AE410 : NAVIGATION & GUIDANCE

ASSIGNMENT - 01

FUNDAMENTAL OF GPS

Submitted by : Vishal
Roll no. : 200010085
Prog. : B.Tech (Aerospace)

Department of Aerospace Engineering
Indian Institute of Technology Bombay
Oct 30, 2023

1. **Write a MATLAB/Python/C/C++ program to compute circular autocorrelation of PRN 8 with a delayed PRN code by 200 chips and plot the results.**

MATLAB CODE :

```
% Q1
% Generate a PRN 8 code
prn8 = generateCAcode(8);
% Delay the PRN code by 200 chips
delayed_prn8 = circshift(prn8, 200);
circular_autocorrelation = xcorr(prn8, delayed_prn8);
figure;
plot(circular_autocorrelation);
xlabel('Lag');
ylabel('Correlation');
title('Circular Autocorrelation of PRN 8 with a Delayed PRN Code by 200 Chips');
```

In this problem, we have been asked to compute the circular autocorrelation of PRN 8 with a delayed PRN 8 code by 200 chips and plot the result. So, first of all I generated the PRN 8 code by using MATLAB inbuilt function `generateCAcode(8)`,also we have to generate a delayed PRN 8 code by 200 chips, So I used `circshift(prn8, 200)` inbuilt function to delayed the PRN code generated earlier as prn8 by 200 chips. Then I used `xcorr()` function to find the correlation between PRN 8 with delayed PRN 8 by 200 chips and plotted them using `plot()` function labelling x and y axis accordingly.

The plot is shown below, and from this plot, maximum is occuring at x = 823 (i.e. 1023-200)

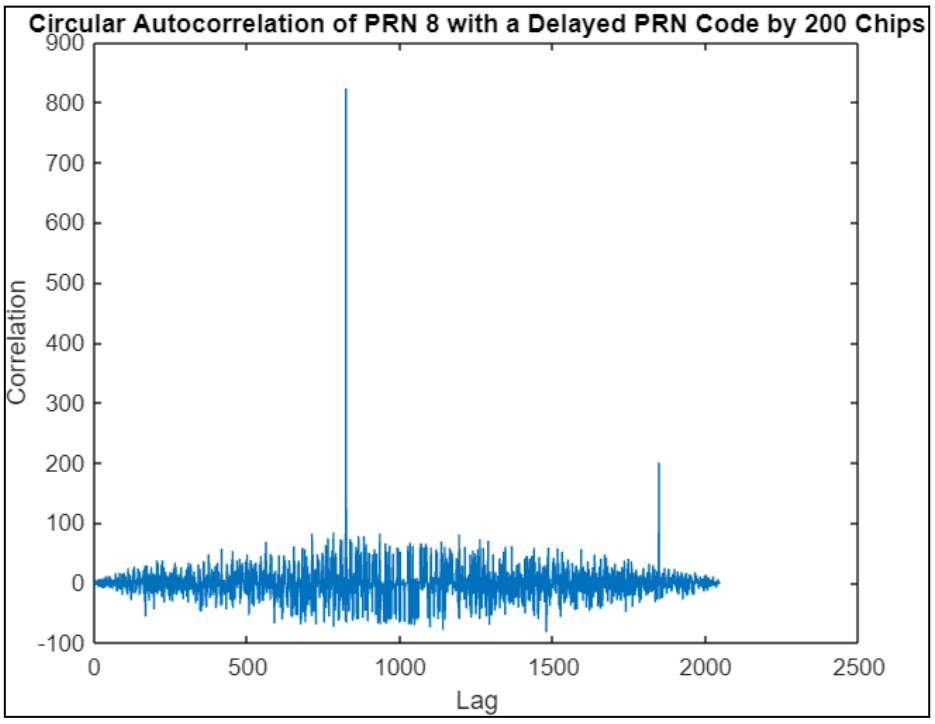

Fig 1

2. **Write a MATLAB/ Python/C/C++ program to compute circular cross correlation of PRN 8 with a delayed copy of a PRN 16 by 900 chips and plot the results.**

MATLAB CODE :

```
%Q2
% PRN 16 code generation bn
prn16 = generateCAcode(16);
delayed_prn16 = circshift(prn16, 900);
circular_cross_corr = xcorr(prn8, delayed_prn16);
figure;
plot(circular_cross_corr);
xlabel('Lag');
ylabel('cross Correlation');
title('Circular cross correlation of PRN 8 with a Delayed PRN 16 Code by 900
Chips');
```

In this part, we have to compute the cross circulation of PRN 8 with a delayed copy of PRN 16 by chips and plot the results. So, earlier I have generated PRN 8 so now, I generated PRN 16 using the same `generateCAcode(16)` function and after that I generated a delayed PRN 16 by using `circshift(prn16, 900).` After that I computed the circular cross correlation of PRN 8 with delayed PRN 16 by using the `xcorr()` function, after that I plotted the circular cross correlation using `plot()` function and then labelled the plot accordingly.

The plot is shown below :

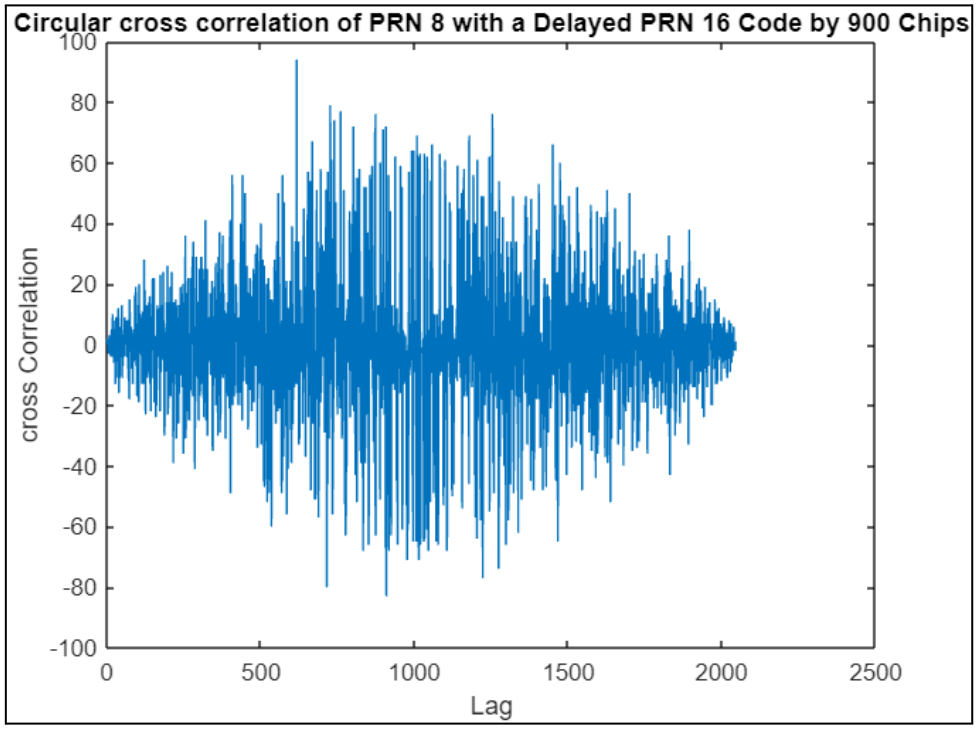

Fig 2

3. **Write a MATLAB/Python/C/C++ program to compute autocorrelation of PRN 8 with a delayed PRN code by -1, 0, and 1 chip, respectively and plot the correlation with the delay chip.**

MATLAB CODE :

```
%Q3
% Generate a PRN 8 code
prn8 = generateCAcode(8);

% Delay the PRN code by -1, 0, and 1 chip
delayed_prn8_neg_one = circshift(prn8, -1);
delayed_prn8_zero = prn8;
delayed_prn8_pos_one = circshift(prn8, 1);

% Compute the autocorrelation
autocorrelation_neg_one = xcorr(prn8, delayed_prn8_neg_one);
autocorrelation_zero = xcorr(prn8, delayed_prn8_zero);
autocorrelation_pos_one = xcorr(prn8, delayed_prn8_pos_one);

% Plot the autocorrelation
figure;
subplot(1,3,1);
plot(autocorrelation_neg_one, 'b--');
xlabel('Lag');
ylabel('Correlation');
title('Delayed by -1 chip');

subplot(1,3,2);
plot(autocorrelation_zero, 'g-');
xlabel('Lag');
ylabel('Correlation');
title('Delayed by 0 chips');

subplot(1,3,3);
plot(autocorrelation_pos_one, 'r:');
xlabel('Lag');
ylabel('Correlation');
title('Delayed by 1 chip');

sgtitle('Autocorrelation of PRN 8 with a Delayed PRN Code by -1, 0, and 1
Chip, Respectively');
```

In this question we have to compute the autocorrelation of PRN 8 with a delayed PRN code by -1, 0, and 1 chip respectively and then we have to plot the correlation of PRN 8 with all three delayed PRN 8 chips. So, first of all I computed the PRN 8 using `generateCAcode()`. Thereafter, I computed delayed PRNs by -1, 0, and chips using `circshift()` function, then I computed the correlation of PRN 8 with these delayed PRNs using `xcorr()` function and then

plotted the subplots of these correlations using `plot()` function after doing appropriate labelling.

The plot is shown below :

From this plot, the peak for 1st case is occurring at x = 1024 (i.e. 1023- (-1)), for 2nd case at x = 1023 (i.e. 1023-0), and for 3rd case at x = 1022 (i.e. 1023-1)

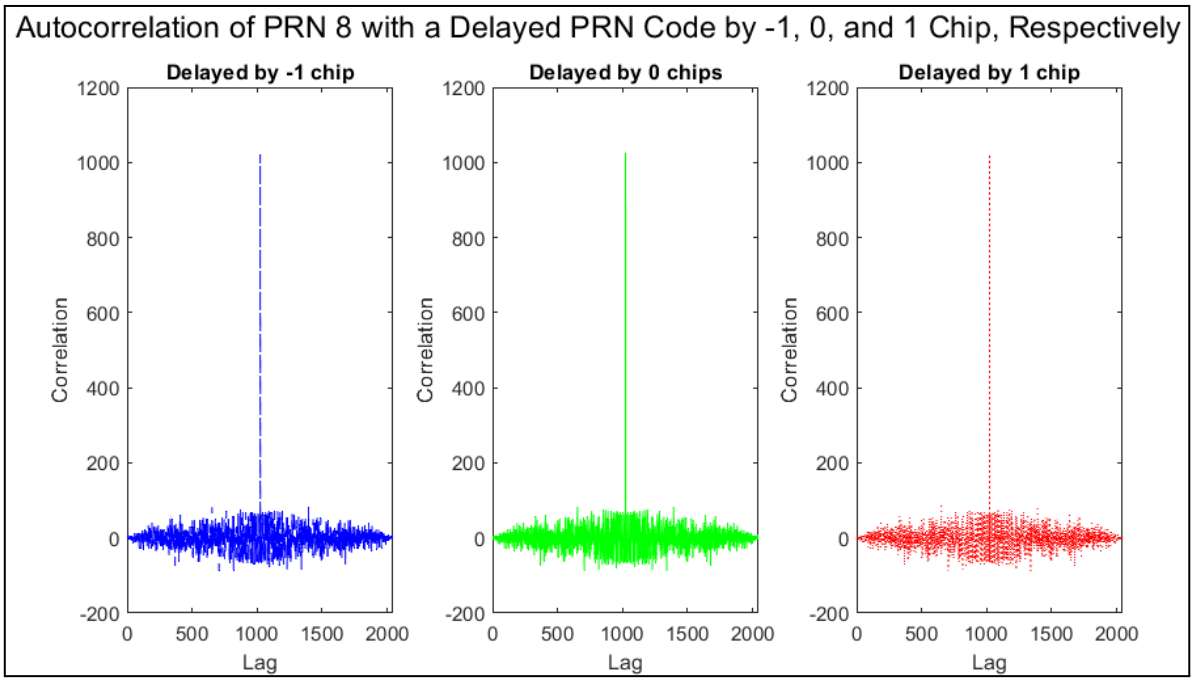

Fig 3

4. **Write a MATLAB/ Python/C/C++ program to compute circular autocorrelation of PRN 8 with a noisy PRN code delayed by 200 chips and plot the results. Assume the noise is white Gaussian additive and generated with mean zero and standard deviation of 4.**

MATLAB CODE :

```
%Q4
% Generate a PRN 8 code
prn8 = generateCAcode(8);

% Generate a noisy PRN code delayed by 200 chips
delayed_prn8 = circshift(prn8, 200);

% Normal dist. with size of transpose(prn8), delayed by 200 chips
a = randn(length(delayed_prn8), 1) * 4 + 0; % this is vector (size = 1023x1)

% noise
b = a.'; % Transpose of a to make size = 1x1023

% adding noise to the delayed_prn8
noisy_delayed_prn8 = delayed_prn8 + b;

% Compute the circular autocorrelation
circular_autocorrelation = xcorr(prn8, noisy_delayed_prn8);
```

```
% Plot the circular autocorrelation
figure;
plot(circular_autocorrelation);
xlabel('Lag');
ylabel('Correlation');
title('Circular Autocorrelation of PRN 8 with a Noisy PRN Code Delayed by 200
Chips');
```

In this problem, I have to compute the autocorrelation of PRN 8 with a noisy PRN 8 delayed by 200 chips, given that the noise is additive and follows a gaussian with mean = 0, and standard deviation = 4. So, again I computed the PRN 8 using `generateCAcode()` function, thereafter I computed the delayed PRN 8 code by using `circshift()` function. Also I have to make this delayed PRN noisy by adding gaussian of mean = 0, standard deviation = 4, Therefore I computed a gaussian whose size is equal to the length of delayed PRN 8 having mean=0 and s.d = 4 by using `randn()` function, after this I get a vector of size *(2023* x *1)*, but for adding this gaussian profile in delayed PRN (of size *1* x *1023*), they both need to be of equal size, therefore I have taken the transpose of it to get a gaussian profile of size *1* x *1023*. Thereafter I added this gaussian profile in delayed PRN 8 to make delayed PRN8 noisy. Now after that I computed the correlation between PRN 8 by using `corr()` function and plotted the correlation of PRN 8 with noisy delayed PRN 8 by using `plot()` function by doing appropriate labelling.
The plot is shown below :
From this plot, the peak is occurring at x = 823, with correlation of 941.85

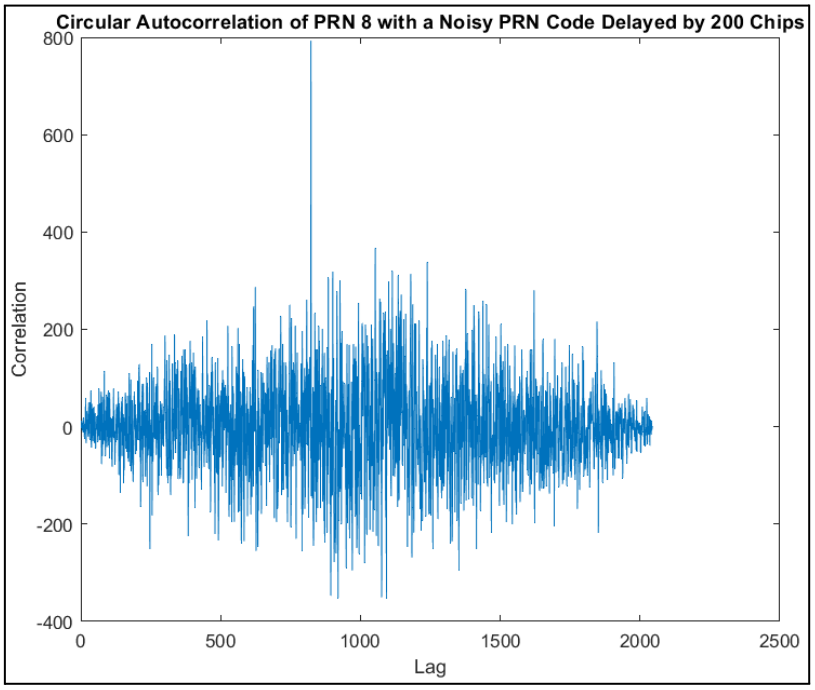

Fig 4

Link to matlab file : https://github.com/Vish-2003/GPS-fundamental-assignment.git