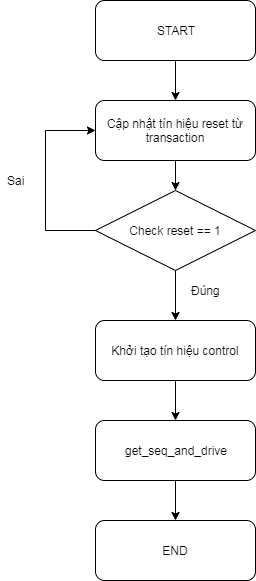
Task run\_phase:

1. Start
2. Cập nhật tín hiệu reset

S0

1. Kiểm tra tín hiệu reset, nếu reset == 0 thì quay lại bước 1, ngược lại thì chuyển tới bước 3
2. Khởi tạo tín hiệu control:

+ cipher\_en <= 0

S1

+ chain\_en <=0

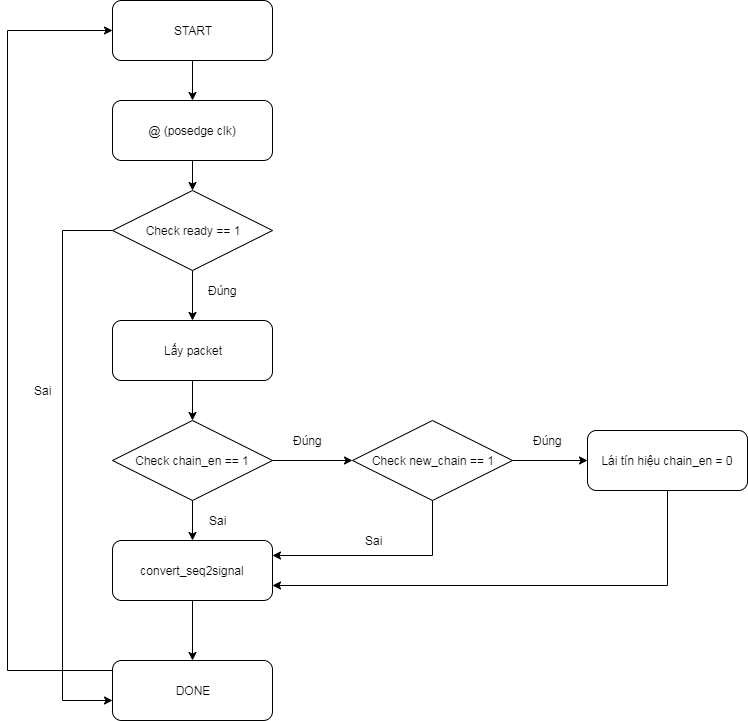
1. Task get\_seq\_and\_drive(sẽ được mô tả bên dưới)
2. END

S3

S2

S4

S5

Task get\_seq\_and\_drive:

1. START
2. Đợi tới cạnh lên của clock tiếp theo

S0

1. Kiểm tra tín hiệu ready từ DUT, nếu bằng 0 thì chuyển tới bước 6 ngược lại thì chuyển tới bước 3.
2. Lấy 1 packet.

S1

1. Kiểm tra tín hiệu chain\_en

+ Nếu chain\_en bằng 0 thì chuyển tới bước 5.

+ Nếu chain\_en bằng 1 thì kiểm tra tín hiệu new\_chain, nếu new\_chain = 0 thì chuyển tới bước 5, ngược lại nếu chain\_en = 1 thì lái tín hiệu chain\_en xuống 0 và chuyển tới bước 5.

S2

1. Task convert\_seq2signal.
2. DONE chuyển sang bước 0.

S3

S4

S5

S6

Task convert\_seq2signal:

1. Start
2. IDLE\_STAGE: Biến int\_delay sẽ được gán từ userTransaction.aes\_blockDelay và từ đó ta sẽ tạo ra những chu kì delay. Vì biến int\_delay có giá trị nhỏ nhất là 1 nên tại IDLE\_STAGE sẽ có ít nhất 1 chu kì.
3. START\_STAGE: ta sẽ bắt đầu lái tín hiệu điều khiển và data theo packet. Ta sẽ lái trong vòng 1 chu kì.
4. Đợi tới chu kì tiếp theo, kết thúc giai đoạn START
5. Bắt đầu giai đoạn BUSY, ta sẽ lái 2 tín hiệu điều khiển cipher\_en, decipher xuống 0.
6. DONE