

BÁO CÁO THỰC TẬP

Người thực hiện: Nguyễn Quang Trường

Thời gian: 07/07/2022 đến 16/07/2022

1. Nội dung công việc:

- Tìm hiểu về mạch Zynq 7000
- Cài đặt phần mềm lập trình nhúng Vivado và Petalinux
- Cài đặt môi trường ubuntu linux
- Làm quen với linux programming với ngôn ngữ C
- Lập trình bài toán: Sử dụng luồng (Thread) trong C để lấy mẫu thời gian thực tế của hệ thống (đơn vị nanosecond) với những thời gian lấy mẫu cho trước.

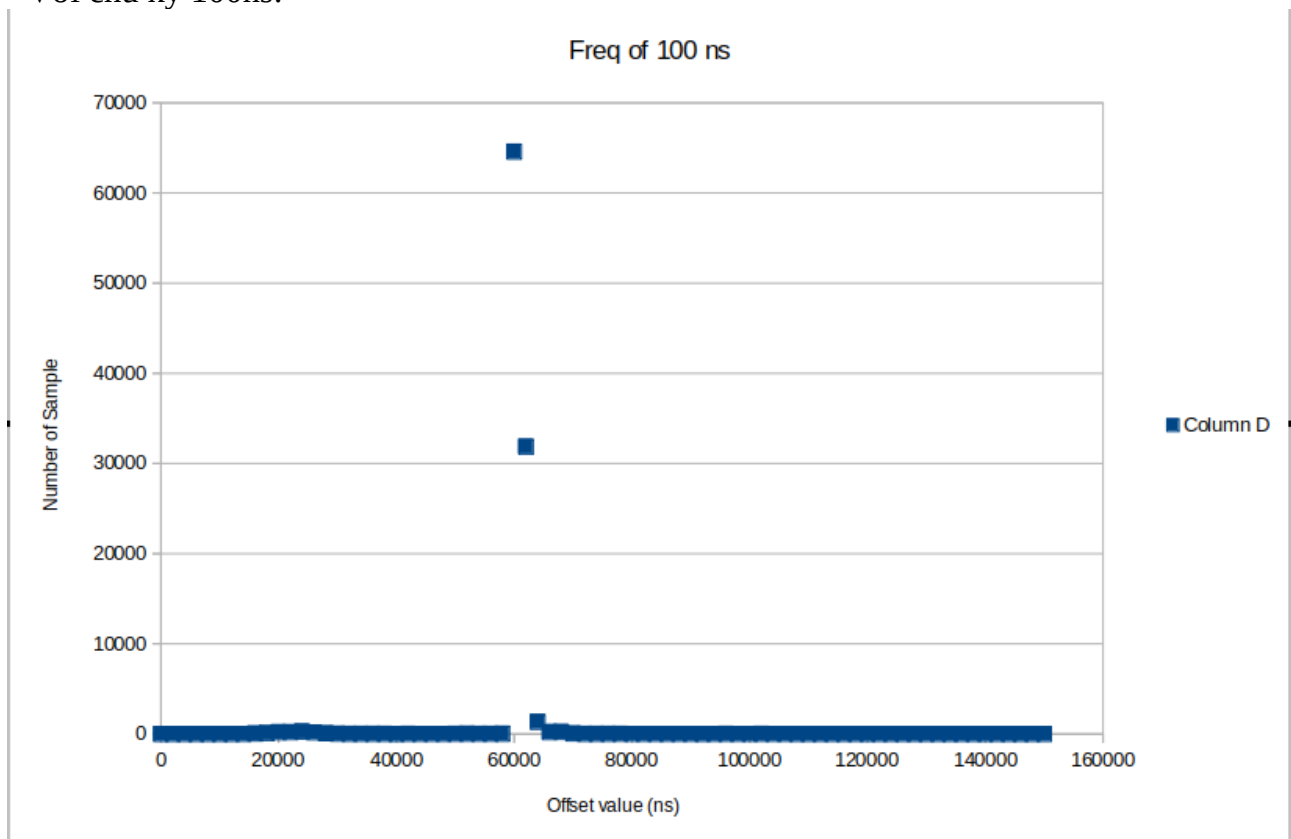
Các bước giải quyết bài toán:

- Tạo các luồng thực hiện từng nhiệm vụ của bài toán:
- INPUT thực hiện lấy giá trị của chu kỳ lấy mẫu thời gian từ file freq.txt
- SAMPLE thực hiện đọc giá trị thời gian hiện tại của hệ thống với đơn vị là nanosecond
- LOGGING tính toán giá trị offset giữ hai lần lấy mẫu, từ đó in thời gian lấy mẫu và giá trị offset vào file time_and_interval.txt
- Thư viện “pthread.h” được sử dụng để làm việc với luồng với các hàm pthread_create() để tạo luồng, pthread_join() thực hiện đảm bảo luồng đã đóng, pthread_mutex_t khóa, mở khóa, chờ luồng thực hiện quá tr.
- Thư viện “time.h” được khai báo để sử dụng các hàm clock_gettime() lấy thời gian thực của hệ thống, clock_nanosleep() để tạo khoảng thời gian trong chu kỳ lấy mẫu.

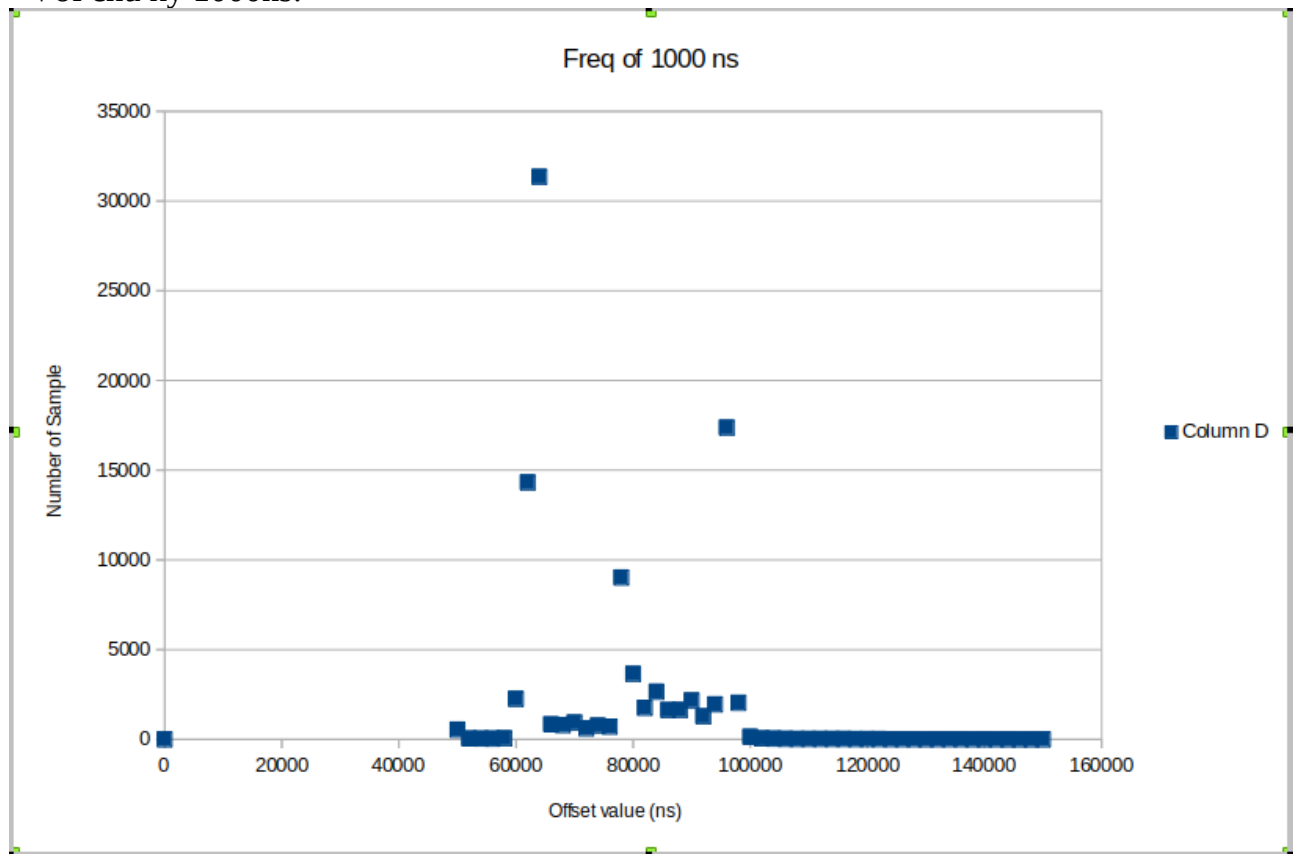
2. Kết quả và đánh giá

Kết quả được thể hiện trong các đồ thị sau:

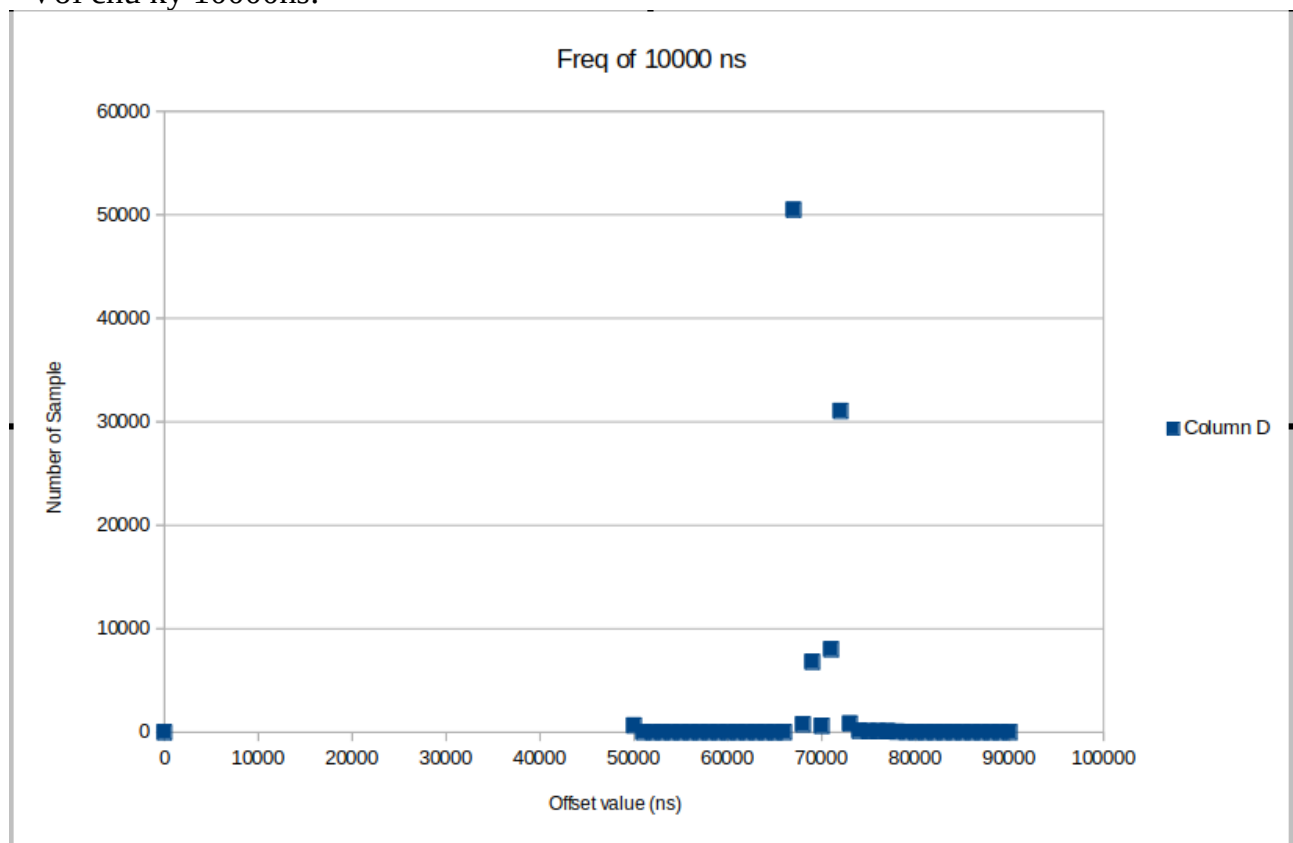
- Với chu kỳ 100ns:



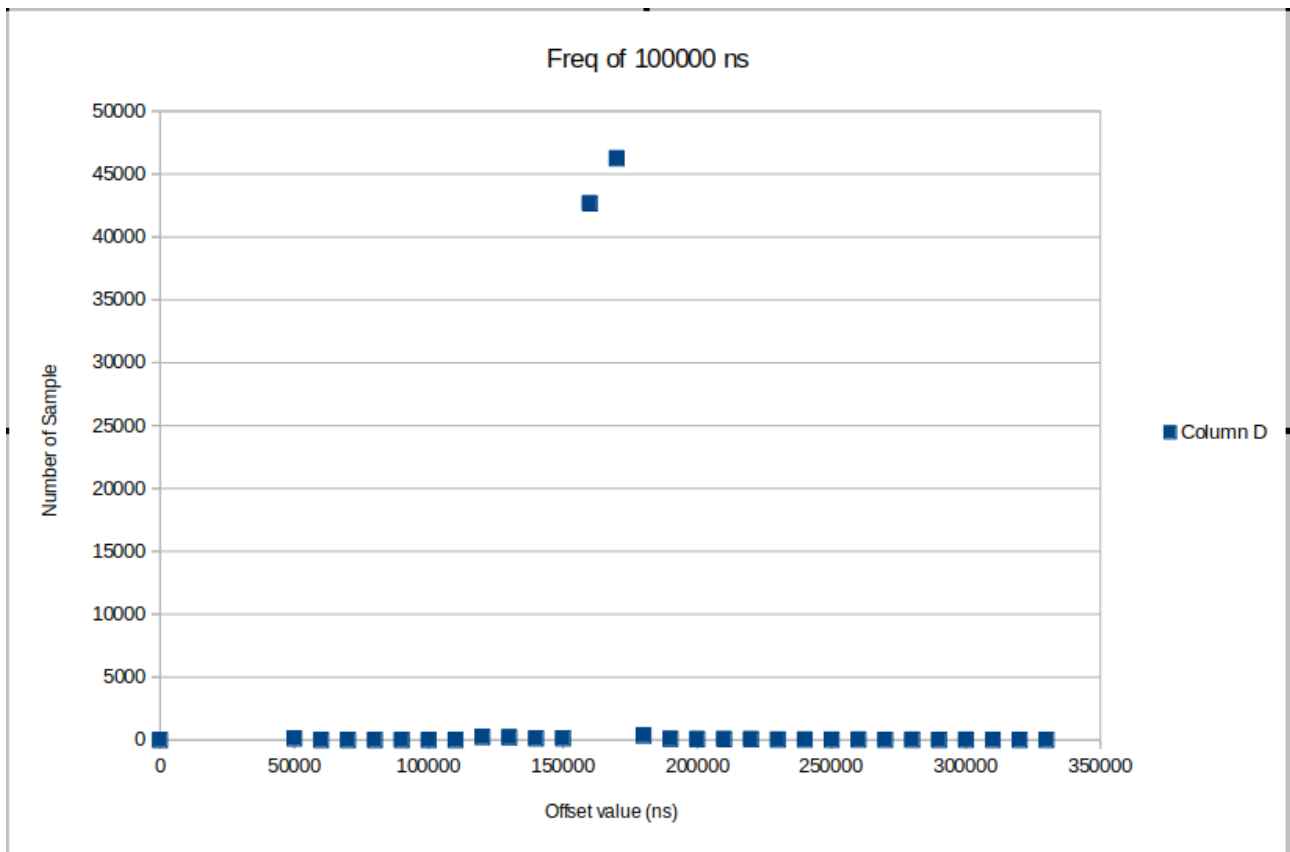
- Với chu kỳ 1000ns:



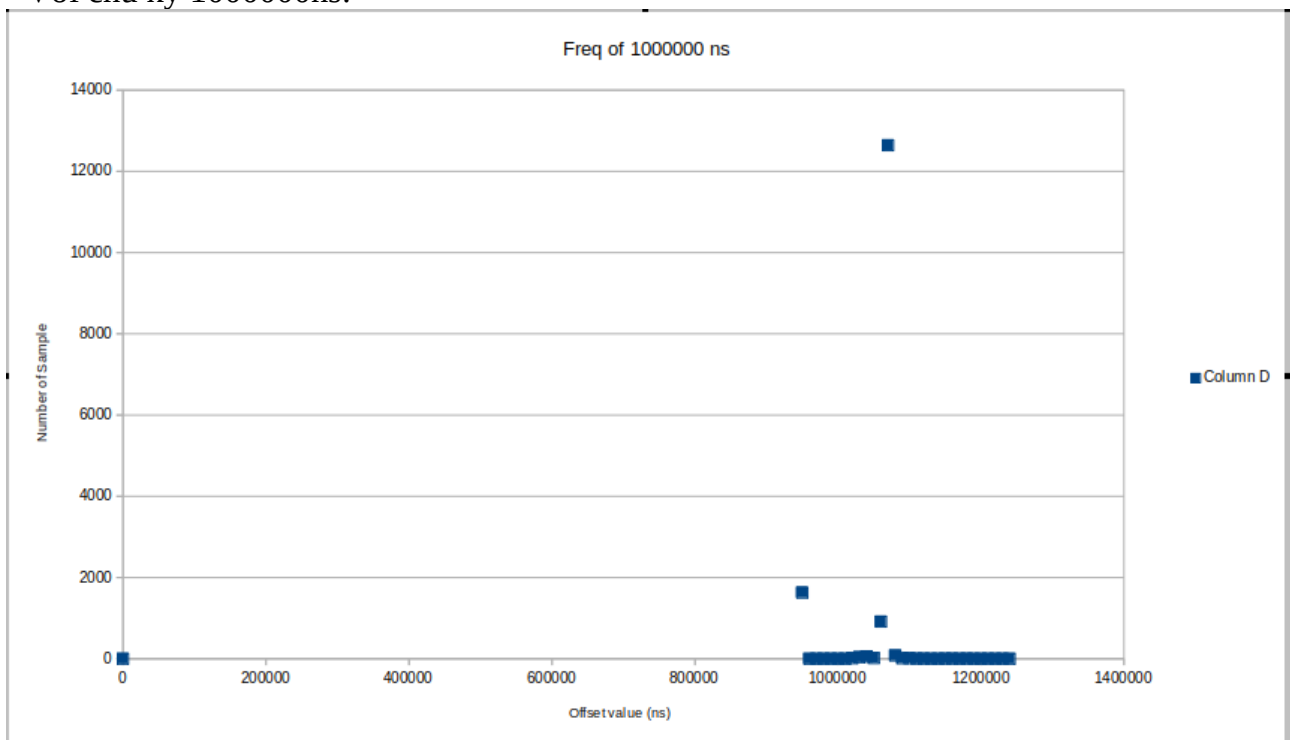
- Với chu kỳ 10000ns:



- Với chu kỳ 100000ns:



- Với chu kỳ 1000000ns:



Đánh giá kết quả:

- Ở các chu kỳ lấy mẫu 100ns, 1000ns, 1000ns và 10000ns, giá trị offset giữa hai lần lấy mẫu có độ sai lệch rất lớn. Các giá trị offset ở các chu kỳ này thường có giá trị khoảng từ 50000ns đến 80000ns. Từ đó cho thấy chương trình không thể lấy mẫu chính xác ở các chu kỳ này.

- Ở chu kỳ lấy mẫu 100000ns và 1000000ns, giá trị offset có độ chênh lệch với giá trị chu kỳ là 60000ns đến 70000ns. Khoảng sai số này còn khá lớn và cần được tối ưu về chương trình. Các giá trị offset ở hai chu kỳ này khá ổn định và phân bố tập chung gần với giá trị lấy mẫu. Từ đó cho thấy chương trình lấy mẫu thời gian hoạt động ổn định trong hai chu kỳ này.

3. Khó khăn và hướng khắc phục

Khó khăn:

- Chưa nắm vững các nội dung và kiến thức về luồng và thời gian trong hệ thống linux
- Chưa làm quen với lập trình với Thread và Time trong C
- Thuật toán ban đầu còn chưa được tối ưu và hoàn thiện

Hướng khắc phục:

- Tìm hiểu kiến thức về Thread và Time trong sách The Linux Programming Interface
- Chỉnh sửa chương trình code với sự góp ý từ các thành viên và mentor hướng dẫn