

IP module을 위한 UART의 VLSI 설계

박성일*, 최병윤
동의대학교 컴퓨터공학과

VLSI design of a UART for IP module

Sung-il PARK*, Byeong-Yoon CHOI.

Dept. of Computer Engineering, Dong-Eui University

E-mail : 02gm104@dongeui.ac.kr*

요 약

본 논문에서는 UART(Universal Asynchronous Receiver-Transmitter)를 soft IP(Intellectual Property) 모듈 형태로써 VLSI 설계과정을 통하여 구현하였다. 이 모듈은 현재 각종 통신 디바이스에서 최하 말단에서 직렬 데이터를 시스템으로 받아들이거나 병렬 데이터를 직렬 라인에 실어 보내는 중요한 역할을 담당한다. 본 연구에서 설계한 UART는 간단한 모듈 형태로 제작되어 있어 Verilog-HDL을 사용하여 직렬 송·수신을 필요로 하는 시스템에 내장되어 사용될 수 있다. 본 논문에서는 설계 순서에 따라 UART를 설계하고 Simulation을 하고 Synopsys Tool을 사용하여 Compile 과 Synthesis 후 Gate Area 와 Delay를 검출해 내었다. 합성결과 0.25 μ m 공정의 CMOS Cell Library를 사용하였을 경우 전체 면적은 1,013 gate가 나왔다. 본 논문에서 설계한 UART의 최장경로가 최대 4.12ns로 나타났으며, 최대 동작 클럭 주파수는 200MHz 로써 150Mbps 이상의 전송 속도를 가진다.

1. 서론

디지털 시스템에서 데이터는 일반적으로 병렬로 처리된 것으로, 처리된 데이터를 다른 시스템에 전달하는 방법은 거리와 비용을 생각하여 결정하여야 하는데 가까운 거리의 경우는 병렬로 전송할 수 있지만 먼 거리에서는 긴 배선에 따라 비용 상승의 문제가 있다. 따라서 먼 거리의 통신의 경우 문제를 해결하기 위해서는 직렬로 데이터를 송신하고 수신하는 방식을 택하게 된다. 이러한 직렬통신은 UART가 Serial Data를 송/수신하게 된다[1-2][6][7].

본 논문에서는 전송속도에 맞추어 병렬데이터를 직렬데이터로 변환하여 전송하고, 직렬데이터를 전송속도에 맞추어 수신하여 병렬데이터로 변환하고 CPU에 전달하도록 하였다.

2. UART 송·수신 개념

직렬 통신에서는 병렬 데이터를 하나의 라인으로

순차적으로 송/수신 하게되는데, 병렬 데이터의 가장 하위 bit를 먼저 보내고 가장 상위 bit까지 보내게 된다. 이때, 같이 실어 보내는 신호로는 Start bit 와 Parity bit, 마지막에는 Stop bit가 있다[2].

그림1.은 비동기 직렬 통신하게 되는 데이터의 포맷이다.

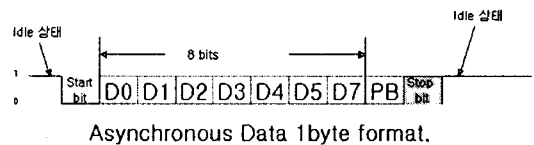


그림1. 비동기 전송시의 데이터 포맷

그림1.에서와 같이 데이터가 존재하지 않는 경우에

는 line에서 idle 상태가 계속 유지가 된다. 이러한 idle 상태에서 start bit를 수신하게 되면 Asynchronous 통신이 시작되고, 뒤를 이어 들어오는 8개의 데이터를 받고, 패리티 비트를 받은 다음에 Stop bit를 끝으로 1 byte의 데이터 수신 싸이클을 끝내게 된다. 이러한 데이터 포맷을 송/수신하기 위한 UART의 구조를 살펴보면, 송신부와 수신부로 나눌 수 있으며 송신부의 경우 그림2. 와 같은 구조를 가진다[2].

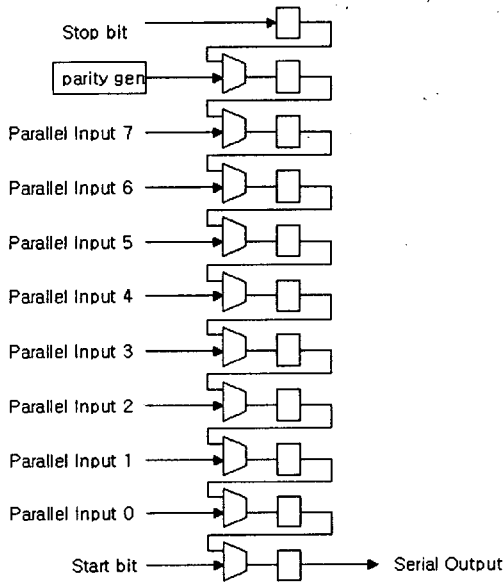


그림2. 송신부의 구조

그림1.에서처럼 데이터 포맷을 만들어 내기 위해서는 그림2.와 같은 구조의 송신부가 필요한데 첫 비트는 Start 비트를 송신 Clock에 맞추어 실어보내게 되고 뒤를 이어 병렬 Data를 첫 번째 비트부터 마지막 비트까지 연속해서 보내게 되며 마지막에 패리티 검사를 통하여 생성된 parity 비트를 실어 보내고 송신 완료 비트인 Stop bit를 실어 보내면 1 Cycle을 끝내게 된다.

수신부의 경우는 그림3.에서와 같은 구조를 통하여 데이터를 수신하여 버퍼에 저장하게 된다[2].

그림4.에서처럼 데이터의 수신은 수신시작을 알리는 Start 신호가 감지 될 때까지 Idle 상태에서 계속 해서 기다리게 된다. 이때 Start 신호가 감지되게 되면 Start 상태가 되고 다음으로 Data bit를 버퍼에 저장하게 된다. Data bit를 모두 버퍼에 넣게 되면 수신되는 Parity bit를 패리티비트 버퍼에 넣게되고 마지막

으로 stop 비트를 감지 후에 다시 Idle 상태에 들어가게 된다.

송신 부에서는 수신 측에서 받아들이는 비트의 역순으로 전송을 하므로 상태도의 방향을 반대로 생각하게 되면 송신 부의 상태도를 얻어 낼 수 있다[2].

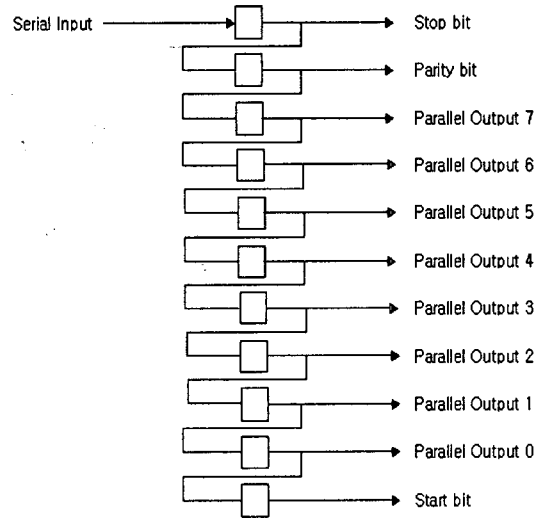


그림3. 수신부의 구조

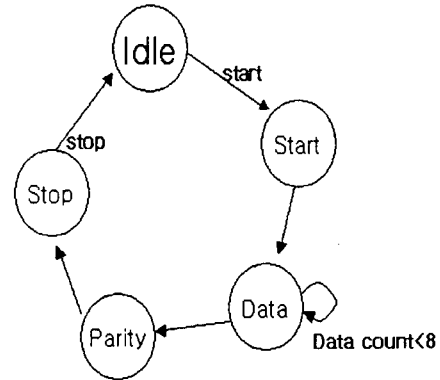


그림4. 데이터 수신상태도

3. 송·수신 클럭 주파수 결정

시스템 Clock을 적절히 변화시켜 Baud 전송 속도에 맞추어 데이터를 실어 보내고 또한 수신측에서도 수신되는 데이터를 적절한 Baud 수신 속도에 맞추어 Sampling 하게 되는데, 이때 필요한 것이 Baud rate를 맞추는 것이다[3][5]. 2,400bps의 데이터를 샘플링

하기 위한 baud rate를 계산하여 보면 다음과 같다.

2,400bps의 경우를 예로 들면 초당 300문자(300byte, 즉 300*8bit = 2,400bit)를 전송하게 된다. 이때 1문자(1byte)를 전송하기 위해 걸리는 시간은 약 3.3ms가 걸리게 된다. ($\frac{1}{300} = 0.0033.. = \text{약 } 3.3\text{ms}$)

문자(Byte)외에도 데이터의 시작을 알리는 Start bit와 에러 검출을 위한 Parity bit, 데이터의 끝을 알리는 Stop bit도 포함되어야하므로 1비트를 전송하는데 걸리는 시간은 ①과 같으며, 송신 Clock의 주파수는 ②와 같다.

$$\frac{0.0033}{11} = 0.0003030\text{초} = 0.30\text{ms} \dots\dots\dots \text{①}$$

$$\frac{1}{0.00030} = 3.3\text{kHz} \dots\dots\dots \text{②}$$

시스템 Clock이 20MHz라고 했을 경우 외부 Clock을 적절히 baud rate에 맞추어 분기시키고 송신 Clock과 수신 Clock을 따로 생성 해주어야 한다.

수신시는 송신시의 Clock의 16배에 해당하는 Clock으로 카운트를 하여 1비트의 송신 Clock 시간에 대해 안전한 Sampling을 마련해 주어야 한다.

실제적으로 외부 Clock을 baud rate로 나누는 부분의 Block은 그림5와 같다. CPU로부터 지정된 baud rate으로 카운트를 통하여 카운트하고 baud clock을 발생시킨다.

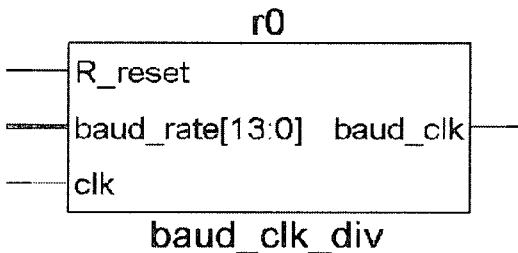


그림5. Baud Clock 발생기

발생된 baud clock은 각각 송신 Clock 발생기와 수신 Clock 발생기로 전달되어 목적에 맞는 Clock을 발생하도록 해야 한다.

외부 Clock을 분주하여 사용하게 되므로 외부 Clock의 주기에 따라서 분주시의 Baud 속도가 달라지게 된다. 하지만, Sampling 시 1비트를 검출해내고, 송신시 1bit를 만들어내는 것은 외부 Clock의 영향을 받지 않는다.

그림6.에서는 수신시 전송되어오는 Start bit에서

falling edge를 감지하고 8 clock 후에 Sampling Clock을 발생하여 start bit를 Sampling 한다[3].

이후, 카운트를 16분주 카운터의 매 8마다 Sampling 하게 되면 정확히 16 clock을 주기로 전송되는 비트의 약 $\frac{1}{2}$ 지점에서 안전하게 Sampling을 하게 된다.

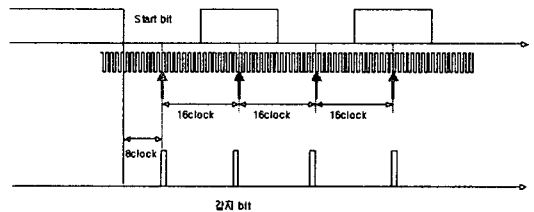


그림6. 데이터 Sampling시의 타이밍

Sampling시 Stop bit를 마지막으로 1 Cycle을 끝내게 된다.

그림6.에서의 falling edge를 감지하는 부분은 그림 7.과 같이 구성된다.

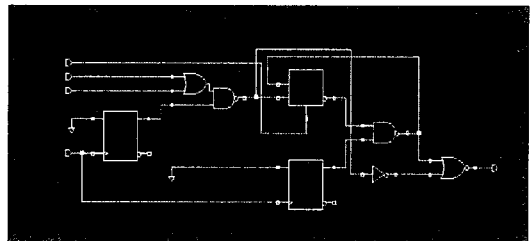


그림7. falling edge 감지부

falling edge는 각 데이터의 시작을 나타내므로 마지막 Stop 비트는 필히 high 상태를 유지하여야 한다.

4. 회로설계

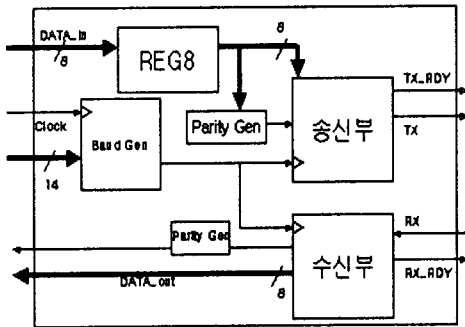
2. 3절의 설계 사양을 바탕으로 UART를 Verilog HDL을 사용하여 설계하였다[4].

각각의 Baud rate에 따른 Baud Count는 표1.에서와 같이 2,400bps에서부터 28,800bps까지 외부에서 조정하도록 하였다.

설계시 외부 Clock의 주파수를 20Mhz로 설정하였으며 그에 따라 이론적으로 Baud rate의 최대 속도는 1.2 Mbps까지로 설정 할 수 있다. 그러나 Synthesis

후 Critical path가 최대 4.12ns로 나타났다. 이는 외부 Clock이 최대 200MHz 정도일 경우에도 동작한다는 결론이 나온다.

UART의 TOP Block은 그림8. 과 같은 구조를 가지고 있으며 설계사양은 표1. 과 같다.



[그림 8] UART block

표 1. UART 특성

UART 특성	
동작 Clock	20MHz
직렬 out(TX)	1 bit
Parallel in(TX)	8 bit
직렬 in(RX)	1 bit
Parallel out(RX)	8 bit
reset	1 bit
RX_ready	1 bit
TX_ready	1 bit
parity bit(수신용)	1 bit
select	0: 송신시 1: 수신시
Baud rate Config. (baud rate / decimal count)	2,400bps / 6060
	4,800bps / 3030
	9,600bps / 1515
	14,400bps / 1010
	19,200bps / 758
	28,800bps / 505

송신과 수신시의 구분을 위해서 select선을 두어CPU 측에서 선택하도록 하였다.

UART의 설계과정은 그림9.와 같은 순서에 의하여 수행되었으며 컴파일 및 Synthesis는 Synopsys Tool을 사용하였다. 0.25µm CMOS Cell Library를 이용하여 delay test및 Synthesis를 하였다.

설계시와 Simulation시 사용된 Verilog-HDL 을 사용하였으며, Synthesis Tool로 Synopsys Tool을 사용하였다.

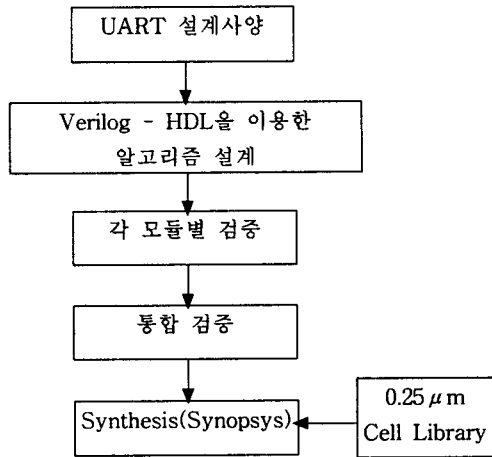


그림9. UART 설계 과정

UART의 Synopsys Tool Synthesis후의 결과 화면은 그림10.와 같다. Synopsys 의 design analyzer Tool을 이용하여 면적을 계산한 결과 표2.의 결과가 나왔다.

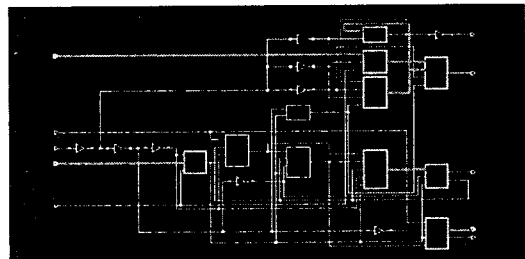


그림10. Synopsys Synthesis결과

표 2. Synopsys design analyzer의 Area report

Area	
Combinational area	414
Noncombinational area	589
Net Interconnect area	9
Total cell area	1,004
Total area	1,013

그림11.은 Signal Scan Tool을 이용하여 UART의 Simulation 시의 Wave form을 나타내고 있다. Test 시에는 48,000bps의 baud rate으로 동작 Simulation 시의 화면이다.

그림11.의 ①은 수신시의 Wave form이며 ②는 송신시의 Wave form 이다.

①에서 RX_data로부터 외부 데이터를 받아 들여서

D_out과 parity_bit_out으로 8bit 데이터와 parity bit 를 받아 내게 된다. ②에서 송신할 데이터는 16(decimal)이며, 이때 TX_DATA를 통하여 start bit 와 data[7:0], parity bit, stop bit 가 송신되는 모습을 볼 수 있다.

5. 결론

설계된 UART는 삼성 0.25 μ m Cell Library를 이용하여 Synthesis 시 1,013 게이트가 나왔으며, 최대 Clock 주파수는 200MHz가 나왔다.

ASIC 설계 과정을 통하여 살펴본 UART의 구현을 통하여 하나의 IP(intellectuals Property) 모듈로써 그 가능성을 엿볼 수 있었으며 작은 면적을 차지하므로 SOC 공정에 있어서 많은 도움이 될 것으로 판단된다. 차기 설계에서 보다 최적화된 구조를 가지기 위해서는 입출력 레지스터를 하나로 통합하여 입력과 출력시의 레지스터 면적을 반으로 줄일 수 있을 것이다.

[참고문헌]

- [1] Eli Sternheim, Rajvir Singh, Rajeev Madhavan ,Yatin Trivedi, "Digital Design and Synthesis with Verilog HDL", EDN, 1993
- [2] 박현철, HDL을 이용한 반도체 회로 설계 한성 출판사, 2001
- [3] P. Vatsolaki, "Design of a High-Speed UART VLSI Library Cell". Technical Report FORTH-ICS/TR-50, June 1992
- [4] Mentor Graphics Web Site <http://www.mentor.com/>
- [5] 박세현, "VHDL에 의한 디지털 컴퓨터 설계와 구현(개정판)"그린출판사, 2000
- [6] O'Neill, B.C.; Clark, S.; Wong, K.L. "Serial communication circuit with optimized skew characteristics", IEEE Communications Letters , Volume: 5 Issue: 6 , June 2001 Page(s): 260 -262
- [7]Yeandel, J.; Thulborn, D.; Jones, S."An on-line testable UART implemented using IFIS" VLSI Test Symposium, 1997., 15th IEEE , 1997 Page(s): 344 -349

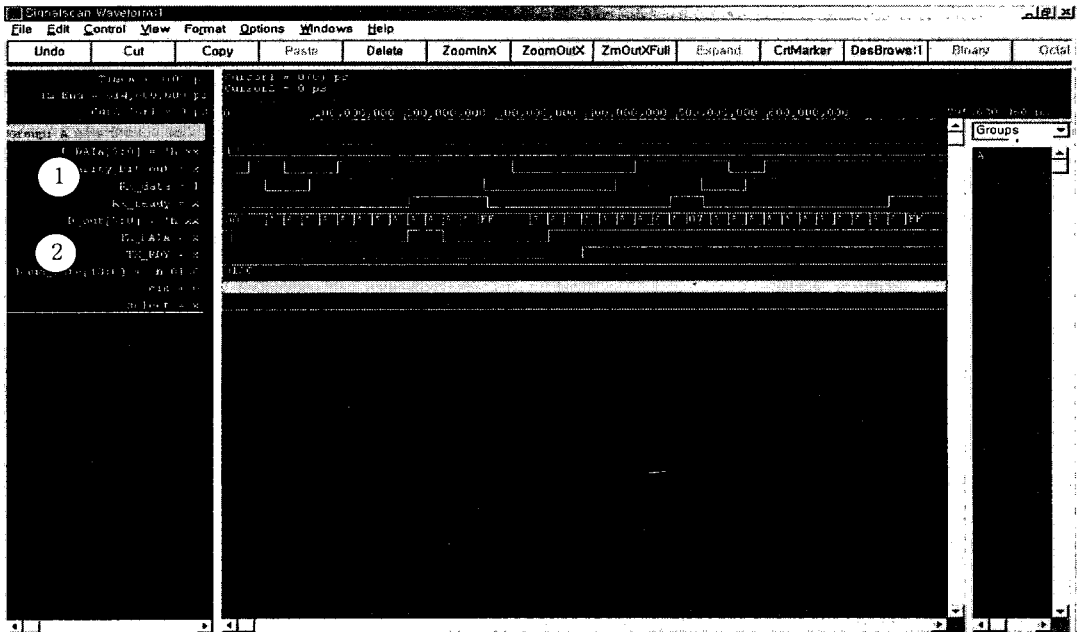


그림11. 결과 파형