Alexandria University
Faculty of Engineering
Electrical and Electronics Engineering
Department

جامعة الإسكندرية
كلية الهندسة
قسم الهندسة الكهربية
الفصل الدراسى الأول, 2022/2021

Lab2: Communication systems.

(Section-6)

GPS module

ID	Name
172	محمد علاء مصطفى خليل هيكل
185	محمود صلاح احمد محمد زهران
189	محمود فوزي طه العربي عبد الحليم

Problem 1: Report about GPS

- **GPS history**

  Originally named **Navstar** GPS.

  Started in 1960 under the US air force. Then in 1973 the first satellite was launched into space. And in 1980 civilians were allowed to use GPS. Finally in 1995 the system was fully operational.

- **GPS applications.**

Civilian application	Military application
Navigation in cars	Target tracking
Clock synchronization	Projectile guidance

- **GPS modulation.**

  BPSK but some systems are using **QAM.**

- **Spectrum frequency band.**

  L1 1.57 GHz

  L2 1.2276 GHZ

  L3, L4 and L5 frequencies exist but L1 and L2 are the most common.

- **GPS multi access.**

  CDMA (code division multiple access): each satellite has a code which is used to send the data.

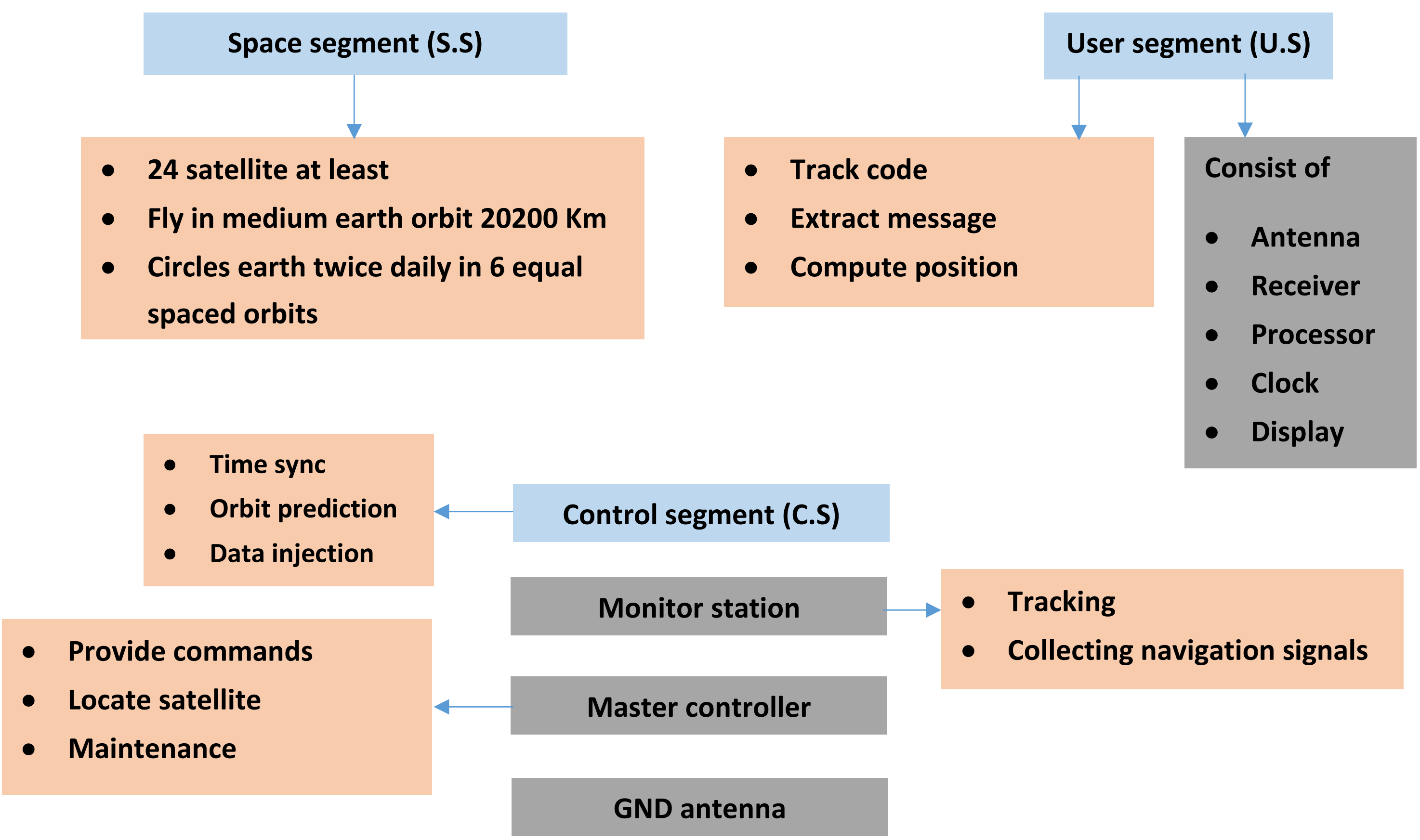
  DSSS (direct sequence spread spectrum): primarily used to reduce overall signal interference.

- **GPS accuracy.**

  Time: accurate from 10ns to 100ns due to drift in clock

  Position: accurate within 4.9m

- **GPS segments.**



- **the Coarse Acquisition (C/A) code**

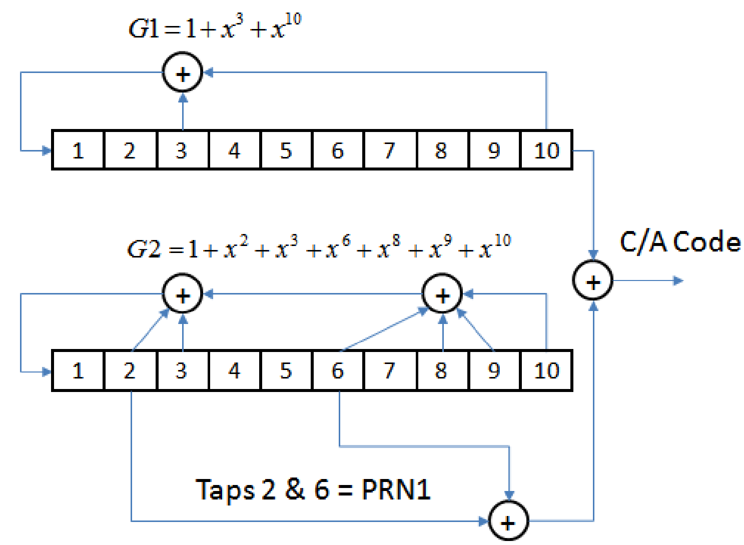
  Code rate: 1.023 MChip/Sec

  Spectrum frequency band: L1

  Used for civilian applications

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

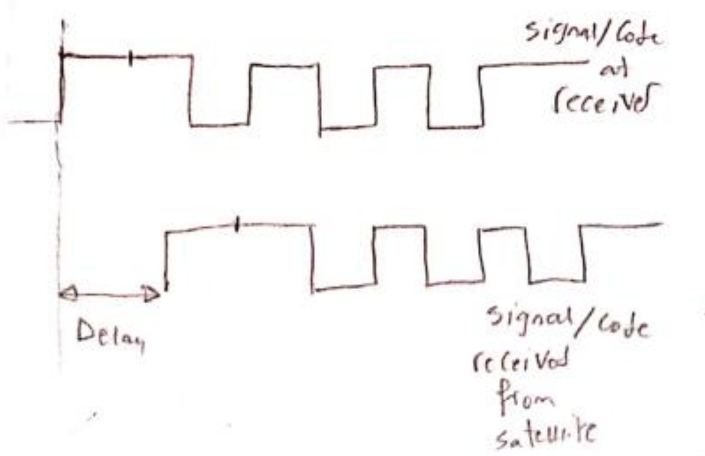
- **the p-code**

  Code rate: 10.023 MChip/Sec

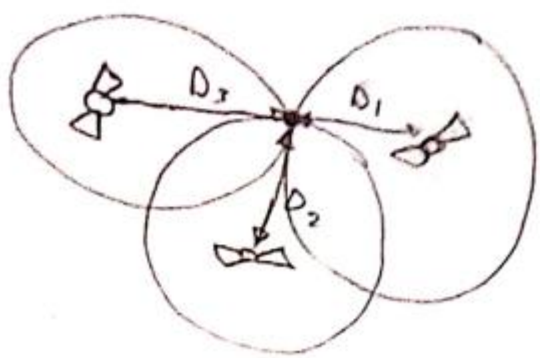
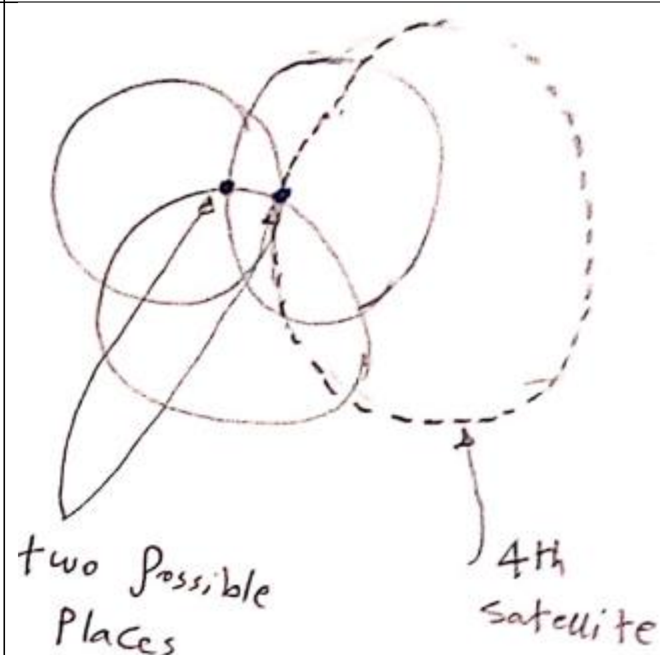
  Spectrum frequency band: L1 and L2

  Used for military applications

- **GPS positioning.**
  1) Satellite and receiver generate the same code
  2) We will have the same code for the receiver and from the received code from the satellite but with delay **td**



  3) Distance from satellite **Di** = **td** * **C** (3x10^8)

2D	**3D**
Needs 3 satellites	Needs 4 satellites
	

- **The sources of error in positioning using GPS.**

  Atmospheric Interference.

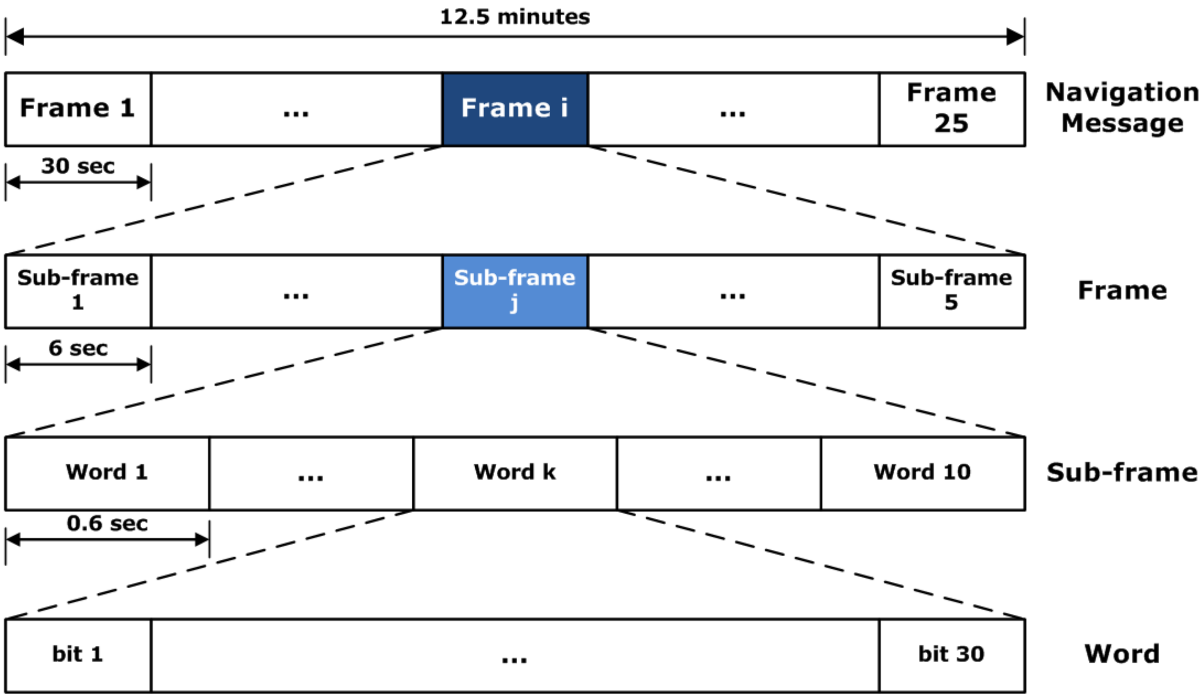
  Calculation and rounding errors.

  Data errors.

- **GPS time.**

  The GPS system is based on atomic clock technology. Each GPS satellite has multiple atomic clocks, synchronized to a ground-based master clock. The GPS clock provides everyone on Earth with access to atomic time standards without needing a local atomic clock.

- **NAV Message content and format.**



Problem 2: the Experiment

1) GPGLL (geographic position latitude/longitude)

```
$GPGLL,5107.0013414,N,11402.3279144,W,205412.00,A,A*73
```

Latitude: 5107.0013414

Latitude direction: N

Longitude: 11402.3279144

Longitude direction: W

UTC time (hours/minutes/seconds/decimal seconds): 205412.00

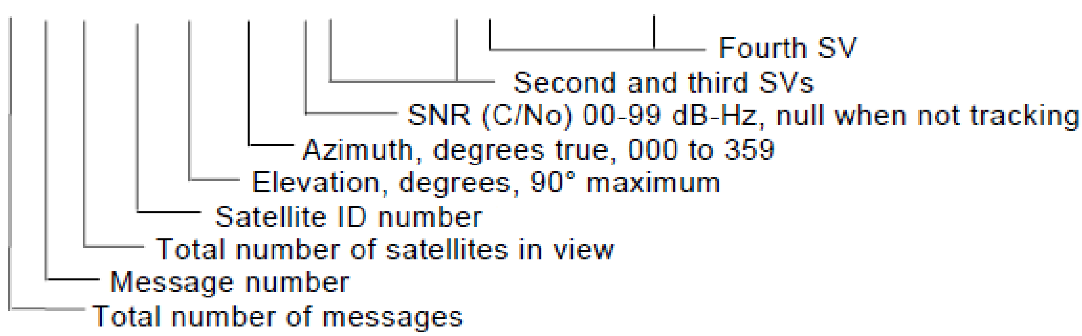
Data status: A (valid)

Mode indicator: A

*xx (check sum): *73

GPGSV (geographic position satellite/view)

$--GSV, x, x, xx, xx, xx, xxx, xx ...., xx, xx, xxx, xx*hh<CR><LF>



2) No, because we are working indoors.

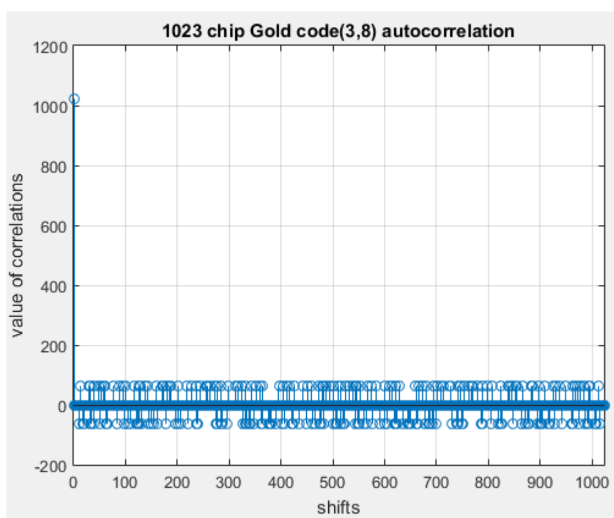
3) To edit the UART

```
UART1_Init(9600);

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

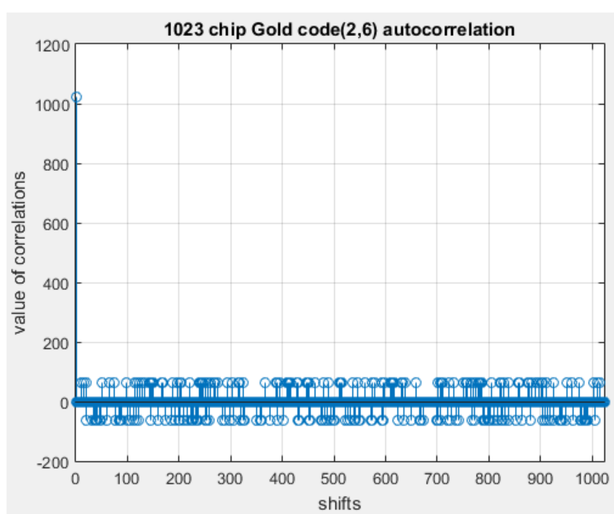
4)

```
UART1_Init(9600);
```

```
%%%%%%%%%%%%%%%%%%%%%%%C/A code with phase tap(3,8)%%%%%%%%%%%%%%%%%%%%%%%
%Register1
SR1=[1 1 1 1 1 1 1 1 1 1];%initial fill of first shift register
SR1_code=zeros(1,1023);% just initialization of the o/p code vector of SR1
and the values have to be changed in the following for loop
for i=1:1023
    SR1_code(1,i)= SR1(1,10);                    % taking the output
    new_value    = bitxor(SR1(1,10),SR1(1,3));   % creating new value to be
added
    SR1          = circshift(SR1',1)';           % shifting
    SR1(1,1)     = new_value;                    % adding new value from the
left
end

%Register2
SR2=[1 1 1 1 1 1 1 1 1 1];%initial fill of second shift register
SR2_code=zeros(1,1023);   % just initialization of the o/p code vector of SR2
and the values have to be changed in the following for loop
for i=1:1023
    SR2_code(1,i) = bitxor(SR2(1,3),SR2(1,8));
    new_value    =
bitxor(SR2(1,10),bitxor(SR2(1,9),bitxor(SR2(1,8),bitxor(SR2(1,6),bitxor(SR2(1
,3),SR2(1,2))))));
    SR2          = circshift(SR2',1)';
    SR2(1,1)     = new_value;
end

CA_code_1=xor(SR1_code,SR2_code);    % xor between G1 and G2 to get C/A
code_1

%autocorrelation calculation
CA_code_1=CA_code_1';   % transpose the code
CA_code_1=2*CA_code_1-1;%change 1/0 to 1/-1

for shift=0:1022
shifted_code= circshift(CA_code_1,shift);             %shifted version of C/A
code 1
autocorrelation_1(shift+1) = CA_code_1'*shifted_code;
end

figure
stem(autocorrelation_1)
grid on
xlabel('shifts');
xlim([0,1023]);
ylabel('value of correlations');
title('1023 chip Gold code(3,8)
autocorrelation')
```



```
%%%%%%%%%%%%%C/A code with phase tap (2,6)%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Register1
SR1=[1 1 1 1 1 1 1 1 1 1];%initial fill of first shift register
SR1_code=zeros(1,1023); % just initialization of the o/p code vector of SR1
and the values have to be changed in the following for loop
for i=1:1023
    SR1_code(1,i)= SR1(1,10);                    % taking the output
    new_value     = bitxor(SR1(1,10),SR1(1,3));  % creating new value to be
added
    SR1           = circshift(SR1',1)';          % shifting
    SR1(1,1)      = new_value;                   % adding new value from the
left

end

%Register2
SR2=[1 1 1 1 1 1 1 1 1 1];%initial fill of second shift register
SR2_code=zeros(1,1023);% just initialization of the o/p code vector of SR2
and the values have to be changed in the following for loop
for i=1:1023
    SR2_code(1,i) = bitxor(SR2(1,2),SR2(1,6));
    new_value     =
bitxor(SR2(1,10),bitxor(SR2(1,9),bitxor(SR2(1,8),bitxor(SR2(1,6),bitxor(SR2(1
,3),SR2(1,2))))));
    SR2           = circshift(SR2',1)';
    SR2(1,1)      = new_value;
end

CA_code_2=xor(SR1_code,SR2_code);              % xor G1 and G2 to get C/A
code_2

% autocorrelation calculation

CA_code_2=CA_code_2';   % transpose the code
CA_code_2=2*CA_code_2-1;%change 1/0 to 1/-1

for shift=0:1022
shifted_code= circshift(CA_code_2,shift);            %shifted version of C/A
code 2
autocorrelation_2(shift+1) =
CA_code_2'*shifted_code;
end

figure
stem(autocorrelation_2)
grid on
xlabel('shifts');
xlim([0,1023]);
ylabel('value of correlations');
title('1023 chip Gold code(2,6)
autocorrelation')
```



```
%%%%%%%%%%%%Cross Correlation between first and second C/A codes %%%%%%%%%%
for i = 0:1:1022
    CA_code_1_sh    = circshift(CA_code_1,[i,0]);
    cross_correlation(i+1) = CA_code_2' * CA_code_1_sh;
end

figure
stem(cross_correlation)
grid on
xlabel('shifts');
xlim([0,1022]);
ylabel('value of correlations');
title('Gold code(3,8)and Gold code(2,6) crosscorrelation')
```