

ECE 545 Fall 2009
Exam 1

Introduction & tasks:

The RC5 cipher is specified below using its

- a. pseudocode
- b. interface
- c. table of input/output ports
- d. communication protocol
- e. timing requirements.

Perform the following four phases of the design process for the RC5 cipher implementation:

- 1. Block diagram of the Datapath**
- 2. Interface with the division into the Datapath and the Controller**
- 3. RTL VHDL code of the Datapath (main functionality required, entire functionality for bonus points)**
- 4. Testbench for the entire circuit**

Pseudocode:

The RC5 cipher is defined below:

Input message block M has $2 \cdot w$ bits (where w is a parameter of the cipher).

The corresponding ciphertext block (i.e., encrypted message block) has also $2 \cdot w$ bits.

In order to encrypt a message block M , the algorithm performs the following operations:

Split M into two halves A and B , w bits each

$A = A + S[0]$

$B = B + S[1]$

for $j = 1$ to r do

```
{
  A' = ((A⊕B) <<< B) + S[2j]
  B' = ((B⊕A') <<< A') + S[2j+1]
  A = A'
  B = B'
}
```

$C = A || B$

Notation:

$A, B, A', B' = w$ -bit variables

$S[2j], S[2j+1] =$ a pair of round keys, each round key is a w -bit variable

$\oplus =$ an XOR of two w -bit words

$+$ = unsigned addition mod 2^w

$A \lll B$ = rotation of the variable A by a number of positions given by the current value of the variable B

$A \parallel B$ = concatenation of A and B.

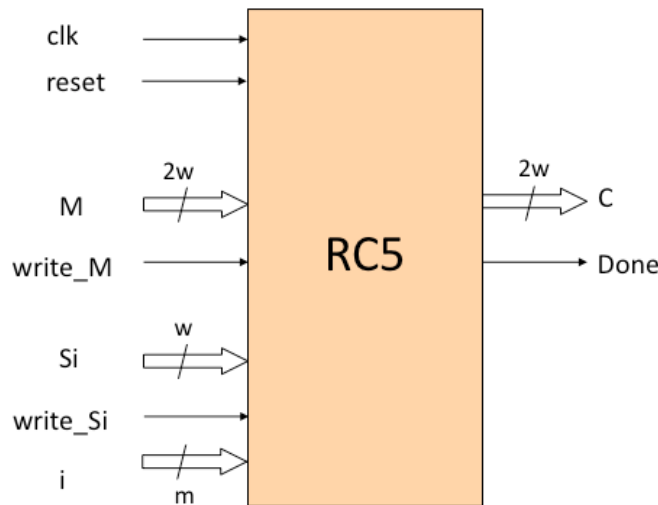
Please note that the algorithms has two parameters:

- r = number of rounds (e.g., 3)
- w = word size (always a power of 2, e.g., $w = 2^4 = 16$)

These parameters should be treated as generics or constants in your VHDL code. Their values are defined at the compilation time.

Interface:

Assume the following interface to your circuit:



| Port | Width | Meaning |
|----------|-------|---|
| clk | 1 | System clock. |
| reset | 1 | System reset – clears internal registers. |
| M | $2w$ | Message block. |
| write_M | 1 | Synchronous write control signal for the message block M. After the block M is written to the RC5 unit, the encryption of M starts automatically. |
| Si | w | Round key $S[i]$ loaded to one of the two internal memories. The first memory stores values of $S[i=2j]$, i.e., only round keys with even indices. The second memory stores values of $S[i=2j+1]$, i.e. only round keys with odd indices. |
| write_Si | 1 | Synchronous write control signal for the round key $S[i]$. |
| i | m | Index of the round key $S[i]$ loaded using input Si. |
| C | $2w$ | Ciphertext block = Encrypted block M. |
| DONE | 1 | Asserted when ciphertext is ready, and available at the output. |

Note:

m is a size of index i. It is a minimum integer, such that $2^m \geq 2r+2$.

Protocol:

An external circuit first loads all round keys
 $S[0], S[1], S[2], \dots, S[2r], [2r+1]$
to the two internal memories of the RC5 unit.

The first memory stores values of $S[i=2j]$, i.e., only round keys with even indices. The second memory stores values of $S[i=2j+1]$, i.e. only round keys with odd indices.

Loading round keys is performed using inputs: $S_i, i, \text{write_}S_i, \text{clk}$.

Then, the external circuits, loads a message block M to the RC5 unit, using inputs: $M, \text{write_}M, \text{clk}$.

After the message block M is loaded to the RC5 unit, the encryption starts automatically.

When the encryption is completed, signal Done becomes active, and the output C changes to the new value of the ciphertext.

Please assume that the output C keeps the last value of the ciphertext at the output, until the next encryption is completed. Before the first encryption is completed, this output should be equal to zero.

Tasks & Assumptions:

Task 1: Draw block diagram of the Datapath

Assume that

- one round of the main for loop of the pseudocode executes in one clock cycle
- you can access only one position of each internal memory of round keys per clock cycle

As a result, the entire encryption of a single message block M should last $r+1$ clock cycles.

Task 2: Draw an Interface of the RC5 unit with the division into the Datapath and the Controller

Task 3: Write the RTL VHDL code corresponding at least to the datapath of the main loop of the pseudocode (without memories of round keys, and without support for all signals of the interface)

Assume that:

- values of parameters w, r, m are specified as constants or generics

Task 4: Testbench for the entire circuit

Write a simple testbench for the entire RC5 unit, following specified above interface, assuming that:

- $r=2, m=3, w=16$
- $M = 0x0000ABCD$ (in hex)
- $S[0]=0xB134, S[1] = 0x5634$
 $S[2]=0x5347, S[3] = 0x4728$
 $S[4]=0x4923, S[5] = 0x90BC$ (all values in hex)