Electrical Engineering Department,

Fourth Year - Communications & Electronics.

EE484 COMMUNICATION SYSTEMS

Experiment 2: GPS (GNSS Protocol)

PREPARED BY	SECTION	SEAT.NO.
Mahmoud Mohamed Kamal Ismail	7	250
Mahmoud Alaa-Elden Mahmoud	7	249
Mahmoud AbdElHady Mahmoud	7	248
Mahmoud Adel Hassan Mohamed	7	247
Mostafa Khaled Gaber Rahal	7	262

➢ Contents

https://github.com/MahmoudFierro98/EE484_Communication_Systems_Lab

1. Problem 1: Report about GPS

Originally (NAVSTAR **GPS**): Navigation Satellite Timing and Ranging Global Positioning System.
LOS (line-of-sight) is needed with satellites for accurate positioning as obstacles lead to weak **GPS** signals.

1- Discuss the History of **GPS** system.

- The work on **GPS** started in 1960 under the auspices of the US. Air Forces.
- In 1973 the first satellite was launched into space.
- In 1980s system began to be available for civilian as originally it was designed only for military applications.
- In 1995 the system was fully operational.
- Now It is available everywhere in smart phones and computers.

2- What are the applications of **GPS**?

- For military: Target tracking – Search - Rescues – Coordinate traps – Missile guidance.
- For civilian: Positioning (It provides the coordinates of current position) – Navigation in cars – Clock synchronization (Timing).

3- What is modulation type used in **GPS**?

- BPSK (Some satellites use QAM).
- The C/A code is transmitted on the L1 frequency as a 1.023 MHz signal using a Bi-Phase Shift Key (BPSK) modulation technique. The P(Y)-code is transmitted on both the L1 and L2 frequencies as a 10.23 MHz signal using the same BPSK modulation, however the P(Y)-code carrier is in quadrature with the C/A carrier; meaning it is 90° out of phase.

4- Mention the spectrum frequency band used in military and civilian applications.

- L_1: frequency 1.57542 GHz - L_2: frequency 1.2276 GHz - L_3, L_4, L_5 frequencies exist but $L_1 and\ L_2$are most common.
- For civilian applications: Modulates L_1 carrier.
- For military applications: Modulates L_1, L_2 carrier.

5- What is the multiple access used in **GPS**?

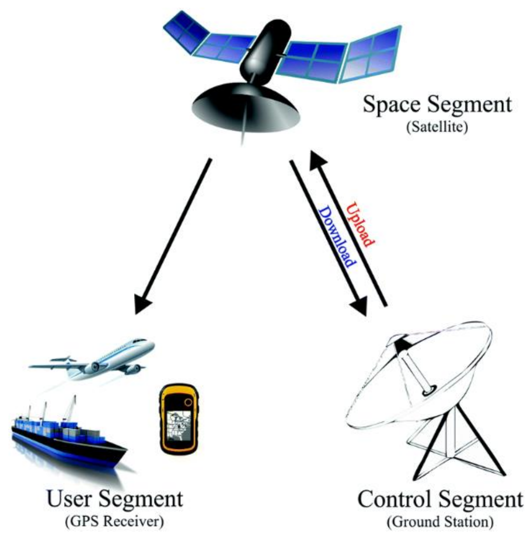
- CDMA (Code Division Multiple Access) each satellite unit is assigned a code differs from (orthogonal) to other satellites.

6- What is the accuracy of **GPS** system?

- Position: Accurate within 4.9m.
- Time: Accurate within 10ns to 100ns due to drift in clock.

7- Describe in detail the **GPS** segments (space segment, control segment, user segment).

- Space Segment (SS): 24 Operational **GPS** satellites at least – Satellites fly in medium earth orbit at 20200 km above earth, with an inclination 55° - Each Satellite circles earth twice a day in 6 equally spaced orbits – At least 4 satellites are needed at any point on earth to determine its position.



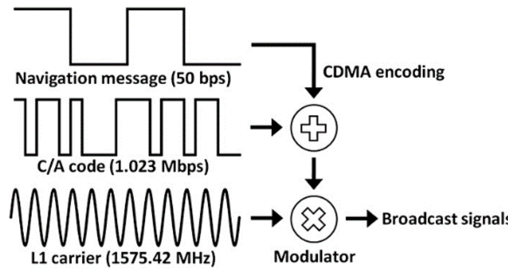
- Control Segment (CS): Consists of a global network of ground facilities that Track **GPS** satellites, analyze their Transmission and send commands data to satellite units.

Monitor Station	Master Control	Ground Antenna
- Tracks **GPS** satellites. - Collect navigation signals and atmospheric data.	- Provides commands, control **GPS**. - Compute precise location of satellite. - Provides maintenance.	- Send commands data upload to satellite units.

- User Segment (Receiver) (US): Consists of antenna, receiver processor, clock and display – Collect signals receives from 4 satellites to convert them into position, velocity and time estimate.

8- Describe the Coarse Acquisition (C/A) code used in **GPS**, how it is constructed(generated), it’s code rate, the spectrum frequency band it is used with, and is it used for military or civilian applications.

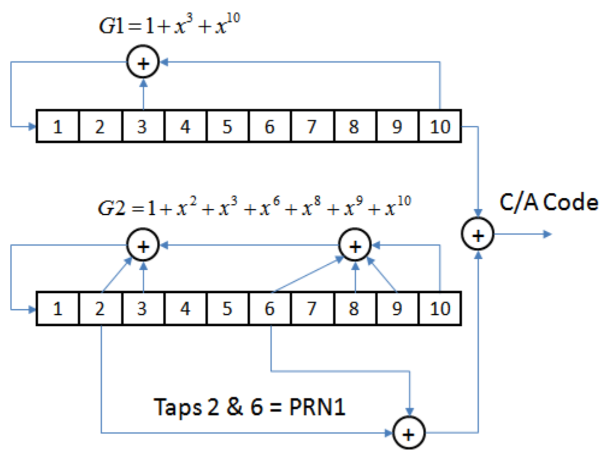
- Modulates L_1 carrier.
- Code rate is 1.023 Mchip/sec and code repeats each 1023 chips.
- Multiply (XORing) 50 bit/sec Navigation message.



- Different code for each satellite CDMA allows identification of satellites even they transmit over L_1 frequency.
- Used for civil standard positioning service (SPS).

GPS C/A Code Generator

PRN ID	G2 Taps	PRN ID	G2 Taps
1	2 & 6	17	1 & 4
2	3 & 7	18	2 & 5
3	4 & 8	19	3 & 6
4	5 & 9	20	4 & 7
5	1 & 9	21	5 & 8
6	2 & 10	22	6 & 9
7	1 & 8	23	1 & 3
8	2 & 9	24	4 & 6
9	3 & 10	25	5 & 7
10	2 & 3	26	6 & 8
11	3 & 4	27	7 & 9
12	5 & 6	28	8 & 10
13	6 & 7	29	1 & 6
14	7 & 8	30	2 & 7
15	8 & 9	31	3 & 8
16	9 & 10	32	4 & 9



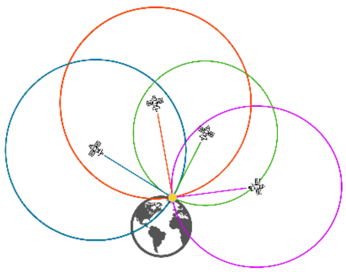
A different C/A code is generated by selecting different taps off of G2, which results in delaying the G2 code relative to G1

9- Describe the p-code used in **GPS**, how it is constructed(generated), it’s code rate, the spectrum frequency band it is used with, and is it used for military or civilian applications.

- Very long code with rate 10.23Mcps (Seven days period 6.19×10^{12}.
- Can be encrypted to get Y-code (Military only) to prevent unauthorized users.
- Modulates L_1, L_2 carrier frequencies.
- Used for military precise positioning system (PPS).

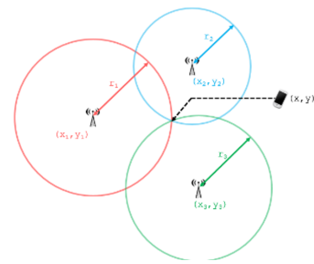
10- Describe in detail how **GPS** calculate positioning and what is the minimum number of satellites needed to calculate positioning.

- Satellite and receiver generate the same code.
- Receiver can correlate received code from satellite with delay with its code.



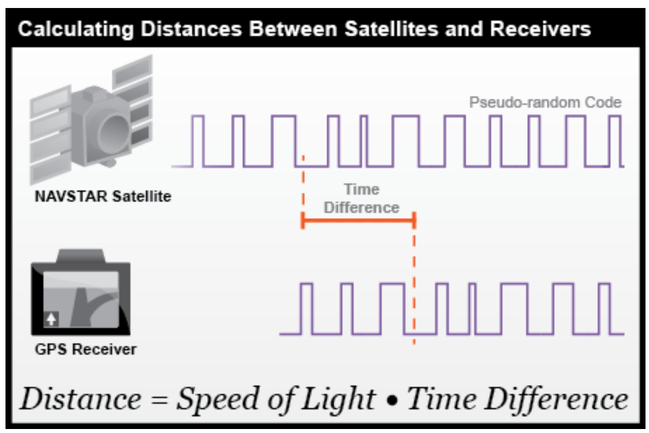
3D

- Triangulation from 3 satellite, you can triangulate position accurately in 2D, but in 3D you need more satellite because 3 circles intersect in 2 points.



2D

- By decoding navigation data, we get orbital data that describes elliptical path of satellite, its position and assist position calculation.



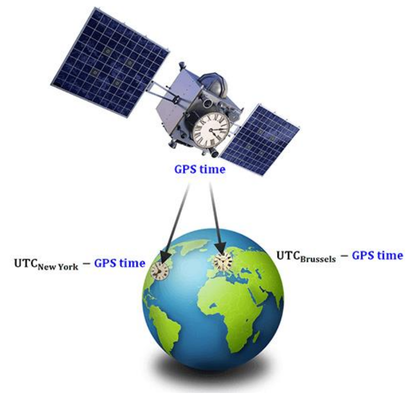
11- What are the sources of error in positioning using **GPS**?

- Atmospheric Interference.
- Calculation and rounding errors.
- Ephemeris (orbital path) data errors.
- Multi-path effects.

12- How **GPS** time is established?

- Each timing lab contributing to UTC measures its own version of UTC for example, UTCBrussels is the Belgian measure of UTC. So how does BIPM compare the performance of all these different clocks? The answer is that it uses the **GPS** receivers or more accurately, GNSS (Global Navigation Satellite System) receivers which besides **GPS**, also track constellations such as GLONASS, Galileo, BeiDou and IRNSS.

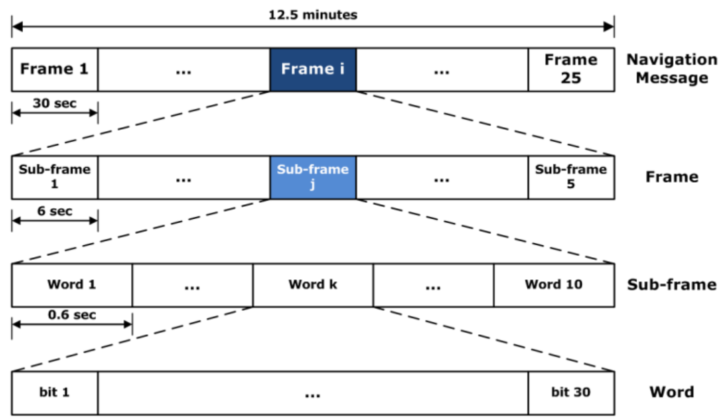
- The precise measurement of time is at the heart of every **GPS** receiver. The distances between satellite and receiver, used to calculate position, are determined by measuring the transit times of the satellite signals to the receiver. An error of 1 nanosecond in the transit time translates into an error of 30cm in the distance. The **GPS** satellite constellation uses its own precise measure of time called **GPS** time with each satellite having its own, on-board set of atomic clocks. Satellites can thus be viewed as very accurate flying clocks.



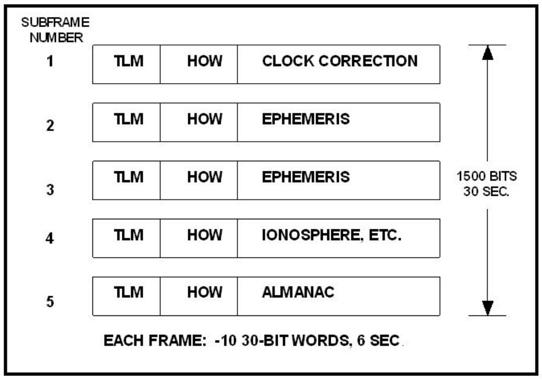
- By tracking a **GPS** satellite, a receiver can record the time differences between its own receiver clock and the satellite clock, e.g. $UTC_{Brussels}$ - **GPS** time. These time differences, along with other information, are collected in a data format called CGGTTS and sent to BIPM. Using CGGTTS and other data, BIPM can compare a clock in Brussels with a clock in New York by subtracting the individual differences with **GPS** time: a technique known as "common view".
- $UTC_{Brussels}$ - $UTC_{New\ York}$ = ($UTC_{Brussels}$ - **GPS** time) - ($UTC_{New\ York}$ - **GPS** time)
- The two **GPS** time terms above cancel each other out leaving the difference between $UTC_{Brussels}$ and $UTC_{New\ York}$.
- Each **GPS** satellite contains multiple atomic clocks that contribute very precise time data to the **GPS** signals. **GPS** receiversdecode these signals, effectively synchronizing each receiver to the atomic clocks. This enables users to determine the time to within 100 billionths of a second, without the cost of owning and operating atomic clocks.

13- Describe NAV Message content and format.

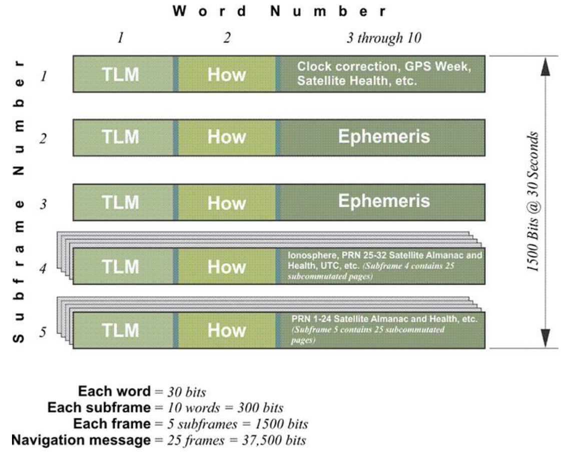
- The current “legacy” Navigation Message (NAV) is modulated on both carriers at 50 bps. The whole message contains 25 pages (or ’frames’) of 30 seconds each, forming the master frame that takes 12,5 minutes to be transmitted. Every frame is subdivided into 5 sub-frames of 6 seconds each; in turn, every sub-frame consists of 10 words, with 30 bits per word (see figure 3). Every sub-frame always starts with the telemetry word (TLM), which is necessary for synchronism. Next, the transference word (HOW) appears. This word provides time information (seconds of the **GPS** week), allowing the receiver to acquire the week-long P(Y)-code segment.



- The content of every sub-frame is as follows:
  - Sub-frame 1: contains information about the parameters to be applied to satellite clock status for its correction. These values are polynomial coefficients that allow converting time on board to **GPS** time. It also has information about satellite health condition.
  - Sub-frames 2 and 3: these sub-frames contain satellite ephemeris.
  - Sub-frame 4: provides ionospheric model parameters (in order to adjust for ionospheric refraction), UTC information (Universal Coordinate Time), part of the almanac, and indications whether the Anti-Spoofing, A/S, is activated or not (which transforms P code into the encrypted Y code).
  - Sub-frame 5: contains data from the almanac and the constellation status. It allows to quickly identify the satellite from which the signal comes. A total of 25 frames are needed to complete the almanac.

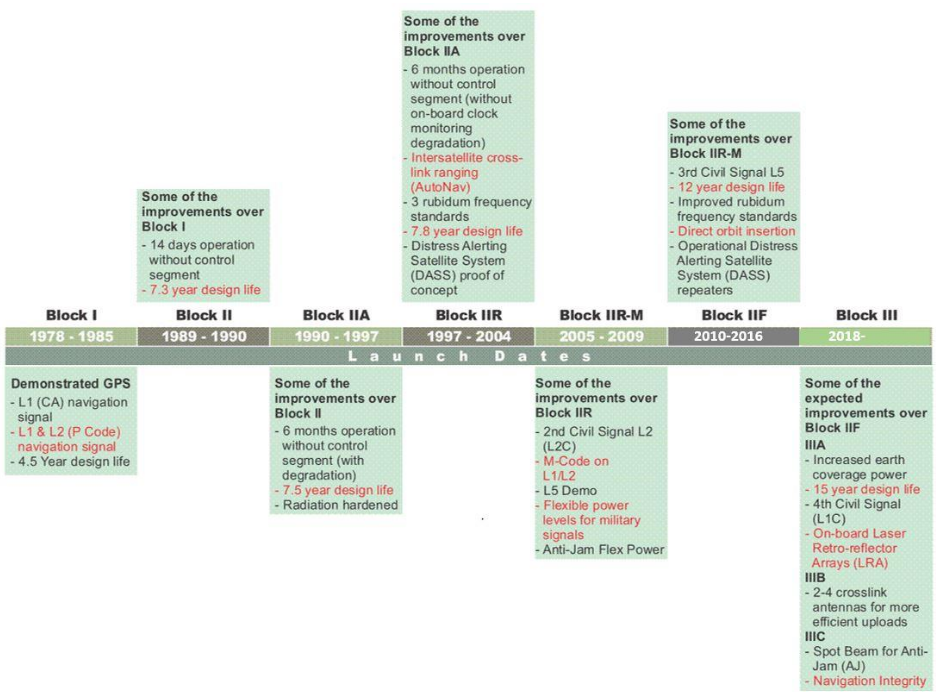


TLM = Telemetry Word HOW = Handover Word (contains Z-count)



- Sub-frames 1, 2 and 3 are transmitted with each frame (i.e., they are repeated every 30 seconds). Sub-frames 4 and 5 contain different pages (25 pages each) of the navigation message (see figure 1). Thence, the transmission of the full navigation message takes 25 × 30 seconds = 12.5 minutes. The content of sub-frames 4 and 5 is common for all satellites. Thence, the almanac data for all in orbit satellites can be obtained from a single tracked satellite.

14- Compare (in brief) between the main features of the **GPS** satellite blocks and what are the satellite blocks used in L5 interface.



- Block IIF used in L5 interface.

2. Problem 2: Questions Related to the Experiment

1. In the output we got in lab we received some **GPS** messages. Explain how $GPGLL messages are used to determine the current position and give an example of that, also show how $GPGSV messages are used to give information about satellite vehicles in view and give an example of that.

- $GPGLL: GLL, Geographic Latitude and Longitude, contains position information, time of position fix and status.

Example:

```
$GPGLL,3150.7849,N,11711.9292,E,110950.000,A,D*52<CR><LF>
$GNGLL,3150.7822,N,11711.9278,E,110332.000,A,D*4B<CR><LF>
```

Field	Description
$	Each NMEA message starts with '$'
--GLL	Message ID
Latitude	Latitude in format 'ddmm.mmmm' (degree and minutes)
N/S	'N'=North 'S'=South
Longitude	Longitude in format 'dddmm.mmmm' (degree and minutes)
E/W	'E'=East 'W'=West
UTC Time	Time in format 'hhmmss.sss'
Data Valid	'V'=Invalid 'A'=Valid
Positioning Mode	'N'=No fix 'A'=Autonomous GNSS fix 'D'=Differential GNSS fix
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of message

- $GPGSV: GSV, GNSS Satellites in View. One GSV sentence can only provide data for at most 4 satellites, so several sentences might be required for the full information. Since GSV includes satellites that are not used as part of the solution, GSV sentence contains more satellites than GGA does.

Example:

```
$GPGSV,4,1,13,22,74,270,17,18,56,162,11,25,53,128,51,14,51,325,49*77<CR><LF>
$GPGSV,4,2,13,42,45,141,39,12,41,067,46,193,37,171,21,31,31,243,16*48<CR><LF>
$GPGSV,4,3,13,24,23,047,44,04,10,304,19,15,09,097,35,21,05,189,14*74<CR><LF>
$GPGSV,4,4,13,32,01,319,*40<CR><LF>
$GLGSV,2,1,07,72,74,021,49,74,66,010,45,75,51,230,,71,39,128,50*6F<CR><LF>
$GLGSV,2,2,07,65,25,329,33,73,16,031,37,76,01,218,*50<CR><LF>
```

Field	Description
$	Each NMEA message starts with '$'
GPGSV	Message ID
Number of Message	Number of messages, total number of GPGSV messages being output (1~3)
Sequence Number	Sequence number of this entry (1~3)
Satellites in View	Total satellites in view
Satellite ID 1	Satellite ID
Elevation 1	Elevation in degree (0~90)
Azimuth 1	Azimuth in degree (0~359)
SNR 1	Signal to Noise Ration in dBHz (0~99), empty if not tracking
Satellite ID 2	Satellite ID
Elevation 2	Elevation in degree (0~90)
Azimuth 2	Azimuth in degree (0~359)
SNR 2	Signal to Noise Ration in dBHz (0~99), empty if not tracking
Satellite ID 3	Satellite ID
Elevation 3	Elevation in degree (0~90)
Azimuth 3	Azimuth in degree (0~359)
SNR 3	Signal to Noise Ration in dBHz (0~99), empty if not tracking
Satellite ID 4	Satellite ID
Elevation 4	Elevation in degree (0~90)
Azimuth 4	Azimuth in degree (0~359)
SNR 4	Signal to Noise Ration in dBHz (0~99), empty if not tracking
*	End character of data field
Checksum	Hexadecimal checksum
<CR><LF>	Each of message

- Reference: https://docs.rs-online.com/3824/0900766b8147dbf6.pdf https://www.bjtek.com.tw/gps-engine-board/gps-engine-board-rb1612-ub7x.html

2. Is the output we got in lab accurate or not, and why?
   - No and data invalid.
   - Because Module should be outdoor not indoor (LOS).

3. How can we change the rate of updates delivered by the UART?
   - By Baud rate.

Standard Baud	LSPCLK CLOK (37.5MHz)		
	BRR	*Actual Baud*	*Error Ratio*
2400	1952（7A0H）	2400	0
4800	976（3D0H）	4798	-0.04
9600	487（1E7H）	9606	-0.06
19200	243（00F3H）	19211	0.06
38400	121（0079H）	38422	0.06

4. How can we change the initial parameters of the UART transmission/reception (Baud rate…………)?
   - From initialization functions we can change parameters of the UART (Baud rate – parity – number of data bits – number of stop bits …)
   - To change baud rate.

```
UART1_Init(9600);
```

   - To change all parameters

```
UART3_Init_Advanced(9600, _UART_8_BIT_DATA, _UART_NOPARITY, _UART_ONE_STOPBIT, &_GPIO_MODULE_USART3_PD89);
```

3. Problem 3: Mini-Simulations

3.1. Code

Experiment_2_GPS.m

```
%%
% Alexandria University
% Faculty of Engineering
% Electrical and Electronic Engineering Department
%
% Course: Communication Systems Lab.
%
% Experiment 2: GPS (GNSS Protocol).

%%
clear;
close all;
clc;

%% Generate G1
n_bits  = 10;
chips   = 2^(n_bits)-1;
x_axis  = 0:1:(chips-1);
bit_seq = ones(1,n_bits);
G1      = zeros(1,chips);
for i = 1:chips
    G1(1,i)       = bit_seq(1,10);
    new_value     = bitxor(bit_seq(1,10),bit_seq(1,3));
    bit_seq       = circshift(bit_seq',1)';
    bit_seq(1,1) = new_value;
end

%% A. For phase tap (3,8)
bit_seq = ones(1,n_bits);
G2_38pt = zeros(1,chips);
C_A_1   = zeros(1,chips);
% Generate G2 phase tap (3,8)
for i = 1:chips
    G2_38pt(1,i) = bitxor(bit_seq(1,8),bit_seq(1,3));
    new_value    = bitxor(bit_seq(1,10),bitxor(bit_seq(1,9),bitxor(bit_seq(1,8),bitxor(bit_seq(1,6),bitxor(bit_seq(1,3),bit_seq(1,2))))));
    bit_seq      = circshift(bit_seq',1)';
    bit_seq(1,1) = new_value;
end
% Generate C/A Code
for i = 1:chips
    C_A_1(1,i) = bitxor(G1(1,i),G2_38pt(1,i));
end
% Plot
C_A_1 = 2*C_A_1-1;
for i = 0:1:(chips-1)
    C_A_1_sh      = circshift(C_A_1',[i,0]);
    output1(i+1) = C_A_1 * C_A_1_sh;
end
figure('Name','Autocorrelation For phase tap (3,8)');
stem(x_axis,output1);
title('Autocorrelation For phase tap (3,8)');

%% B. Change the phase taps to (2, 6)
bit_seq = ones(1,n_bits);
G2_26pt = zeros(1,chips);
C_A_2   = zeros(1,chips);
% Generate G2 phase tap (3,8)
for i = 1:chips
    G2_26pt(1,i) = bitxor(bit_seq(1,6),bit_seq(1,2));
    new_value    = bitxor(bit_seq(1,10),bitxor(bit_seq(1,9),bitxor(bit_seq(1,8),bitxor(bit_seq(1,6),bitxor(bit_seq(1,3),bit_seq(1,2))))));
    bit_seq      = circshift(bit_seq',1)';
    bit_seq(1,1) = new_value;
end
% Generate C/A Code
for i = 1:chips
    C_A_2(1,i) = bitxor(G1(1,i),G2_26pt(1,i));
end
% Plot
C_A_2 = 2*C_A_2-1;
for i = 0:1:(chips-1)
    C_A_2_sh   = circshift(C_A_2',[i,0]);
    output2(i+1) = C_A_2 * C_A_2_sh;
end
figure('Name','Autocorrelation For phase tap (2,6)');
stem(x_axis,output2);
title('Autocorrelation For phase tap (2,6)');

%% C. Cross-correlation
for i = 0:1:(chips-1)
    C_A_1_sh   = circshift(C_A_1',[i,0]);
    output3(i+1) = C_A_2 * C_A_1_sh;
end
figure('Name','Cross-correlation');
stem(x_axis,output3);
title('Cross-correlation');
```

3.2. Plots

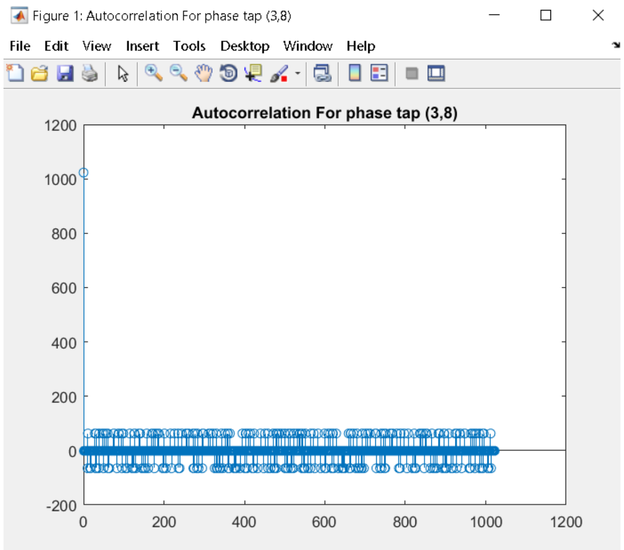


Figure 1: Autocorrelation For phase tap (3,8)
File Edit View Insert Tools Desktop Window Help
Autocorrelation For phase tap (3,8)
1200
1000
800
600
400
200
0
-200
0
200
400
600
800
1000
1200

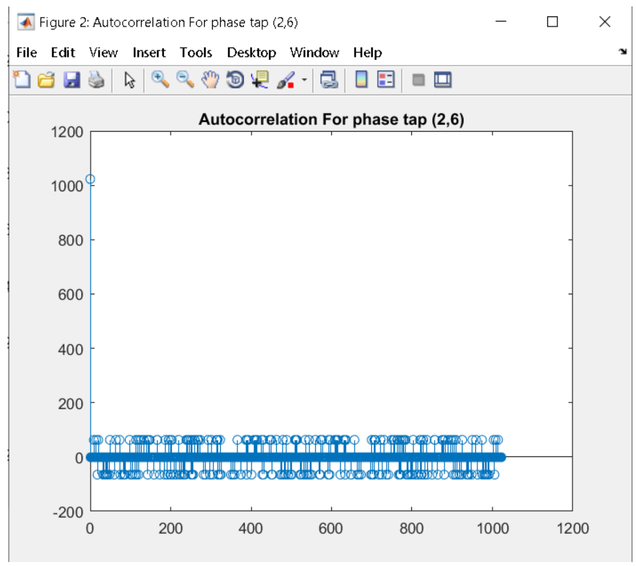


Figure 2: Autocorrelation For phase tap (2,6)
File Edit View Insert Tools Desktop Window Help
Autocorrelation For phase tap (2,6)
1200
1000
800
600
400
200
0
-200
0
200
400
600
800
1000
1200

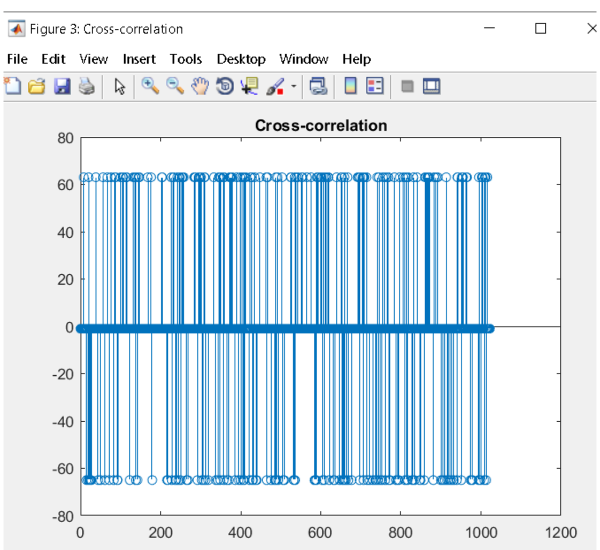


Figure 3: Cross-correlation
File Edit View Insert Tools Desktop Window Help
Cross-correlation
80
60
40
20
0
-20
-40
-60
-80
0
200
400
600
800
1000
1200