# UART

## [CẤU TRÚC VÀ CÁC THÀNH PHẦN TRONG](https://nguyenquanicd.blogspot.com/2017/06/cancontroller-area-netwwork-cau-truc-va.html) UART

### UART là gì ?

UART (Universal Asynchronous Receiver/Transmitter) là một trong những phương thức giao tiếp sớm nhất được áp dụng cho máy tính và nguồn gốc của nó ít nhất là từ những năm 1960 khi nó được sử dụng để kết nối máy tính mini với máy đánh chữ từ xa - 'teletypes', như trước đây được gọi phổ biến hơn. Về cơ bản, đây là những bàn phím điện có khả năng truyền các thao tác gõ phím đến và in ra phản hồi từ máy chủ. Trong những năm 1970, UART được sử dụng để cho phép các máy vi tính đời đầu lưu trữ và tải các chương trình cũng như dữ liệu từ băng cassette. Trong những thập kỷ tiếp theo, nó được sử dụng để khiến máy tính cá nhân giao tiếp với các dịch vụ trực tuyến thông qua modem.

Cho đến khi USB xuất hiện, máy tính cá nhân đã có cổng nối tiếp để kết nối với các thiết bị khác. UART là phương tiện giao tiếp cơ bản. Ngày nay, UART được sử dụng chủ yếu bởi các thiết bị dựa trên vi điều khiển và các thiết bị phức tạp hơn, bao gồm cả imp.

UART có nhiều tên, nhưng dù được gọi là gì đi nữa, nó luôn liên quan đến việc gửi dữ liệu qua hai dây - một dây để truyền, dây kia để nhận dữ liệu đến. Thông tin được truyền đi từng bit nhị phân; vì vậy nó là một phương thức liên lạc 'nối tiếp'. Các bit này được nhóm lại với nhau thành 'khung' - một định dạng cố định để truyền tải một phần dữ liệu có ý nghĩa.

UART được cho là 'phổ quát' vì các tham số của nó - tốc độ, kích thước dữ liệu, v.v. - không cố định và có thể được định cấu hình để đáp ứng nhu cầu của một yêu cầu giao tiếp nhất định, mặc dù điều này có nghĩa là cả hai bên của cuộc trò chuyện cần phải có sẵn thống nhất về các thông số này. Nó 'không đồng bộ' vì nó không yêu cầu đồng hồ do người gửi cung cấp để đồng bộ hóa việc truyền và nhận dữ liệu.

### Cấu trúc phần cứng

A red and green lines

Description automatically generated

**Hình 1.** Cấu trúc phần cứng UART

Chuẩn giao tiếp UART hoạt động theo cơ chế truyền dữ liệu bất đồng bộ (asynchronous), tức là không có một tín hiệu đông hồ chung (xung clock) giữa hai thiết bị để đồng bộ việc truyền nhận dữ liệu . Thay vào đó goa thức UART sử dụng hai tín hiệu riêng biệt là RX(Receiver) và TX(Transmitter) để truyền và nhận dữ liệu giữa các thiết bị.

Truyền thông bất đồng bộ (asynchronous transmission) không sử dụng xung clock để đồng bộ mà sử dụng bit start và bit stop cho việc bắt đầu và kết thúc quá trình truyền dữ liệu, đồng thời dùng bit chẵn (parity bit) để đảm bảo tính toàn vẹn dữ liệu, và những quy ước thuộc về phần cứng giữa 2 phía truyền-nhận.

## CẤU TRÚC KHUNG TRUYỀN DỮ LIỆU VÀ KHUNG NHẬN DỮ LIỆU

### Khung truyền dữ liệu của giao thức UART

UART sẽ truyền dữ liệu nhận được từ một bus dữ liệu (Data Bus). Bus dữ liệu được sử dụng để gửi dữ liệu đến UART bởi một thiết bị khác như CPU, bộ nhớ hoặc vi điều khiển. Dữ liệu được chuyển từ bus dữ liệu đến UART truyền ở dạng song song. Sau khi UART truyền nhận dữ liệu song song từ bus dữ liệu, nó sẽ thêm một bit start, một bit chẵn lẻ và một bit stop, tạo ra gói dữ liệu. Tiếp theo, gói dữ liệu được xuất ra nối tiếp từng bit tại chân Tx. UART nhận đọc gói dữ liệu từng bit tại chân Rx của nó. UART nhận sau đó chuyển đổi dữ liệu trở lại dạng song song và loại bỏ bit start, bit chẵn lẻ và bit stop. Cuối cùng, UART nhận chuyển gói dữ liệu song song với bus dữ liệu ở đầu nhận.

UART truyền dữ liệu nối tiếp, theo một trong ba chế độ:

* Full duplex: Giao tiếp đồng thời đến và đi từ mỗi master và slave.
* Half duplex: Dữ liệu đi theo một hướng tại một thời điểm.
* Simplex: Chỉ giao tiếp một chiều.

![A diagram of a number

Description automatically generated with medium confidence]()

**Hình 2.** Cấu trúc khung truyền UART

Dữ liệu truyền qua UART được tổ chức thành các gói. Mỗi gói chứa 1 bit bắt đầu, 5 đến 9 bit dữ liệu (tùy thuộc vào UART), một bit kiểm tra chẵn lẻ tùy chọn và 1 hoặc 2 bit dừng.

IDLE (Mức trạng thái lý tưởng) : Khi mà không có sự truyền nhận ở trên đường giao tiếp thì IDLE sẽ luôn luôn ở mức cao để phát hiện điều kiện bắt đầu của (Start bit)

Start bit (Bit khởi đầu): Đường truyền dữ liệu trong giao tiếp UART thường được giữ ở mức **điện áp cao(IDLE)** khi nó không truyền dữ liệu. Để bắt đầu truyền dữ liệu, UART truyền sẽ kéo đường truyền từ mức cao xuống mức thấp trong một chu kỳ đồng hồ. Khi UART 2 phát hiện sự chuyển đổi điện áp cao xuống thấp, nó bắt đầu đọc các bit trong khung dữ liệu ở tần số của tốc độ truyền (Baud rate).

Frame (Khung dữ liệu): Khung dữ liệu chứa dữ liệu thực tế đang được truyền. Nó có thể dài từ 5 bit đến 8 bit nếu sử dụng bit Parity (bit chẵn lẻ). Nếu không sử dụng bit Parity, khung dữ liệu có thể dài 9 bit. Trong hầu hết các trường hợp, dữ liệu được gửi với bit LSB (bit có trọng số thấp nhất) trước tiên.

Data bits (Bit dữ liệu) [0-8]: Nếu muốn gửi dữ liệu thì phải gửi 1 chuỗi bit. 1 thời điểm chỉ gửi được 1 bit data. Nếu gửi 1 byte thì gửi 8 lần. Gửi bit có trọng số thấp trước (LSB)

(Parity) Bit chẵn lẻ: Bit chẵn lẻ là một cách để UART nhận cho biết liệu có bất kỳ dữ liệu nào đã thay đổi trong quá trình truyền hay không. Bit có thể bị thay đổi bởi bức xạ điện từ, tốc độ truyền không khớp hoặc truyền dữ liệu khoảng cách xa. Sau khi UART nhận đọc khung dữ liệu, nó sẽ đếm số bit có giá trị là 1 và kiểm tra xem tổng số là số chẵn hay lẻ. Nếu bit chẵn lẻ là 0 (tính chẵn), thì tổng các bit 1 trong khung dữ liệu phải là một số chẵn. Nếu bit chẵn lẻ là 1 (tính lẻ), các bit 1 trong khung dữ liệu sẽ tổng thành một số lẻ. Khi bit chẵn lẻ khớp với dữ liệu, UART sẽ biết rằng quá trình truyền không có lỗi. Nhưng nếu bit chẵn lẻ là 0 và tổng là số lẻ; hoặc bit chẵn lẻ là 1 và tổng số là chẵn, UART sẽ biết rằng các bit trong khung dữ liệu đã thay đổi.

(Stop bit) Bit dừng: Để báo hiệu sự kết thúc của gói dữ liệu, UART gửi sẽ điều khiển đường truyền dữ liệu từ điện áp thấp đến điện áp cao trong ít nhất hai khoảng thời gian bit.

### Khung nhận dữ liệu của giao thức UART

A diagram of a data analysis

Description automatically generated

Bên nhận sẽ đọc liên tục đường **IDLE** của RX. Khi phát hiện sự thay đổi logic từ 1 xuống 0 thì **Start bit** được đọc và bắt đầu nhận dữ liệu. **LSB** nhận trước.

A diagram of a diagram

Description automatically generated

Delay ½ chu kì và tiếp tục delay 1 chu kì và bắt đầu lấy mẫu. Bởi vì dữ liệu ở giữa ổn định nhất. Cứ thế sau 1 chu kì lại đọc tiếp.

**Chú ý:** Truyền nhận phải cùng tốc độ Baus bởi vì truyền bất đồng bộ.

A table with numbers and a few lines

Description automatically generated with medium confidence