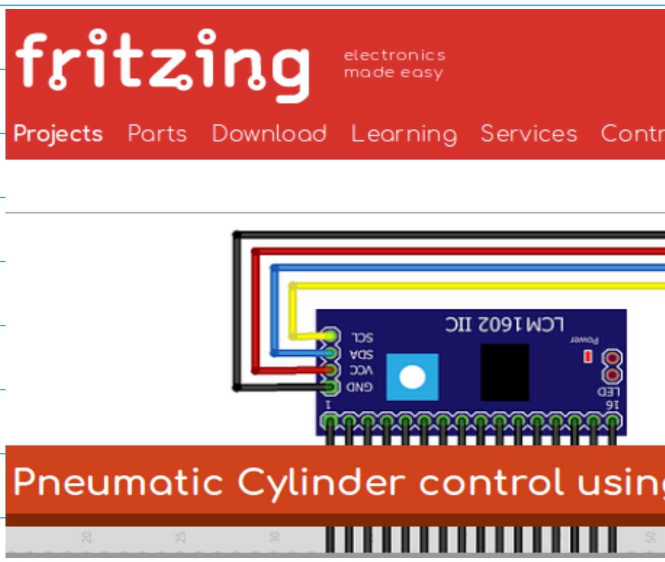
For target platform
LPC1769. P0.2 (J6-21) — DIO0 (J2-10)
P0.3 (J6-22) — DIO1 (J2-9)
P0.21 (J6-23) — DIO2 (J2-8)



LPC1769 LoRA

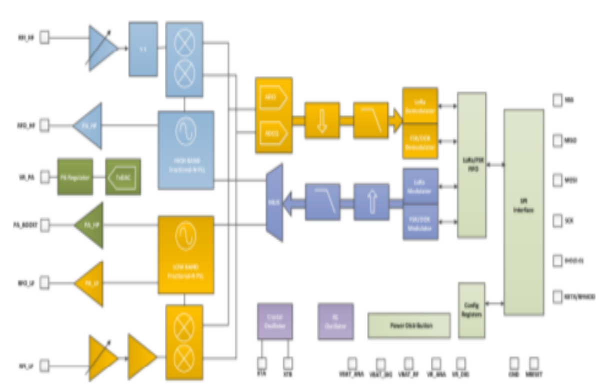
ANT (NC)
MOSI P0.9 / J6-5
MISO P0.8 / J6-6
SCK P0.7 / J6-7
nSSEL P0.6 / J6-8

https://fritzing.org/



Example: LoRA Datasheet

sx1276_77_78_79.pdf



Fussion360 On-Line

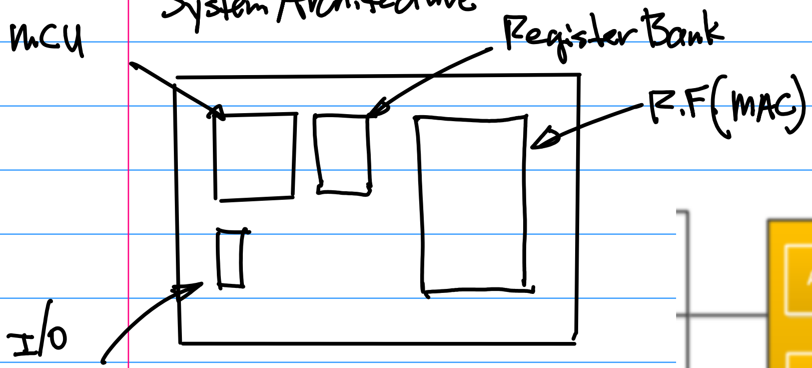
https://accounts-staging.autodesk.com/logon?resume=%2Fas%2F80WAa%2Fresume%2Fas%2Fauthorization.ping&spentity=null#username

Cmpe245

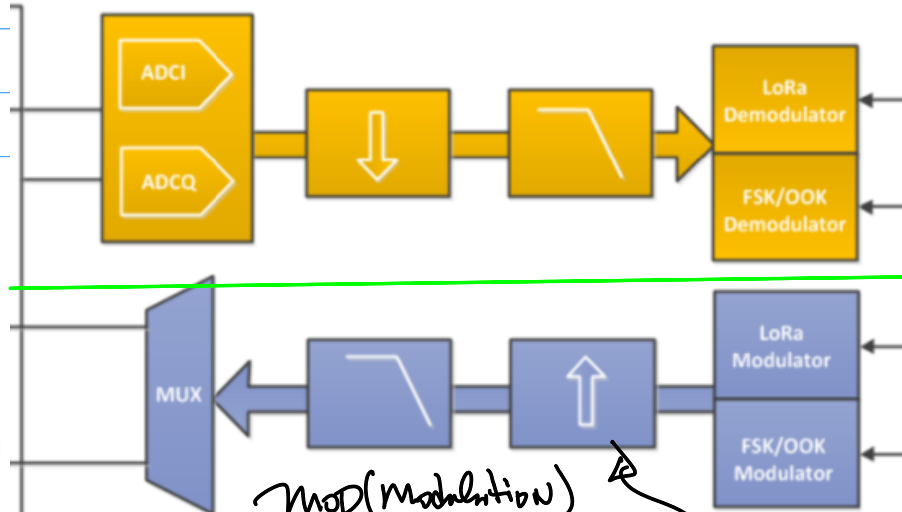
Oct.26,22

32

System Architecture



R.F. SubSystem: { Tx
Rx



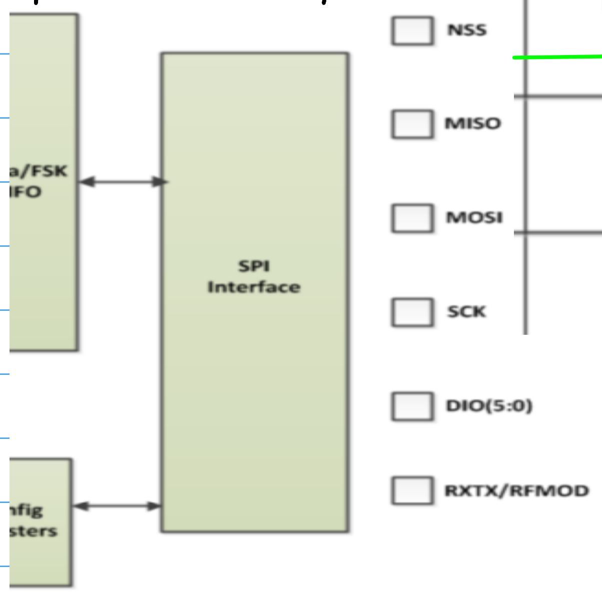
MoD (Modulation)

Move Base Band Signal to the Desired Freq. Range, to Gain Efficiency and Robustness.

Freq. Doubling, Trippling

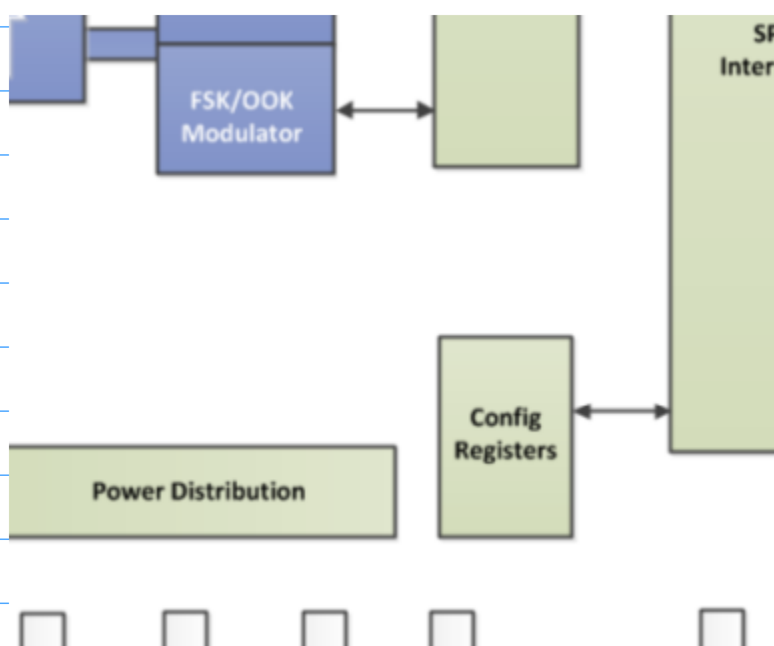
X

a. I/O Block: SPI I/F.



b. Register Bank, Configuration Registers.

Special Purpose Registers { CoN: (Control/Config)
DR.: (Data Register).



Oct.31 (Monday).

1. In Class Project Demo.
2. Review Session for the coming midterm on Wed (2nd Nov.).

1° 3 Questions.

About

Design, Algorithmic Description; LISA (Sync.), Scrambling/De-Scrambling; BaseBand Signal, Bandwidth, Energy Index; Power Spectrum, CmpE 245

Oct. 31, 22

D.F.T, FFT;

2° Bring Prototype System and the Laptop with the code of your Homework, projects. (Power Spectrum, FFT, LISA)

3° Hardware Design Question:

a. Target platform, pin Connectivity; Schematics of the Board Implementation;

b. Please Bring Your cellphone to take photo of your System;

4° Screen Capture of the program execution may be required. Have screen Capture Software tool Ready;

5° Theoretical Questions, such as Power Spectrum, Hand Calculation is required;

6° Need Internet connection.

7° Bring Printer Papers to write your answer;

8° One page Formula is allowed, But No. Calculation Examples or Verbal Explanation.

IEEE 802.11b ⇒ Sync Field → LISA

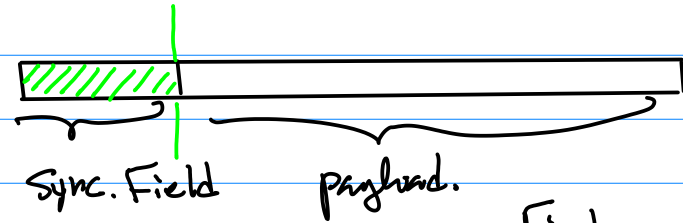


Fig.1

Payload: Timing Extraction / Recovery.

7th order By IEEE.

System Block Diagram.

Note: Scrambling / De-Scrambling are only Applied to the payload;

Base Band Signal, $A_T \frac{\sin \pi f_T}{\pi f_T}$

Bandwidth; $B.W = 2/T$

D.F.T, Power Spectrum.

$$P(m) = \sqrt{Re^2[X(m)] + Im^2[X(m)]}$$

$$X(m) = \frac{1}{N}\sum_{n=0}^{N-1} x(n) e^{-j2\pi\frac{mn}{N}}$$

Matrix Formula ✓

$$e^{-j2\pi\frac{mn}{N}} = \cos 2\pi\frac{mn}{N} - j\sin 2\pi\frac{mn}{N}$$

Energy Index η.

Nov. 7 (Monday).

Note: 1° midterm Exam Key to Be Posted on CANVAS By today;

2° LoRA module. Alternative on AMZN. SPI I/F Recommended, UART is also Available.

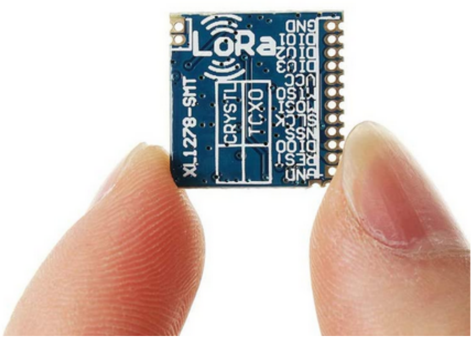
CmpE245

Nov. 7, 22

34

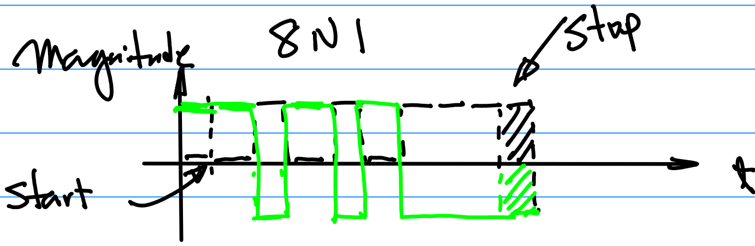
https://www.amazon.com/Comimark-SX1278-Wireless-Mental-Arduino/dp/B07W6ZPH7D/ref=sr_1_4?keywords=lora+module&qid=1667862533&sr=8-4

- Super Anti-jamming (Channel Suppression Ratio: 56db)
- High Receiving Sensitivity - 139 dbm. (32M Passive 10ppm Crystal) ISM multi-band, no need to apply for free use of frequency.
- Optional multi-frequency, multiple transmission rates. It can be used in FDMA and FM technology.
- Intelligent reset, low voltage monitoring, timed wake-up, low power mode, sleep mode
- Low Power Receiving Current: 10-12 mA 256-bit FIFO TX/RX



For RS232

Fig. 1b



Logic "0" → [3.3 ~ 18 VDC];

Logic "1" → [-3.3 ~ -18 VDC];

Distance is short.

PSK → BPSK → QPSK
↳ DBPSK → DQPSK → CCK

RS485.

Modified: { Differential Pair is added. Removal "GND" pin in Communication.

Has multiple client.

Extended distance. By Changing the Freq. Characteristics of the Signal.

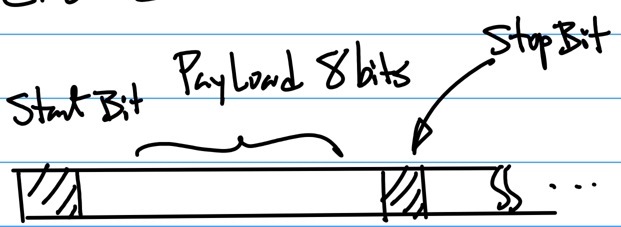
Example: Modulation & Demodulation

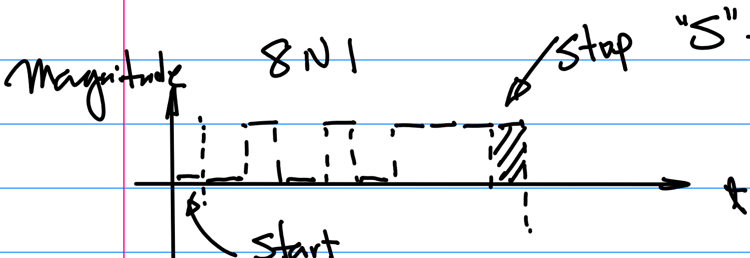
2017F-108-Characterization of BB Signals.pdf

Given A Base Band Signal $x(n)$

Its F.T. $X(m)$

{ UART: Protocol Level
{ RS232:





"5" → 0x57
0101 ; 0111

Fig. 1a.

Consider the Theoretical Background.

Given $x(t)$ and its F.T.

$x(t) \longleftrightarrow X(f)$

Suppose

$x_1(t) \longleftrightarrow X_1(f)$

$x_2(t) \longleftrightarrow X_2(f)$

then,

$a x_1(t) \longleftrightarrow a X_1(f)$

$\left(X(f) \triangleq \frac{1}{T}\int_T x(t) e^{-j2\pi ft} dt \right.$, So for F.T. of $aX(t)$:

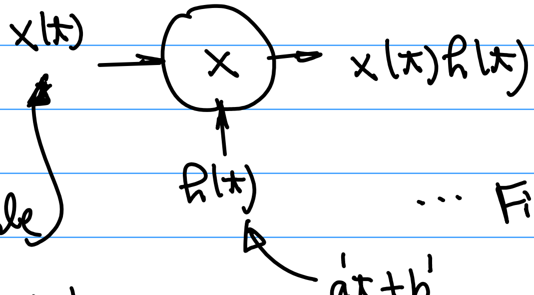
$$\frac{1}{T}\int_T a x(t) e^{-j2\pi f t}\, dt$$

$$= a \underbrace{\left(\frac{1}{T}\int_T x(t) e^{-j2\pi f t}\, dt\right)}$$

$$= a X(f)$$

$$a x_1(t) + b x_2(t) \leftrightarrow a X_1(f) + b X_2(f)$$

Now, Consider 2 Signals,
$x(t)$ as a Base Band Signal
$h(t)$ as a modulating Signal



Example

$at+b$

... Fig. 2.

$$(at+b)(a't+b') = aa't^2 + \cdots$$

Example: Suppose in Fig. 2, we have modulating signal $h(t) = \cos 2\pi f_c t$

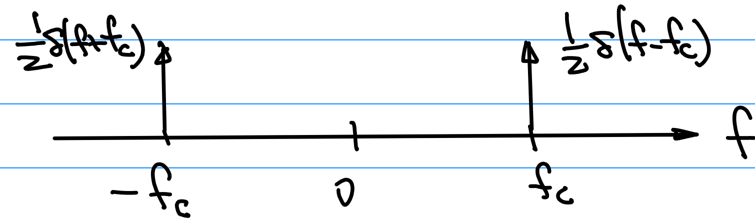
C: "Carrier frequency"

Observation 1: $h(t) \leftrightarrow H(f)$ ... (1)

if $h(t) = \cos 2\pi f_c t$, then

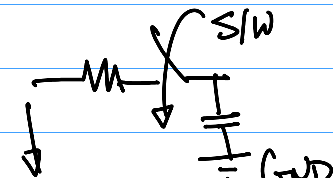
$$H(f) = \frac{1}{2}\left[\delta(f-f_c) + \delta(f+f_c)\right] \quad \ldots (2)$$

for Eq.(2),



Property 1: Impulse Function $\delta(t)$ can be utilized to Sample a given function in the Time Domain.

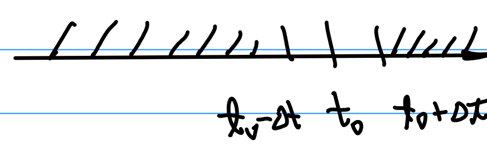
$x(t)$



$x(t)\,\delta(t-t_0) \rightarrow$ the Sampling of the input signal

$$\int_{-\infty}^{+\infty} x(t)\,\delta(t-t_0)\,dt$$

$$= \int_{-\infty}^{t_0-\Delta t} x(t)\,\cancel{\delta(t-t_0)}^{\,0}\,dt + \int_{t_0-\Delta t}^{t_0+\Delta t} x(t)\,\delta(t-t_0)\,dt + \int_{t_0+\Delta t}^{+\infty} x(t)\,\cancel{\delta(t-t_0)}^{\,0}\,dt$$



$$\int_{t_0-\Delta t}^{t_0+\Delta t} x(t)\,\delta(t-t_0)\,dt = x(t_0)$$

Also Note $\int_{-\infty}^{+\infty} \delta(t)\,dt = 1$ for $t = 0$

$\int_{-\infty}^{+\infty} \delta(t-t_0)\,dt = 1$ for $t = t_0$

Nov. 9 (Wed).

Example: Theoretical Discussion on modulation.

Ref.: 2017F-108-Characterization of BB Signals.pdf

Table C.1. Transform theorems.

Name of theorem	Signal	Fourier transform
(1) Superposition	$a_1x_1(t) + a_2x_2(t)$	$a_1X_1(f) + a_2X_2(f)$
(2) Time delay	$x(t - t_0)$	$X(f)\exp(-j2\pi f t_0)$
(3) Scale change	$x(at)$	$\lvert a\rvert^{-1}X(f/a)$
(4) Frequency translation	$x(t)\exp(j2\pi f_0 t)$	$X(f - f_0)$
(5) Modulation	$x(t)\cos 2\pi f_0 t$	$\frac{1}{2}X(f - f_0) + \frac{1}{2}X(f + f_0)$
(6) Differentiation	$\frac{d^n x(t)}{dt^n}$	$(j2\pi f)^n X(f)$

Note: a. In time domain, signal multiplied together, $x(t)e^{j2\pi f_c t}$, e.g., $f(t)h(t)$, see Block Diagram.

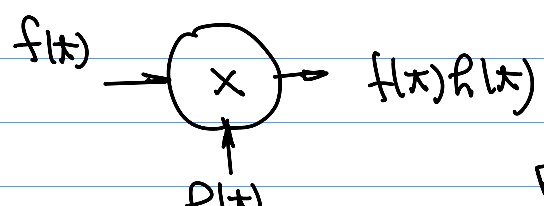


Fig. 1.

Modulator.

b. Frequency Domain Characterization of the modulation.

Suppose we have a modulating function (signal) as $\cos 2\pi f_c t$. And its F.T. (Fourier Transform) is $\frac{1}{2}[\delta(f-f_c)+\delta(f+f_c)]$ ... (1)

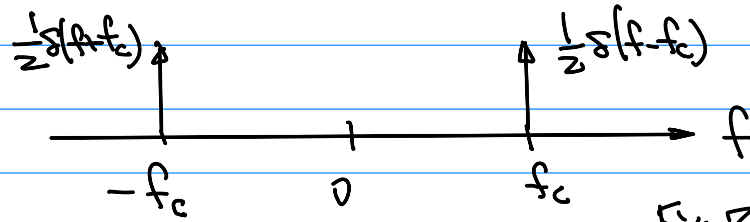
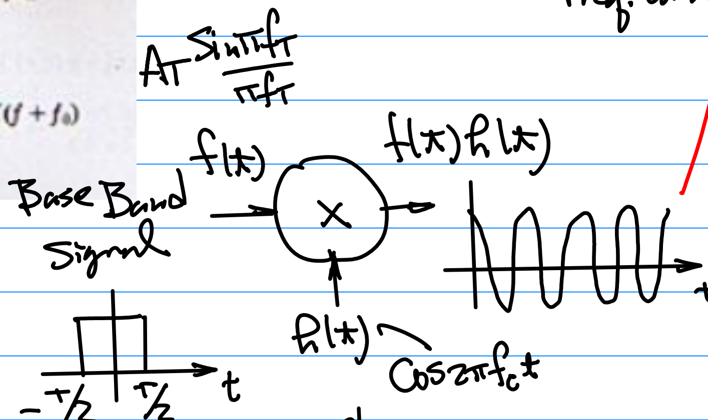


Fig. 2

Now, let's inspect the Frequency Domain Characteristics of Fig. 1.

Freq. Domain



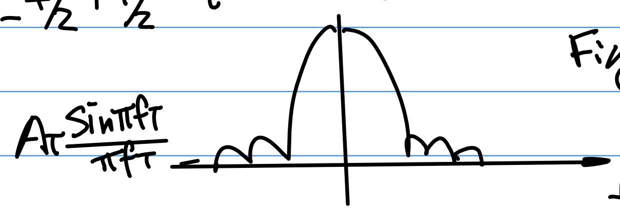


Fig. 3

The modulated signal $f(t)h(t)$ in Freq. Domain is the convolution of $FT[f(t)] * FT[h(t)]$ ... (2)

To illustrate Eqn (2), we have

CmpE245

Nov. 9, 22

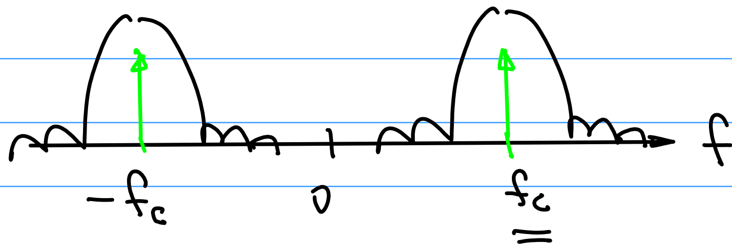


Fig.4.

f_c: Carrier Frequency.

from IEEE 802.11b, $f_c = 2.4\,GHz$

Ref to IEEE 802.11b. B.W. $\simeq 20\,MHz$

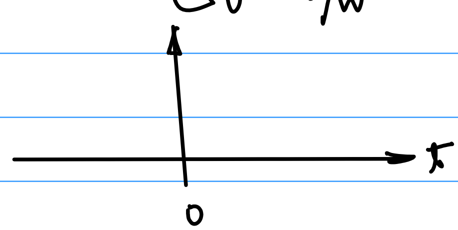
$$\frac{A_T}{2}\frac{\sin\pi(f-f_c)T}{\pi(f-f_c)T} + \frac{A_T}{2}\frac{\sin\pi(f+f_c)T}{\pi(f+f_c)T} \quad \dots (3)$$

Background:

1° Impulse Function
2° Sampling a Signal By Using Impulse

Impulse Function Definition.

$$\delta(t) = \begin{cases} +\infty & t=0 \\ 0 & o/w \end{cases} \quad \dots (4)$$



$$\int_{-\infty}^{+\infty} \delta(t)\,dt = 1 \quad \dots (4\text{-}b)$$

Using Impulse Function to Sample a Signal

$$\int_{-\infty}^{+\infty} \delta(t)\,dt = 1$$

$$\downarrow$$

$$\int_{-\infty}^{+\infty} \delta(t-t_0)\,dt = \int_{-\infty}^{+\infty} \delta(t-t_0)\,d(t-t_0)$$

$$= \int_{-\infty}^{+\infty} \delta(\tau)\,d\tau = 1$$

Now $\downarrow$

$$\int_{-\infty}^{+\infty} \alpha\delta(t)\,dt = \alpha$$

$$\downarrow$$

$$\int_{-\infty}^{+\infty} \alpha\, \delta(t-t_0)\,dt = \alpha \Big|_{at\ t_0}$$

$$\downarrow$$

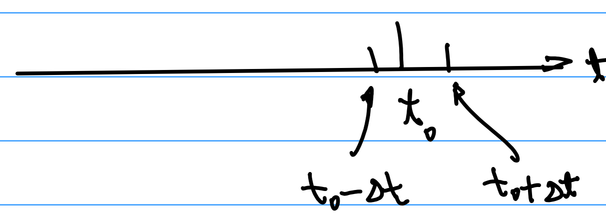
Let $\alpha = f(t)$

$$\int_{-\infty}^{+\infty} f(t)\delta(t)\,dt = f(t)\Big|_{t=0} = f(0)$$

$$\downarrow$$

$$\int_{-\infty}^{+\infty} f(t)\delta(t-t_0)\,dt = f(t)\Big|_{t=t_0} = f(t_0)$$

$$\int_{-\infty}^{t_0-\Delta t} f(t)\delta(t-t_0)\,dt + \int_{t_0-\Delta t}^{t_0+\Delta t} f(t)\delta(t-t_0)\,dt + \int_{t_0+\Delta t}^{+\infty} f(t)\,\delta(t-t_0)\,dt$$

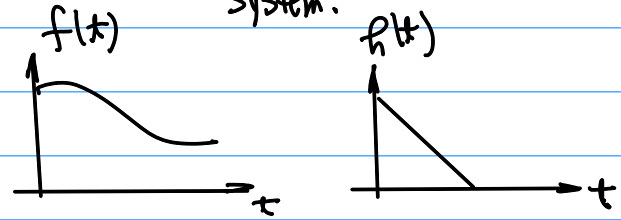


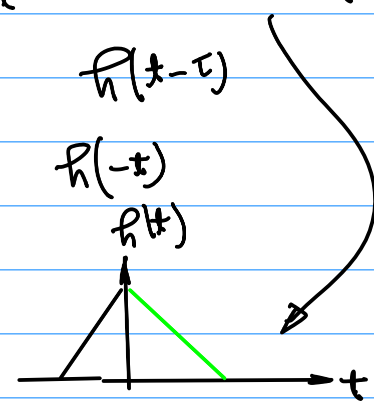
Now, convolution.

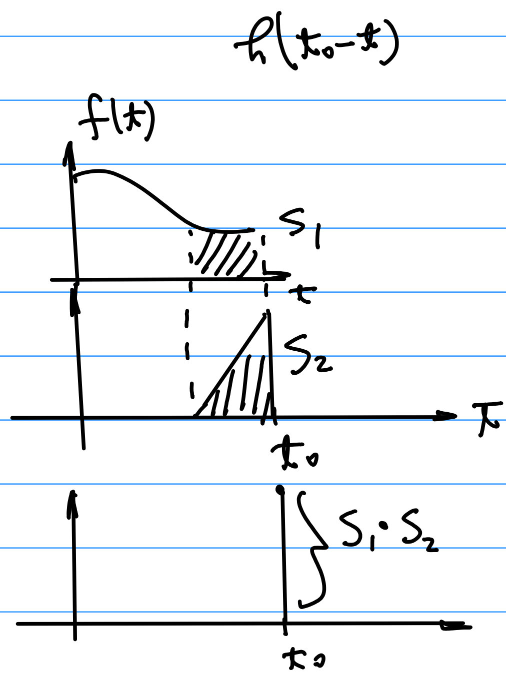
$$f(t) * h(t) = \int_{-\infty}^{+\infty} f(\tau)\, h(t-\tau)\, d\tau \quad \cdots (5)$$

Input $f(t) \longrightarrow \boxed{h(t)} \longrightarrow f(t) * h(t)$

System.







hence, Similarly, in the Freq. Domain,

$$FT[f(t)] * FT[h(t)]$$

$$= A\tau \frac{\sin \pi f \tau}{\pi f \tau} * \left[\frac{1}{2}\delta(f-f_c) + \frac{1}{2}\delta(f+f_c)\right]$$

Response $= \frac{1}{2} A\tau \frac{\sin \pi f \tau}{\pi f \tau} * \delta(f-f_c) + \frac{1}{2} A\tau \frac{\sin \pi f \tau}{\pi f \tau} * \delta(f+f_c)$

$$= \frac{1}{2} A\tau \frac{\sin \pi (f-f_c)\tau}{\pi (f-f_c)\tau} + \frac{1}{2} A\tau \frac{\sin \pi (f+f_c)\tau}{\pi (f+f_c)\tau}$$

Nov. 14 (Monday).

Note: 1° Project To Implement Scrambling/De-scrambling is Due on Nov. 28 (Monday).

Requirements: Scramble the payload only.

2° LoRA + Presentation in Class ON Dec 5th (Monday). To Establish handshaking, & message Back and Forth,

SJSU + CmpE245 + First Name + Last Name + SID

Send Ack message Back to the Transmitter

Consider Pack More Energy in the Lower Frequency Range.

BaseLine Ref: BaseBand Signal

$$g(t) \longleftrightarrow FT[g(t)] = AT\frac{\sin \pi f T}{\pi f T}$$

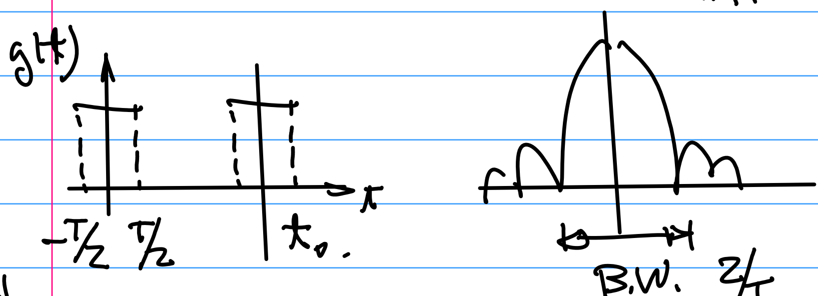


Fig.1

Increase Bit Rate, 2x
Reduce T to T/2 → Find B.W.

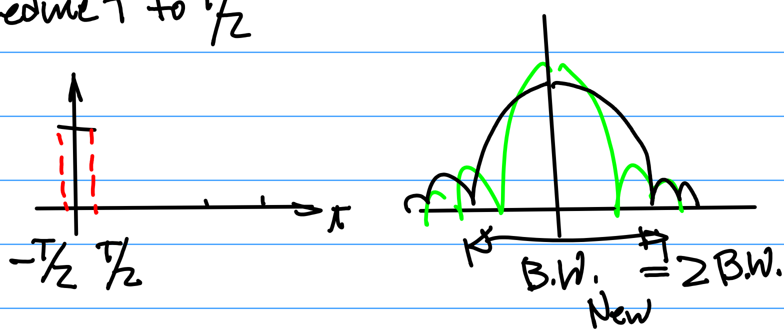


Fig.2

Now, Consider Packing More Energy into the Lower Freq Range.
Since the sharp Edges are approximated By Sine & Cosine Function. e.g. Fourier Transform, the sharp Edge will take more Sine, cosine terms to estimate its value, Hence, more Energy in a higher Freq Range.

Option 1: To Avoid Sharp Edge, By using Sine, Cosine shaped waveforms or triangular waveforms.

Option 2: To Increase the interval of having a stable/constant waveforms.

Homework (0 pts), Bring LORA Board and Prototype System to the Class for quick Show & Tell.

1. Modules Ready;
2. Soldered OR Wire Wrapped;

Option: Software part.

Example:
Implement LORA Module;

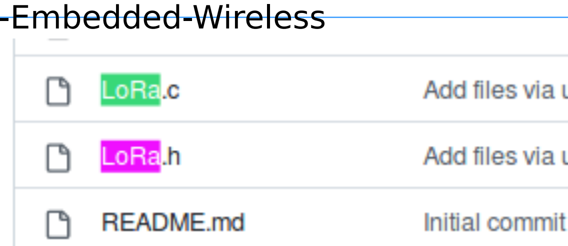
1° Target platform Ready.
LPC1769 → github. SPI I/F Sample code

1.1° SPI. Sample code for FLASH I/F. Datalog. (For CmpE127 Class) → AT45 (8 Mbit Flash) OR Any other SPI Based Flash

1.2° SPI Sample Code for LCD DrawLine Need LCD Hardware

1.3° SPI Sample code for LORA Wireless Communication.

https://github.com/hualili/CMPE245-Embedded-Wireless



CMPE245

Nov.14,22

From LoRa.h.

```
int available();
int read();
int peek();
void flush();

void onReceive(void(*callback)(int));

void receive(int size);
void idle();
void sleep();

void setTxPower(int level);
void setFrequency(long frequency);
void setSpreadingFactor(int sf);
void setSignalBandwidth(long sbw);
void setCodingRate4(int denominator);
void setPreambleLength(long length);
void setSyncWord(int sw);
void crc();
void noCrc();
```

For C.R.

For SPI LCD Ref.

https://github.com/hualili/CMPE240-Adv-Microprocessors/blob/master/2018S-10-LCD-DrawLine.zip

Import this to your MCUXpresso, then go to "src" folder to find S.P.I. module.

For SPI DataLogger Design.

https://github.com/hualili/CMPE127-Microprocessor-Systems/blob/master/SSP_Test(1).zip

Test S.P.I I/F:

"3+1" pins for SPI I/F.

MOSI (Master Out Slave In)

MISO (" IN " Out)

SCK — $f_{SPI} = \dfrac{PCLK}{CPSDVSR(SCR+1)}$

Enable

Config & Init.

From LoRa.c

```
#include "LoRa.h"
#include "ssp.h"
#include "extint.h"
// registers
#define REG_FIFO                    0x00
#define REG_OP_MODE                 0x01
#define REG_FRF_MSB                 0x06
#define REG_FRF_MID                 0x07
#define REG_FRF_LSB                 0x08
#define REG_PA_CONFIG               0x09
#define REG_LNA                     0x0c
#define REG_FIFO_ADDR_PTR           0x0d
#define REG_FIFO_TX_BASE_ADDR       0x0e
#define REG_FIFO_RX_BASE_ADDR       0x0f
#define REG_FIFO_RX_CURRENT_ADDR    0x10
#define REG_IRQ_FLAGS               0x12
#define REG_RX_NB_BYTES             0x13
#define REG_PKT_RSSI_VALUE          0x1a
#define REG_PKT_SNR_VALUE           0x1b
#define REG_MODEM_CONFIG_1          0x1d
#define REG_MODEM_CONFIG_2          0x1e
#define REG_PREAMBLE_MSB            0x20
#define REG_PREAMBLE_LSB            0x21
#define REG_PAYLOAD_LENGTH          0x22
#define REG_RSSI_WIDEBAND           0x2c
```

CMPE244/2021F-107-lpc-cpu-UM10360.pdf

18.6 Register description

The register addresses of the SSP controllers add

Table 370. SSP Register Map

Generic Name	Description	Access
CR0	Control Register 0. Selects the serial clock rate, bus type, and data size.	R/W
CR1	Control Register 1. Selects master/slave and other modes.	R/W
DR	Data Register. Writes fill the transmit FIFO, and reads empty the receive FIFO.	R/W
SR	Status Register	RO

CR0

Table 371: SSPn Control Register 0 (SSP0CR0 - address 0x4008 8000, SSP1CR0 - 0x4003 0000) bit description

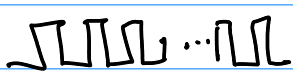
Bit	Symbol	Value	Description	Reset Value
3:0	DSS		Data Size Select. This field controls the number of bits transferred in each frame. Values 0000-0010 are not supported and should not be used.	0000
		0011	4-bit transfer	
		0100	5-bit transfer	
		0101	6-bit transfer	
		0110	7-bit transfer	
		0111	8-bit transfer	
		1000	9-bit transfer	
		1001	10-bit transfer	
		1010	11-bit transfer	
		1011	12-bit transfer	
		1100	13-bit transfer	
		1101	14-bit transfer	
		1110	15-bit transfer	
		1111	16-bit transfer	
5:4	FRF		Frame Format.	00
		00	SPI	
		01	TI	
		10	Microwire	
		11	This combination is not supported and should not be used.	
6	CPOL		Clock Out Polarity. This bit is only used in SPI mode.	0
		0	SSP controller maintains the bus clock low between frames.	

MCU/FSM: Interpretation of the incoming information from the host;

Bank of Special purpose Registers: Control/Config.
Data Tx & Rx

Sample code
DrawLine Sample Code

Osc. or Logic Analyzer.
Check waveforms.



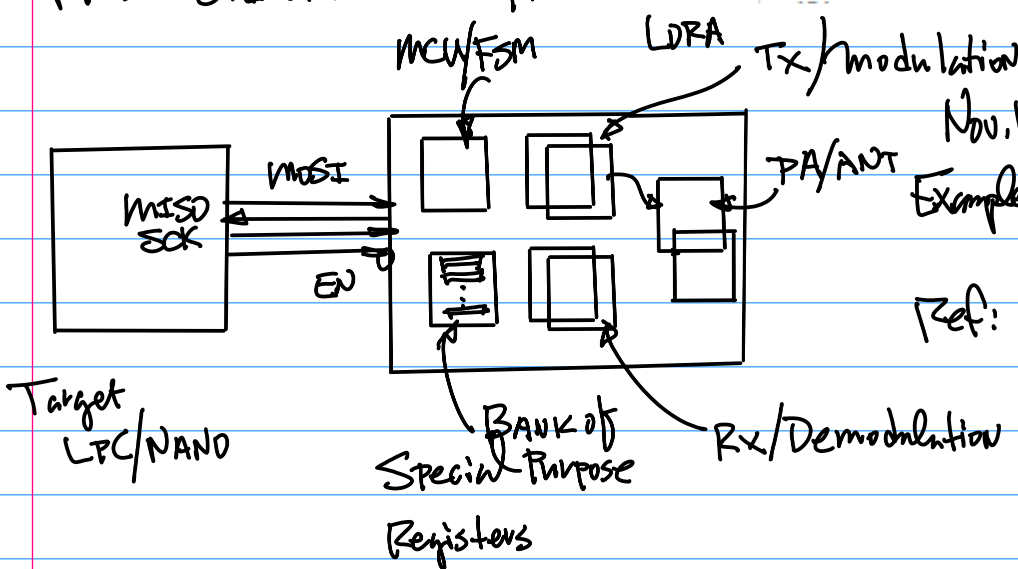
$f_{SPI} = 1KHz$ → LoRa.c Code

```
int LoRabegin(long frequency)
{
  // setup pins
        uint8_t version =0;
        int i=0;
        gpioInit();
  // perform reset

  digitalWrite(LORA_DEFAULT_RESET_PIN, LOW);
  for(i=0;i<100000;i++);
  digitalWrite(LORA_DEFAULT_RESET_PIN, HIGH);
  for(i=0;i<10000000;i++);
```

For NVDA NANO, it is dependent on the Release you have. Jetpack 4.3.? ??
Ubuntu 18.04

Not sure SPI Driver is Ready.



Nov. 16 (Wed).
Example: On Analysis of modulation Systems.
Ref:

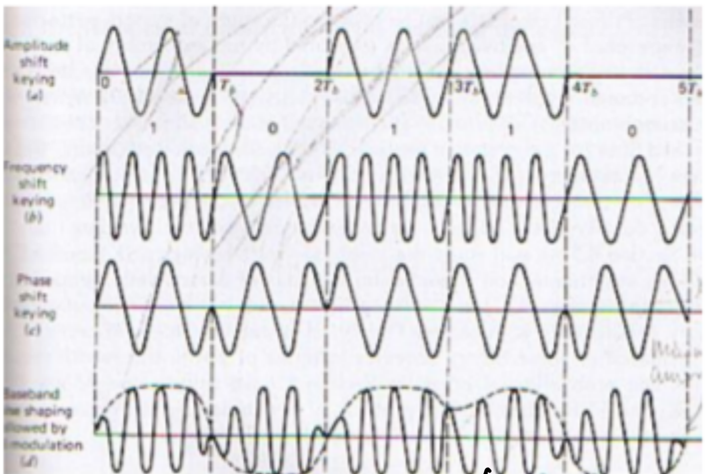
CMPE245

Nov.16,22

Amplitude Shift Keying

ASK, FSK, PSK

Amplitude Frequency Phase



ASK

FSK

PSK Fig.1

Base Band Shaping Analog Modulation (DSB)

See the plot

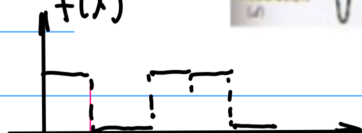


Fig.2

Base Band Signal $f(t) \rightarrow \otimes \rightarrow f(t)h(t)$

$h(t)$ Carrier

$A\cos 2\pi f_c t$

Note: For Different channels.

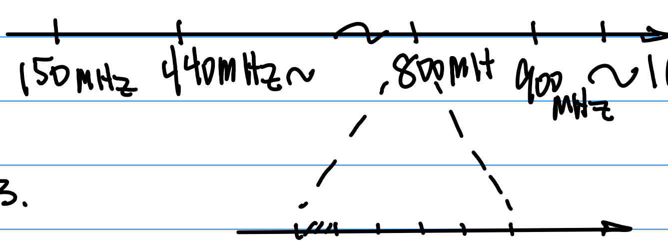


Fig.3.

Note: C.R. (Cognitive Radio) Smarter Communication. By Changing Parameters

to Adapte to the R.F. Environment.

1) Index for Quality of Communication.

S/N

Entropy Based Measurement

2) Change/Adaptation of the Comm.

Power Amplification, PA ↑

Scrambling in Higher order, example 7th → 9th? 11?

Sync.

CRC & Coding for Error Detection & Correction.

PSK: phase Carries Information.

a. Create phase Shift

CmpE245

Nov.16,22

$A\cos(2\pi f_c t + \phi_0) \rightarrow$ Symbol S_0 "0"

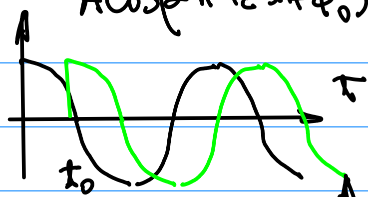


Fig.4

$A\cos(2\pi f_c t + \phi_1)$
↓
Symbol S_1 "1"

SJSU $\xrightarrow{\text{ASCII}}$ "S" 0XF5

1111 0101

$A\cos(2\pi f_c t + \phi_1)$

$A\cos(2\pi f_c t + \phi_0)$

$T_b = \frac{1}{1024}$ sec. $\simeq \frac{1}{1000}$ sec.

B.W. $= 2/T_b \simeq 2$ KHz

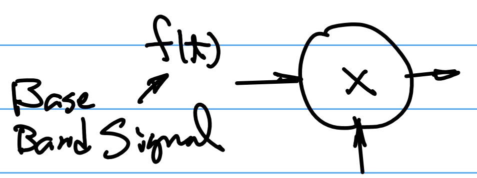
Example: Design modulator to Transmit Base Band Signal @ 433MHz.

Sol First. Design By Providing Block Diagram. (Fig.5)

Note: It is BPSK.

a. Analyze its Power Spectrum for Each of the key Signals,

b. Before & After the modulation



$A\cos 2\pi f_c t$

$A\cos(2\pi f_c t + \phi_i)$

$i = 0$ or 1

"Bi-PSK"
↓
BPSK

Fig.5

Before { Base Band Signal, Carrier Signal

After: modulated (Product of the two in Time Domain)

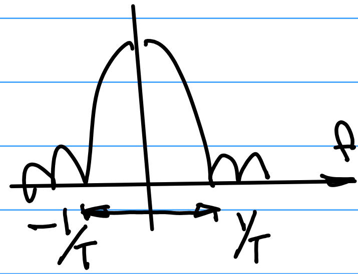
Example: Given a Based Band Signal, find its F.T. (Power Spectrum)

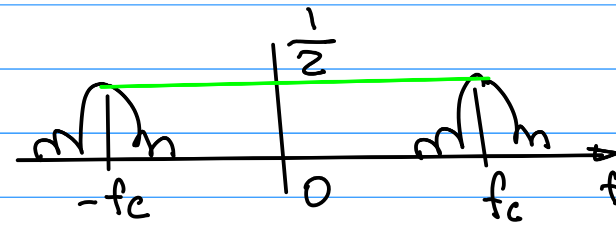
Sol: $g(t) = \begin{cases} 1 & t \in [T/2, -T/2] \\ 0 & \text{o/w}. \end{cases}$ ...(1)

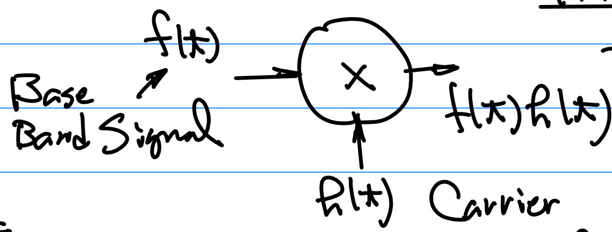
$FT[g(t)] = AT\frac{\sin \pi f T}{\pi f T}\Big|_{A=1}$ ...(2)

Suppose we are transmitting 1Kbps. Find its B.W. = ?

FT: $A\tau \frac{\sin \pi f \tau}{\pi f \tau}$



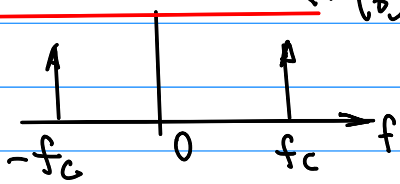




$A\cos 2\pi f_c t$

FT[$\cos 2\pi f_c t$] =

FT: $\frac{1}{2}[\delta(f-f_c)+\delta(f+f_c)]$ ...(3)



Property 1. $f(t)h(t) \longleftrightarrow F(f)*H(f)$ ...(4)

Sol: Convolution

$$F(f)*H(f)=\int_{-\infty}^{+\infty}F(u)H(f-u)\,du \quad \ldots(5)$$

From the Given Condition,

$$\int_{-\infty}^{+\infty}F(u)H(f-u)\,du=\int_{-\infty}^{+\infty}A\tau\frac{\sin\pi u\tau}{\pi u\tau}\left[\frac{1}{2}\delta(f-u-f_c)+\frac{1}{2}\delta(f-u+f_c)\right]du$$

$$=\frac{1}{2}\int_{-\infty}^{+\infty}A\tau\frac{\sin\pi u\tau}{\pi u\tau}\delta(f-u-f_c)\,du+\frac{1}{2}\int_{-\infty}^{+\infty}A\tau\frac{\sin\pi u\tau}{\pi u\tau}\delta(f-u+f_c)\,du$$

when $f-u-f_c=0$

$u=f-f_c$ $\quad \int_{-\infty}^{+\infty}\delta(f-u-f_c)\,du=1$

$$\frac{1}{2}\int_{-\infty}^{+\infty}A\tau\frac{\sin\pi u\tau}{\pi u\tau}\delta(f-u-f_c)\,du.$$

$$=\frac{1}{2}A\tau\frac{\sin\pi(f-f_c)\tau}{\pi(f-f_c)\tau} \quad \ldots(6\text{-a})$$

Similarly,

$$\frac{1}{2}A\tau\frac{\sin\pi(f+f_c)\tau}{\pi(f+f_c)\tau} \quad (6\text{-b})$$

Nov. 21.

Example: Continuation on Modulation & Demodulation.

Based on the Result in Modulation Eqn (6-a) & (6-b), pp. 44, we now Consider Demodulation

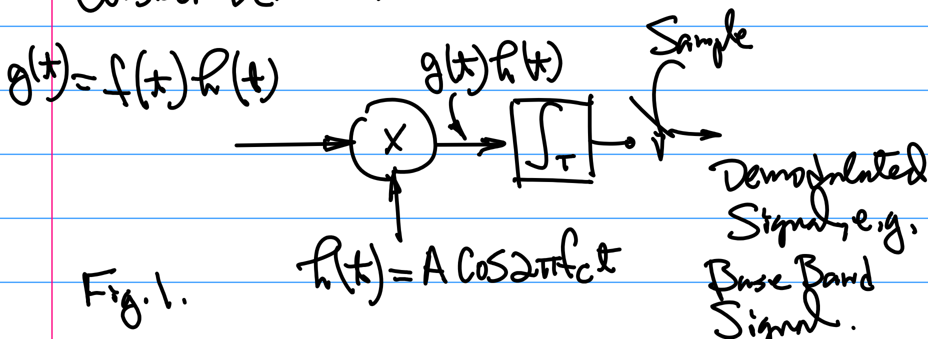


Fig. 1.

In Freq Domain, for $g(t)$, see Equation (6-a), (6-b)

Similarly, In Freq. Domain, for $h(t)$, see Eqn (3)

And, for $g(t)h(t)$, in Freq. Domain, Convolution in Eqn (4).

$$G(f) * \frac{1}{2}\left[\delta(f-f_c)+\delta(f+f_c)\right]$$

$$= \frac{1}{2}\left[G(f-f_c) + G(f+f_c)\right] \quad \cdots (1)$$

where $G(f) = \frac{1}{2}\left[A_T\frac{\sin\pi(f-f_c)T}{\pi(f-f_c)T} + A_T\frac{\sin\pi(f+f_c)T}{\pi(f+f_c)T}\right]$

Then

$$G(f-f_c) = \frac{1}{2}\left[A_T\frac{\sin\pi(f-f_c-f_c)T}{\pi(f-f_c-f_c)T} + A_T\frac{\sin\pi(f-f_c+f_c)T}{\pi(f-f_c+f_c)T}\right]$$

$$= \frac{1}{2}A_T\frac{\sin\pi(f-2f_c)T}{\pi(f-2f_c)T} + \frac{1}{2}A_T\frac{\sin\pi f T}{\pi f T} \quad \cdots (2)$$

Similarly, $G(f+f_c) = \frac{1}{2}\left[A_T\frac{\sin\pi(f+f_c-f_c)T}{\pi(f+f_c-f_c)T} + A_T\frac{\sin\pi(f+f_c+f_c)T}{\pi(f+f_c+f_c)T}\right] \quad \cdots (3)$

Sub Eqn (2) & (3) into (1), so

$$\frac{1}{2}\left[G(f-f_c)+G(f+f_c)\right] =$$

$$\frac{1}{2}\left[\frac{1}{2}AT\frac{\sin\pi(f-2f_c)T}{\pi(f-2f_c)T}+\frac{1}{2}AT\frac{\sin\pi f T}{\pi f T}\right]+$$

$$\frac{1}{2}\,\frac{1}{2}\left[AT\frac{\sin\pi(f+f_c-f_c)T}{\pi(f+f_c-f_c)T}+AT\frac{\sin\pi(f+f_c+f_c)T}{\pi(f+f_c+f_c)T}\right]$$

$$=\frac{1}{4}\left[2AT\frac{\sin\pi f T}{\pi f T}+AT\frac{\sin\pi(f-2f_c)T}{\pi(f-2f_c)T}+AT\frac{\sin\pi(f+2f_c)T}{\pi(f+2f_c)T}\right] \quad \cdots (4)$$

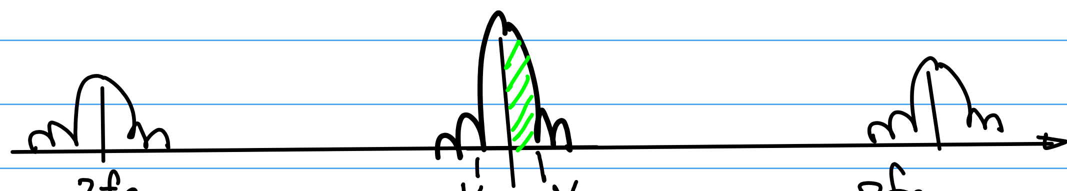


Fig. 2

$\underbrace{\quad}_{\text{B.W.} = 2/T}$

Base Band Signal Recovered from Demodulation By Low Pass Filter.

Consider Increase Bit Rate while Keeping the Same Bandwidth.

First, Binary PSK. BPSK

Signal After modulation:

$$\begin{cases} A\cos(2\pi f_c t+\phi_1) & \text{for "0"} \\ A\cos(2\pi f_c t+\phi_2) & \text{for "1"} \end{cases} \quad \cdots (5)$$

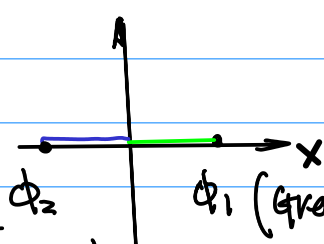


Fig 3.

(Green & Positive x)

(Blue & Positive x)

Quadratic ↓

$$\begin{cases} A\cos(2\pi f_c t+\phi_1) & \text{for } 00 \\ A\cos(2\pi f_c t+\phi_2) & 01 \\ A\cos(2\pi f_c t+\phi_3) & 10 \\ A\cos(2\pi f_c t+\phi_4) & 11 \end{cases}$$

QPSK

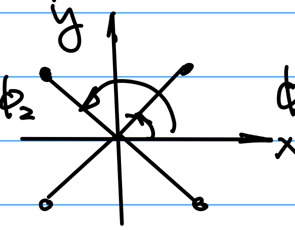


Fig. 4

CmpE245

Nov.21,22

For Demodulation: To Recover the exact Frequency Locally, use modulated Signal with a Delay unit

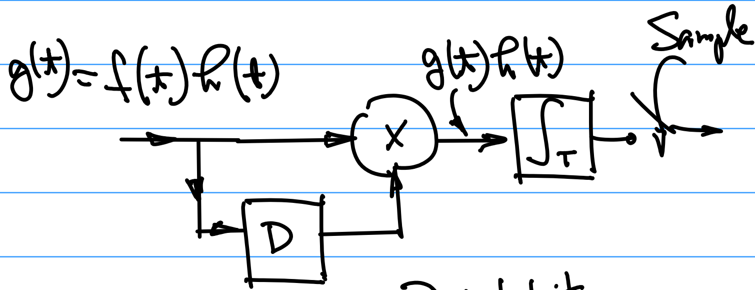


Fig 5.

D: 1 bit

DBPSK

↓

DQPSK, → CCK.

Four modulation formats and data rates are specified for the High Rate PHY. The basic access rate shall be based on 1 Mbit/s DBPSK modulation. The enhanced access rate shall be based on 2 Mbit/s DQPSK. The extended direct sequence specification defines two additional data rates. The High Rate access rates shall be based on the CCK modulation scheme for 5.5 Mbit/s and 11 Mbit/s. An optional PBCC mode is also provided for potentially enhanced performance.

18.4.6.4 Spreading sequence and modulation for 1 and 2 Mbit/s