Processors, FPGAs, and ASICs

Prof. Stephen A. Edwards
sedwards@cs.columbia.edu

NCTU, Summer 2005

Spectrum of IC choices

You choose

<table>
<thead>
<tr>
<th>Choice</th>
<th>Flexibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full Custom</td>
<td>You choose</td>
</tr>
<tr>
<td>ASIC</td>
<td>polygons (Intel)</td>
</tr>
<tr>
<td>Gate Array</td>
<td>circuit (Sony)</td>
</tr>
<tr>
<td>FPGA</td>
<td>wires</td>
</tr>
<tr>
<td>PLD</td>
<td>logic network</td>
</tr>
<tr>
<td>GP Processor</td>
<td>logic function</td>
</tr>
<tr>
<td>SP Processor</td>
<td>program (e.g., Pentium)</td>
</tr>
<tr>
<td>Multifunction</td>
<td>program (e.g., DSP)</td>
</tr>
<tr>
<td>Fixed-function</td>
<td>settings (e.g., Ethernet)</td>
</tr>
<tr>
<td></td>
<td>part number (e.g., 74LS00)</td>
</tr>
</tbody>
</table>

NMOS Transistor Cross Section

Inverter Transistors and Layout

NAND Gate Transistors and Layout

Full-custom ICs

Standard Cell ASICs

Standard Cell ASICs

Channeled Gate Arrays
FPGA Families: Spartan-IIE:

Architectural Description

The Spartan-IIE array, shown in Figure 1, consists of a configuration of blocks called a CLB (Configurable Logic Block) slice. A CLB slice contains six look-up tables (LUTs), two flip-flops, and a multiplexer. The LUTs can be configured as either 4-input LUTs or 6-input LUTs, depending on the design requirements.

IOBs provide the interface between the package pins and the internal logic. Each IOB has access to a clock signal (CLK) shared by the three registers and independent Clock Enable (CE) signals for each register. These IOBs also support a wide variety of I/O signaling standards. These high-speed inputs and outputs are capable of supporting the required reference, output, and termination voltages needed to meet the standard.

Input/Output Block

The Spartan-IIE IOB, as seen in Figure 2, features inputs and outputs that support a wide variety of I/O signaling standards. These high-speed inputs and outputs are capable of supporting the required reference, output, and termination voltages needed to meet the standard.

Versatile multi-level interconnect structure

As can be seen in Figure 1, the CLBs form the central logic structure with easy access to all support and routing structures. The IOBs are located around all the logic and memory elements for easy and quick routing of signals on and off a clock signal.

PLAs/CPLDs: The 22v10

Similarly, the F6 multiplexer combines the outputs of all four F5-multiplexer outputs. This permits the implementation of any 6-input function, an 8:1 multiplexer, or selected functions of up to nine inputs.

Look-UpTable

This combination provides either a function generator that can implement any 5-input function, a 4:1 multiplexer, or selected functions of up to nine inputs.

Carry and Control Logic

Functionally, the F6 multiplexer outputs are combined to provide any 6-input function, an 8:1 multiplexer, or selected functions of up to nine inputs.

Example: Euclid's Algorithm

int gcd(int m, int n)
{
    int r;
    while ((r = m % n) != 0) {
        m = n;
        n = r;
    }
    return n;
}

gcd: pushl %ebp
    movl %ebp,%esp
    movl %ebx,%esi
    movl %eax,%edi
    movl %ecx,%ebp
    movl %edx,%esp
    movl %edi,%edi
    movl %esi,%esi
    movl %eax,%eax
    movl %ebx,%ebx
    movl %ecx,%ecx
    leavel
    ret

I386 Programmer's Model

<table>
<thead>
<tr>
<th>31</th>
<th>0</th>
<th>15</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>eax</td>
<td>Mostly</td>
<td>cs</td>
<td>Code segment</td>
</tr>
<tr>
<td>ebx</td>
<td>General-Purpose</td>
<td>ds</td>
<td>Data segment</td>
</tr>
<tr>
<td>ecx</td>
<td>Purpose</td>
<td>ss</td>
<td>Stack segment</td>
</tr>
<tr>
<td>edx</td>
<td>Registers</td>
<td>es</td>
<td>Extra segment</td>
</tr>
<tr>
<td>esi</td>
<td>Source index</td>
<td>ts</td>
<td>Data segment</td>
</tr>
<tr>
<td>edi</td>
<td>Destination index</td>
<td>gs</td>
<td>Data segment</td>
</tr>
<tr>
<td>ebp</td>
<td>Base pointer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>esp</td>
<td>Stack pointer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eflags</td>
<td>Status word</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eip</td>
<td>Instruction Pointer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Euclid on the 386

gcd: pushl %ebp
    movl %esp,%ebp
    movl %ebp,%esp
    movl %ebx,%esi
    movl %eax,%edi
    movl %ecx,%ebp
    movl %edx,%esp
    movl %edi,%edi
    movl %esi,%esi
    movl %eax,%eax
    movl %ebx,%ebx
    movl %ecx,%ecx
    movl %edx,%edx
    movl %eax,%eax
    movl %ebx,%ebx
    leavel
    ret
The output registers of the calling procedure become the inputs to the called procedure.
The global registers remain unchanged. The local registers are not visible across procedures.

```
gcd:  
save $sp, -112, $sp  
move $t0, $t1  
.LL3:  
move $t0, $t1  
call rem, 0  
move $t0, $t0  
cmp $t0, 0  
bne .LL5  
mov $t0, $t1  
ret  
restore
```
1.0 Introduction

1.1 General Description:
The AX88796 provides industrial standard NE2000 registers level compatible instruction set. Various drivers are easy to port and can be easily reused. No much additional effort to be paid. Software is easily port to various embedded systems with no pain and tears.

- AX88796 use 128-pin LQFP low profile package, 25MHz operation, and single 3.3V operation with 5V I/O tolerance.

- The AX88796 Fast Ethernet Controller is a high performance and highly integrated local CPU bus Ethernet Controller with embedded 10/100Mbps PHY/Transceiver and 8K*16 bit SRAM. The AX88796 supports both 8 bit and 16 bit local CPU interfaces include MCS-51 series, 80186 series, MC68K series CPU and ISA bus. The AX88796 implements both Local Bus Fast Ethernet Controller and Ethernet Controller Registers.

- In comparison with AX88195:
  1) Embedded packet buffer memory
  2) Built-in 10/100Mbps PHY/Transceiver
  3) Replace memory I/F with PHY/Transceiver I/F
  4) Canceling SAX address decoding
  5) Fix interrupt status can ’t always clean up problem of AX88195.
  6) Add upto 3/1 general Purpose In/Out pins.

- The main difference between AX88796 and AX88195 are:
  - program: IRD1
  - memory: IRO1
  - control: IC1
  - serial: IS2
  - parallel: IP1
  - SPI: IB1
  - parallel: IP2
  - SPI: IB2
  - serial: IS1
  - program: IRD2
  - I/O: IO1
  - memory: IRO2
  - I/O: IO2
  - control: IC2
  - serial: IS3
  - program: IRD3
  - memory: IRO3
  - control: IC3
  - serial: IS4
  - program: IRD4
  - memory: IRO4
  - control: IC4
  - serial: IS5
  - program: IRD5
  - memory: IRO5
  - control: IC5
  - serial: IS6
  - program: IRD6
  - memory: IRO6
  - control: IC6
  - serial: IS7
  - program: IRD7
  - memory: IRO7
  - control: IC7
  - serial: IS8

- The AX88796 Ethernet Controller Registers:

- **PAGE 0 (PS1=0,PS0=0)**:
  - Command Register
  - Page Start Register
  - Page Stop Register
  - Interrupt Status Register
  - Transmitter Status Register
  - Receive Status Register
  - Boundary Pointer
  - Remote Start Address Register 0
  - Remote Start Address Register 1
  - Remote DMA Address Register 0
  - Remote DMA Address Register 1
  - Current Page Register
  - Number of Collisions Register

- **PAGE 1 (PS1=1,PS0=0)**:
  - Interrupt Mask Register
  - CRC Errors
  - IFGS2
  - MII/EEPROM Access
  - Test Register

- **PAGE 2 (PS1=1,PS0=1)**:
  - Offset Read
  - Offset Write
  - Offset Read

- **PAGE 3 (PS1=1,PS0=1)**:
  - Offset Write
  - Offset Read
  - Offset Write
  - Offset Read

- **PAGE 4 (PS1=1,PS0=1)**:
  - Offset Read
  - Offset Write
  - Offset Read
  - Offset Write

- **PAGE 5 (PS1=1,PS0=1)**:
  - Offset Write
  - Offset Read
  - Offset Write
  - Offset Read

- **PAGE 6 (PS1=1,PS0=1)**:
  - Offset Read
  - Offset Write
  - Offset Read
  - Offset Write

- **PAGE 7 (PS1=1,PS0=1)**:
  - Offset Write
  - Offset Read
  - Offset Write
  - Offset Read

---

**Philips SAA7114H Video Decoder**

**SAA7114H Registers, page 1 of 7 (!)**

**Fixed-function: The 7400 series**

- 7400: Quad NAND Gate
- 74374: Octal D Flip-Flop