

Introduction to Computer Engineering **Arm Processor**

Kevin Truong

May 14, 2018

1. ARM Processor

1.1 Program Counter

The Program Counter contains the address of the instruction to be executed.

```
module pc (  
    input wire clock,  
    input wire [31:0] in,  
    output wire [31:0] out  
);  
  
// On each clock edge, store in input 'in' into register,  
// unless hlt is 1, in which case don't store the input.  
// Assign the register to be output via 'out' port\  
  
reg [31:0] temp;  
  
always_ff @ (posedge clock) begin  
    temp = in;  
  
end  
    assign out = temp;  
endmodule
```

The program counter is usually incremented by four because each instruction is 32 bits. When the clock edge is 1, the input address becomes the output address.

1.2 Branch

The branch instruction adds PC+8 with a 24-bit immediate. RegSrc is 1 so that the multiplexer selects R15, which contains the program counter. The immediate is extended by four and extended and added to R15. The Aluresult is fed into the PC, so that the program counter is changed.

2. Extra Functionality

2.1 Carry Lookahead Adder

I implemented the carry-lookahead adder. I generated the propagate and generate using a for loop in C.

```
assign p[0] = a[0] | b[0];  
assign g[0] = a[0] & b[0];  
assign p[1] = a[1] | b[1];  
assign g[1] = a[1] & b[1];
```

I used the propagate and generate values to find the carry values.

```
assign carry1[1] = g[0] | (p[0] & carry1[0]);  
assign carry1[2] = g[1] | (p[1] & carry1[1]);  
assign carry1[3] = g[2] | (p[2] & carry1[2]);
```

My ripple carry adder works also.

2.2 Testing

I tested my design with the given testcases.