■ BrainF**k/FPGA

FPGA 上に BrainF**k を書いてみました.
ソースコードは, ↓な感じ.

```
public class BF{

  private final int ARRAYSIZE = 10000;
  private final int CODESIZE  = 10000;

  private IO io = new IO();
  private byte[] prog = new byte[CODESIZE];
  private byte[] data = new byte[ARRAYSIZE];
  private int ptr, pc;

  private void prompt(){
    io.putchar((byte)'>');
    io.putchar((byte)' ');
    return;
  }

  public void read(){
    prompt();
    for(int i = 0 ; i < CODESIZE; i++){
      byte b;
      b = io.getchar();
      if(b == '\n' || b == '\r'){
prog[i] = (byte)0;
break;
      }else{
prog[i] = b;
      }
    }
  }

  public void init(){
    ptr = 0;
    pc = 0;
    for(int i = 0; i < ARRAYSIZE; i++){
      data[i] = 0;
    }
    return;
  }

  public boolean step(){
    byte cmd = prog[pc];
    switch(cmd){
    case 0: return false;
    case '>': ptr++; break;
    case '<': ptr--; break;
    case '+': data[ptr] = (byte)(data[ptr] + 1); break;
    case '-': data[ptr] = (byte)(data[ptr] - 1); break;
    case '.': io.putchar(data[ptr]); break;
    case ',': data[ptr] = io.getchar(); break;
    case '[':
      if(data[ptr] == (byte)0){
  while(true){
    pc++;
    if(prog[pc] == ']'){
      break;
    }
  }
      }
      break;
    case ']':
      if(data[ptr] != (byte)0){
  while(true){
    pc--;
    if(prog[pc] == '['){
      break;
    }
  }
      }
      break;
    default: break;
    }
    pc++;
    return true;
  }
```

```
}
```