

SeeHorse

Cho 2 mảng (mảng 1 và mảng 2) số nguyên 8 bits, mỗi mảng gồm 65 phần tử (tương đương với 65 bước di chuyển của cá ngựa). Tại thời điểm bắt đầu, 2 con cá ngựa sẽ ở vị trí bất kỳ của mỗi mảng, giả dung index1, index2 để quản lý vị trí này. Viết chương trình cho phép di chuyển một trong 2 con cá ngựa thông qua lệnh điều khiển qua UART với các điều kiện di chuyển như sau.

- Số bước di chuyển giới hạn từ 1 đến 6 bước.
- Khi cá ngựa sau di chuyển số bước tới đúng vị trí của cá ngựa trước, thì cá ngựa trước bị đá về vị trí ban đầu (ví dụ cá ngựa 1 đang ở vị trí 10, cá ngựa 2 đang ở vị trí 15, điều khiển cá ngựa 1 di chuyển 5 bước -> vị trí cá ngựa 1 ở vị trí 15, cá ngựa 2 bị đá về vị trí ban đầu).
- Khi cá ngựa sau có số bước di chuyển vượt qua vị trí cá ngựa trước thì không được phép di chuyển (ví dụ cá ngựa 1 đang ở vị trí 10, cá ngựa 2 đang ở vị trí 13, điều khiển cá ngựa 1 di chuyển 5 bước -> không cho phép di chuyển).
- Không được phép di chuyển vượt qua vị trí đích (ví dụ vị trí ban đầu của cá ngựa 1 là 10 -> vị trí đích phải là vị trí 9 -> nếu cá ngựa di chuyển 1 vòng và quay về vị trí 7, sau đó nhận lệnh điều khiển di chuyển 5 bước -> không cho phép, nếu chỉ nhận lệnh di chuyển 2 bước -> cho phép di chuyển, và báo về đã về đích)

Frame truyền của UART được định nghĩa như sau:

STX	TxRxID	Length	Data					ETX
0x02	ID byte	Len byte	byte1	byte 2	byte n	0x03

- **STX**: Start Of Text, chỉ ra việc bắt đầu một frame.
- **TxRxID**: Chỉ ra mục đích của frame để làm gì, xem phần định nghĩa các frame bên dưới.
- **Length**: Chỉ ra số byte data theo sau, giá trị từ 0 đến 255.
- **Data**: Data trong frame.
- **ETX**: End Of Text, chỉ ra việc kết thúc một frame.

Định nghĩa các frames truyền:

1. Frame truyền cho phép bắt đầu cuộc đua giữa 2 cá ngựa.
Frame này cho phép người dùng bắt đầu cuộc đua giữa 2 cá ngựa bằng việc xác định vị trí xuất phát, và xóa giá trị của 2 mảng về vị trí 0.

Nhận	
STX	0x02
TxRxID	0x01
Length	2
Data	Byte 0: giá trị khởi đầu cho index 1, dải từ 0 đến 64, giá trị khác báo lỗi
	Byte 1: giá trị khởi đầu cho index 2, dải từ 0 đến 64, giá trị khác báo lỗi.
ETX	0x03

Phản hồi	
STX	0x02
TxRxID	0x81
Length	1

Data	Status - 0: Thành công, 0xFF lỗi frame truyền
ETX	0x03

2. Frame cho phép di chuyển 1 trong 2 cá ngựa

Frame này cho phép di chuyển 1 trong 2 cá ngựa với các điều kiện di chuyển được quy định như bên trên

Nhận	
STX	0x02
TxRxID	0x02
Length	3
Data	Byte 0: 1 – di chuyển cá ngựa 1, 2 – di chuyển cá ngựa 2, khác = lỗi. Byte 1: Số bước di chuyển (từ 1 tới 6) Byte 2: Giá trị cộng thêm vào mảng sau khi di chuyển (từ 1 tới 6, giá trị khác báo lỗi), ví dụ cá ngựa 1 đang ở vị trí 1, phần tử số 2 của mảng 1 đang có giá trị là 2 -> khi lệnh di chuyển với data: 1, 1, 5 -> lệnh này di chuyển cá ngựa 1 lên 1 ô, giá trị data cộng thêm là 5 -> giá trị tại ô số 2 tăng lên 5 = 7
ETX	0x03

Phản hồi	
STX	0x02
TxRxID	0x82
Length	1
Data	0: Di chuyển thành công 1: Di chuyển tới đích 2: Không di chuyển được. 0xFF: Frame truyền bị sai
ETX	0x03

3. Lệnh cho phép đọc thông tin 1 trong 2 cá ngựa.

Frame này cho phép kiểm tra vị trí hiện tại của cá ngựa, vị trí bắt đầu, số bước cần di chuyển thêm để về đích.

Nhận	
STX	0x02
TxRxID	0x03
Length	1
Data	Byte 0: 1 -> đọc thông tin cá ngựa 1, 2 đọc thông tin cá ngựa 2, giá trị khác = lỗi.
ETX	0x03

Phản hồi	
STX	0x02
TxRxID	0x83
Length	4
Data	Byte 0: status – 0 = success, 0xFF = lỗi frame truyền.

	Byte 1: vị trí bắt đầu (0 đến 64), byte 1: vị trí hiện tại (0 đến 64), byte 2: số bước cần di chuyển thêm.
	Khi status là lỗi, thì 3 bytes sau là giá trị bất kỳ (bỏ qua).
ETX	0x03

4. Cho phép đọc thông tin của 1 trong 2 mảng sau khi 1 trong 2 cá ngựa đã về đích

Chỉ cho phép đọc thông tin của 1 trong 2 mảng khi cuộc đua kết thúc (1 trong 2 cá ngựa về đích)

Nhận	
STX	0x02
TxRxID	0x04
Length	1
Data	Byte 0: 1 = đọc mảng 1, 2 = đọc mảng 2, giá trị khác = lỗi.
ETX	0x03

Phản hồi	
STX	0x02
TxRxID	0x84
Length	66
Data	Byte 0- status: 0 đọc thành công, 1 đọc lỗi do cuộc đua chưa kết thúc, 0xFF lỗi frame truyền
	Byte 1- byte 65: giá trị của mảng
ETX	0x03