

JavaRock 開発日記 (3)

JavaRock は鋭意開発中なわけですが、 のような Java のコードを書いたら、シリアル経由でちゃんとリード/ライトできるハードウェア (HDL) を合成できるようになった。Java 配列を置換している BlockRAM のリードサイクルがずれるバグを発見、修正。演算子も必要に応じて鋭意追加中。構文としては、while を抜けるときに break 使えるようにしたいけど、スコープに応じたステートの管理をもう少しきちんと見直す必要がありそう。

そろそろドキュメントとか使い方とか整備するかな。

```
import net.wasamon.javarock.rt.*;

public class echo implements Synthesizable{
    rs232c obj = new rs232c();
    byte[] data = new byte[128];
    @auto
    public void test(){
        obj.write((byte)'>');
        int i = 0;
        byte c = 0;
        boolean flag = true;
        while(flag){
            c = obj.read();
            if(c == (byte)'#n' || c == (byte)'#r'){
                flag = false;
            }else{
                data[i] = c;
                i++;
            }
        }
        for(int j = 0; j < i; j++){
            c = data[j];
            obj.write(c);
        }
        obj.write((byte)'#n');
    }
}

import net.wasamon.javarock.libraries.*;
import net.wasamon.javarock.rt.*;
public class rs232c implements Synthesizable{
    RS232C_TX tx = new RS232C_TX("66000000", "19200");
    RS232C_RX rx = new RS232C_RX("66000000", "19200");
    public byte read(){
        boolean d_rd = true;
        while(true){
            if(d_rd == false && rx.rd == true){
                return rx.dout;
            }else{
                d_rd = rx.rd;
            }
        }
    }
    public void write(byte data){
        while(tx.ready == false) ;
        tx.din = data;
        tx.wr = true;
        tx.wr = false;
    }
}
```