Chương 1: Quy trình ADC/DAC

Xử lý tín hiệu số

Tín Vũ

tinvu1309@gmail.com

Mục lục

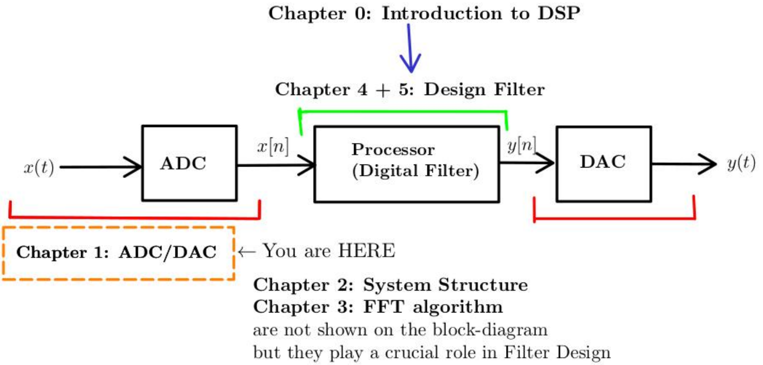
Giới thiệu playlist

- Mình là Tín Vũ, hiện đang là sinh viên học tại Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội. Mình tạo playlist video này để hỗ trợ các bạn học môn **Xử lý tín hiệu số**.
- Khác với môn học tiên quyết Tín hiệu hệ thống trước đó, bài giảng môn học này **hoàn toàn bám sát với đề cương và giáo trình nội bộ** của trường mình, nên các bạn trường khác cần phải lưu ý rất kĩ điều này.
- Không chỉ dừng lại ở lý thuyết, playlist này **có bổ sung hướng dẫn lập trình cơ bản bằng GNU Octave/Matlab** để vẽ phổ tín hiệu, đáp ứng tần số và thiết kế bộ lọc.
- Môn học này bao gồm **6 chương**, các chương đều liên quan rất chặt chẽ với nhau nên hãy học cẩn thận ngay từ Chương 0 để ôn thi cuối kì đỡ vất vả.

Tài liệu tham khảo

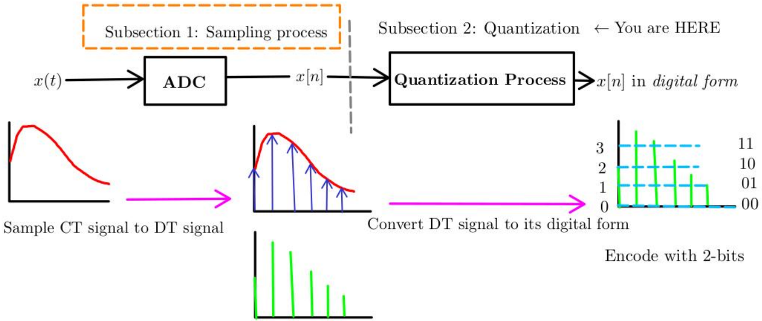
- Tài liệu tham khảo chính: Giáo trình Xử lý tín hiệu số (Nguyễn Linh Trung, Trần Đức Tân, Huỳnh Hữu Tuệ, ĐHCN, 2012).
- Tài liệu tham khảo phụ: Discrete-time Signal Processing (Alan V.Oppenheim, 2nd edition).

Quy trình xử lý tín hiệu số



Hình: DSP Learning Process

Quy trình ADC

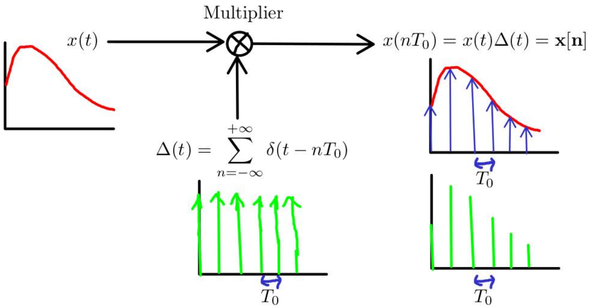


Hình: ADC block diagram

Quy trình ADC

- Quy trình lấy mẫu

Trong môn học tiên quyết Tín hiệu và hệ thống trước, ta đã đề cập định tính về quy trình lấy mẫu để chuyển đổi tín hiệu liên tục thành tín hiệu rời rạc. Bây giờ, ta muốn tìm một mối liên hệ rất chặt chẽ về mặt toán học giữa tín hiệu $x(t)$ và $x[n]$ trong **miền thời gian** và **miền tần số**, và từ kết quả này ta sẽ xây dựng phương pháp chuyển đổi ngược tín hiệu rời rạc thành tín hiệu liên tục.



Hình: Sampling process

Quy trình ADC

Trong môn học Xử lý tín hiệu số (từ giờ ta sẽ gọi tắt là DSP - Digital Signal Processing), các kí hiệu tần số ngược với môn Tín hiệu và hệ thống, ta quy ước kí hiệu Ω cho tần số **liên tục** và ω cho tần số **rời rạc**. Từ mối liên hệ giữa $x(t)$ và $x[n]$, ta có:

$$x[n] = x(nT_0) = x(t)\Delta(t) = x(t)\sum_{n=-\infty}^{+\infty}\delta(t-nT_0)$$

Dễ dàng nhận thấy $\Delta(t)$ có dạng **chuỗi Fourier liên tục (CTFS)**, ta thử tìm cách biểu diễn tín hiệu này về dạng chuỗi CTFS để phân tích trong miền tần số như sau:

$$\Delta(t) = \sum_{n=-\infty}^{+\infty} c_n e^{jn\Omega_0 t}$$

$$c_n = \frac{1}{T_0}\int_{T_0}\Delta(t)e^{-jn\Omega_0 t}dt = \frac{1}{T_0}\int_{-\frac{T_0}{2}}^{+\frac{T_0}{2}}\left[\sum_{n=-\infty}^{+\infty}\delta(t-nT_0)\right]e^{-jn\Omega_0 t}dt = \frac{1}{T_0}$$

Vậy ta thu được chuỗi CTFS của tín hiệu $\Delta(t)$:

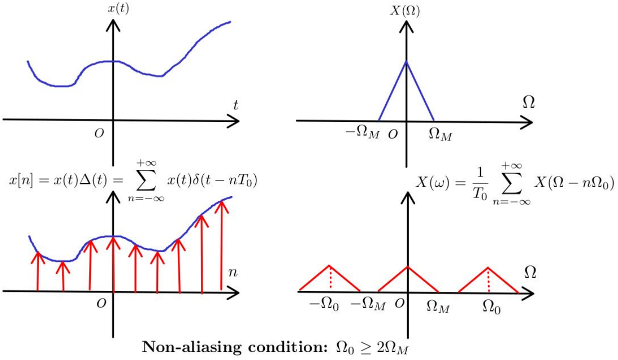
$$\Delta(t) = \frac{1}{T_0}\sum_{n=-\infty}^{+\infty} e^{jn\Omega_0 t}$$

Ta xét biến đổi Fourier liên tục (CTFT) và tính chất **dịch trong miền tần số**:

Quy trình ADC

$$X(\omega) = \mathscr{F}(x(t)\Delta(t)) = \frac{1}{T_0}\mathscr{F}\left(x(t)\sum_{n=-\infty}^{+\infty} e^{jn\Omega_0 t}\right) = \frac{1}{T_0}\sum_{n=-\infty}^{+\infty} X(\Omega - n\Omega_0)$$

Ta phác họa đáp ứng biên độ của tín hiệu liên tục và rời rạc tương ứng, gọi Ω_M là tần số biên (tức thành phần tần số lớn nhất trong phổ của tín hiệu liên tục):



Hình: CT and FT signal spectrum

Quy trình ADC

Để phổ của tín hiệu rời rạc không bị gập (aliasing or folding), từ hình vẽ ta hiển nhiên thấy:

$$\Omega_0 \geq 2\Omega_M \Leftrightarrow f_0 \geq 2f_M$$

Chúng ta gọi hằng số:

$$f_N = 2f_M$$

là tốc độ Nyquist của tín hiệu $x(t)$. Vậy để phổ của tín hiệu rời rạc không bị chồng lấn (overlap) thì ta phải **lấy mẫu tín hiệu $x(t)$ với tốc độ tối thiểu bằng hai lần thành phần tần số cao nhất trong tín hiệu liên tục**.
Hiển nhiên từ công thức:

$$x[n] = x(nT_0)$$

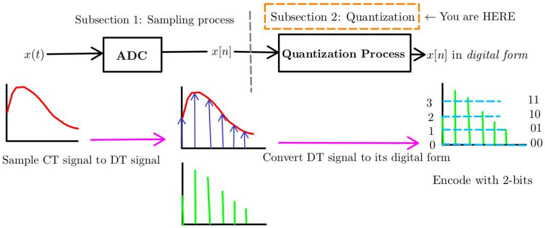
với cặp $\omega - \Omega$ là thành phần tần số tương ứng giữa tín hiệu rời rạc và tín hiệu liên tục, ta dễ dàng suy ra mối liên hệ:

$$\omega = \Omega T_0$$

Kết quả trên hoàn toàn khớp với phân tích định tính ta đã thực hiện trong môn học Tín hiệu và hệ thống trước, phổ của tín hiệu rời rạc **luôn tuần hoàn**.

Quy trình ADC

- Quy trình lượng tử hóa



Hình: Quantization Process

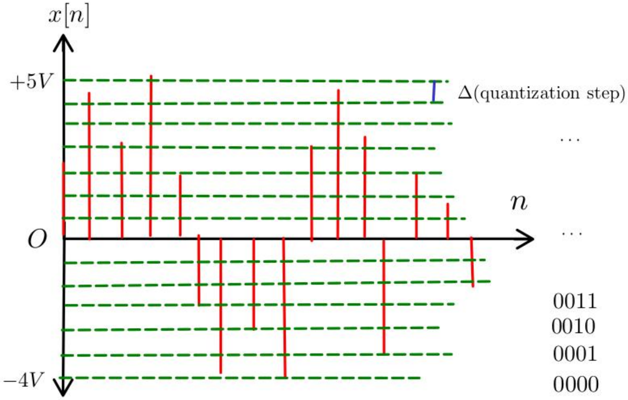
Sau khi đã rời rạc hóa tín hiệu liên tục, ta muốn "định mức" chúng vào các mức lượng tử để mã hóa thành các số nhị phân tương ứng (do hệ thống máy tính số làm việc với các bit nhị phân). Quy trình này được gọi là lượng tử hóa, từ hình minh họa dưới đây, ta định nghĩa các khái niệm:

- Mức lượng tử Δ:

$$\Delta = \frac{V_{max} - V_{min}}{2^n}$$

- Số bit lượng tử n.

Quy trình ADC



Hình: Using 4 bits to quantizate DT signal

Quy trình ADC

Ví dụ: tín hiệu $x(t) = 2\cos(500\pi t)$ được lấy mẫu với tốc độ $f_{sam} = 3k$ (sample/s), sau đó được lượng tử hóa với độ phân giải 8 bit.

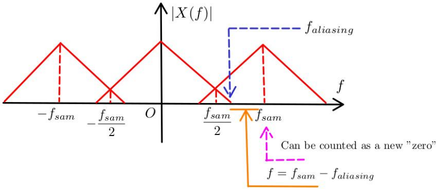
Dễ thấy $f_N = 2f_M = 500Hz$, và $f_{sam} > f_N$ nên **hiện tượng gập phổ không xảy ra**. Tín hiệu rời rạc được xác định qua phương trình:

$$x[n] = x(nT_{sam}) = x\left(\frac{n}{f_{sam}}\right) = 2\cos\left(\frac{\pi n}{6}\right)$$

Hiển nhiên $-2 \leq x[n] \leq +2$, nên áp dụng công thức tìm mức lượng tử Δ ta thu được:

$$\Delta = \frac{V_{max} - V_{min}}{2^n} = \frac{2-(-2)}{2^8} = \frac{1}{64}$$

Nếu ta lấy mẫu với tốc độ $f_{sam} = 400$ (sample/s), phổ của tín hiệu rời rạc sẽ bị ảnh hưởng như thế nào? Hiển nhiên ta thấy khi lấy mẫu ở tốc độ thấp **hiện tượng gập phổ có xảy ra**.



Hình: Aliasing Phenomenon

Quy trình ADC

Từ công thức:

$$0 < \pm f_{aliasing} + kf_{sam} < \frac{f_{sam}}{2}$$

và công thức tìm tín hiệu sau khi lấy mẫu:

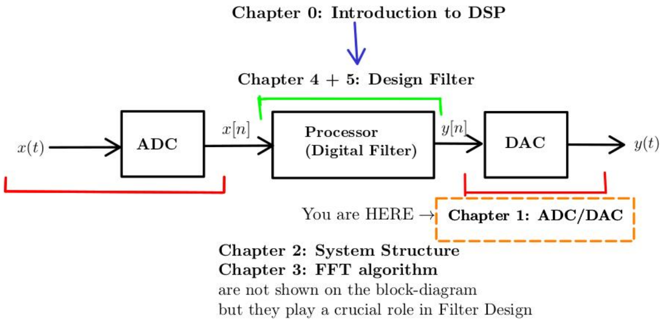
$$x[n] = x\left(\frac{nf}{f_{sam}}\right)$$

Ta dễ dàng xác định được tần số của tín hiệu sau khi bị gập phổ, và tín hiệu rời rạc tương ứng thu được là:

$$x[n] = x\left(\frac{nf}{f_{sam}}\right) = 2\cos\left(\frac{3\pi n}{8}\right)$$

Ta có thể tăng độ phân giải từ 8 bit lên 16 hoặc 32 bit để tín hiệu rời rạc thu được dưới dạng mã nhị phân chính xác hơn. Nói chung, độ phân giải càng cao thì tín hiệu rời rạc càng chính xác.

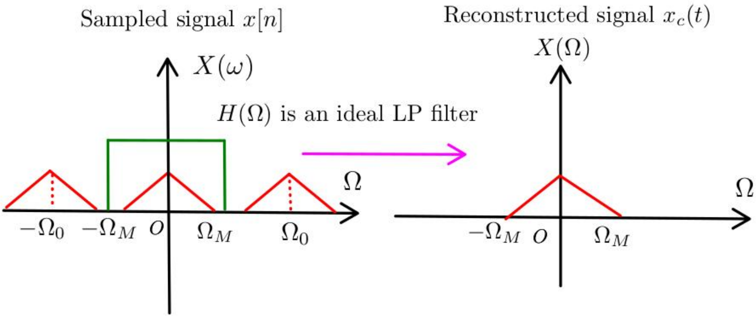
Quy trình DAC



Hình: DSP Learning process

Quy trình DAC

Từ phổ tần số của cặp tín hiệu $x(t)$ và $x[n]$ đã được vẽ ở phần trước; ta dễ dàng thấy rằng để khôi phục được **phổ** tín hiệu $x(t)$ hoàn chỉnh từ $x[n]$, ta cần thiết kế một **bộ lọc thông thấp** (Low-pass filter) với điều kiện tần số cắt (cut-off frequency) như sau (ta tạm thời bỏ qua yếu tố khôi phục độ lớn bằng bộ khuếch đại do hao hụt bởi hằng số, mà chỉ quan tâm đến khôi phục phổ):



Hình: Reconstruct CT signal spectrum

Quy trình DAC

Với đáp ứng biên độ của bộ lọc thông thấp $H(\Omega)$:

$$H(\Omega) = \begin{cases} 1 & (|\Omega| \leq \frac{\Omega_0}{2}) \\ 0 & (\text{otherwise}) \end{cases}$$

Ta chọn tần số cắt của bộ lọc

$$\Omega_C = \frac{\Omega_0}{2}$$

Dễ thấy **thành phần tần số lớn nhất** mà bộ lọc thông thấp có thể khôi phục được hoàn hảo từ tín hiệu rời rạc là:

$$\Omega_M \leq \frac{\Omega_0}{2} \Leftrightarrow f_M \leq \frac{f_0}{2}$$

Ta gọi hằng số $f_N = \frac{f_0}{2}$ là tần số Nyquist của tín hiệu, là thành phần tần số **lớn nhất** của tín hiệu tương tự $x(t)$ có thể khôi phục lại được bằng bộ lọc thông thấp trong khối DAC có **độ lớn bằng một nửa tần số lấy mẫu**.
Ta phải phân biệt rất rạch ròi giữa hai khái niệm tốc độ Nyquist và tần số Nyquist, chúng có kí hiệu giống nhau nhưng **thứ nguyên và ý nghĩa vật lý hoàn toàn khác nhau**, cũng như hai mặt sấp-ngửa của một đồng xu vậy.

Quy trình DAC

Sau khi đã thấy mối quan hệ giữa $x[n]$ và $x_c(t)$ trong miền tần số, ta muốn tìm mối quan hệ của chúng tương ứng trong miền thời gian; và so sánh sự sai khác giữa tín hiệu $x_c(t)$ với tín hiệu $x(t)$. Ta có:

$$\mathscr{F}^{-1}[X_c(\Omega)] = \mathscr{F}^{-1}[X(\omega)H(\Omega)] \Rightarrow x_c(t) = x(nT_0) * h(t)$$

Có nhiều cách để tìm đáp ứng tần số của $H(\Omega)$, các bạn có thể tìm gián tiếp bằng tính chất đối ngẫu (duality) của CTFT, nhưng ở đây mình sẽ tìm trực tiếp bằng công thức ICTFT:

$$H(\Omega) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} H(\Omega)e^{j\Omega t}d\Omega = \frac{1}{2\pi}\int_{-\frac{\Omega_0}{2}}^{+\frac{\Omega_0}{2}} e^{j\Omega t}d\Omega = \frac{\sin\left(\frac{\Omega_0 t}{2}\right)}{\pi t} = \frac{\sin\left(\frac{\pi t}{T_0}\right)}{\pi t}$$

$$= T_0 \text{sinc}\left(\frac{t}{T_0}\right)$$

với hàm sinc được định nghĩa:

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$$

Quy trình DAC

Vậy hiển nhiên ta có:

$$\begin{aligned} x_c(t) = x(nT_0) * h(t) &= \left[\sum_{n=-\infty}^{+\infty} x(t)\delta(t - nT_0) \right] * T_0 \text{sinc}\left(\frac{t}{T_0}\right) \\ &= T_0 \sum_{n=-\infty}^{+\infty} x(t) \text{sinc}\left(\frac{t - nT_0}{T_0}\right) \end{aligned}$$

Đây chính là phương trình của tín hiệu liên tục $x_c(t)$ có thể khôi phục được từ khối DAC. Công thức này còn được gọi là **công thức nội suy của tín hiệu** $x(t)$.