Digital Signal Processor architectures (DSPs)
Reconfigurable processing devices (FPGA)

Lecture 5
9, March, 2016.

Typical Digital Signal Processing System

Remarks: The diagram shows the sampling, processing, and reconstruction of an analog signal. There are applications where processing steps at the digital signal processor, e.g., speech recognition.

Analog Signal Processing
- a part of analog control process

Analog incoming sensor signals (signals of a process) - Analog signal processing unit - Analog output signal for actuators

- Process control
- Typical tasks
  - Filters
  - Multipliers
  - Integrators
  - Differentiators
  - PID controllers
- Discrete analog components (tuned with screw driver)

What Do DSP Processors Need to Do Well?
Most DSP tasks require:
- Repetitive numeric computations
- Attention to numeric fidelity
- High memory bandwidth, mostly via array accesses
- Real-time processing

Processors must perform these tasks efficiently while minimizing:
- Cost
- Power
- Memory use
- Development time

FIR Filtering: A Motivating Problem

Each tap (M+1 taps total) nominally requires:
- Two data fetches
- Multiply
- Accumulate
- Memory write-back to update delay line

FIR Filter on Von Neumann Architecture

loop:
mov *n0,x0
mov *r1,x1
mov x0,y0,a
add a,b
mov y0,*r2
inc r0
inc r1
inc r2
dec ctr
tst ctr
jnz loop

Problems: Bus / memory bandwidth bottleneck, control code overhead
TMS32010 FIR Filter Code

Here X4, H4, etc. are direct (absolute) memory addresses:

LT X4 ; Load T with X(n-4)
MFY H4 ; P = H4*X4
LTD X3 ; Load T with X(n-3); X(n-4) = X(n-3);
            ; Acc = Acc + P
MFV H3 ; P = H3*X3
LTD X2
MFV H2

etc.

* Two instructions per tap, but requires unrolling
* One instruction per tap in the TMS320C25

Features Common to Most DSP Processors

* Data path configured for DSP
* Specialized instruction set
* Multiple memory banks and buses
* Specialized addressing modes
* Specialized execution control
* Specialized peripherals for DSP

Memory Architecture

DSP Processor

* Harvard architecture
* 2-4 memory accesses/cycle

General-Purpose Processor

* Von Neumann architecture
* Typically 1 access/cycle

Specialized Peripherals for DSPs

* Synchronous serial ports
* Parallel ports
* Timers
* On-chip A/D, D/A converters
* Video and audio I/O (analog and/or digital)
* Watchdog
* On-chip peripherals often designed for “background” operation, even when core is powered down.

DSP Tasks

* Speech and audio compression
* Filtering
* Modulation and demodulation
* Error correction coding and decoding
* Servo control
* Audio processing (e.g., surround sound, noise reduction, equalization, sample rate conversion, echo cancellation)
* Asynchronous Transfer Mode (ATM) coding, decoding
* Speech recognition
* Signal synthesis (music, speech, texture, video)
* Image and video processing
DSP manufacturer

- Texas Instruments
  - TMS320 DSP family
- Analog Devices
  - SHARC
- Freescale (Motorola)
  - Freescale

Texas TMS320C DSP family

<table>
<thead>
<tr>
<th>type</th>
<th>fix/ floating</th>
<th>features</th>
<th>status</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMS320C1x</td>
<td>Fix point</td>
<td>First DSP</td>
<td>military only</td>
</tr>
<tr>
<td>TMS320C2x</td>
<td>Fix point</td>
<td>First high volume DSP</td>
<td>live</td>
</tr>
<tr>
<td>TMS320C3x</td>
<td>Floating point</td>
<td>First floating point DSP, DMA</td>
<td>live</td>
</tr>
<tr>
<td>TMS320C4x</td>
<td>Floating point</td>
<td>4 communication ports towards other DSPs</td>
<td>obsolete</td>
</tr>
<tr>
<td>TMS320C5x</td>
<td>Floating point</td>
<td>First static CMOS design, low power</td>
<td>live</td>
</tr>
<tr>
<td>TMS320C6x</td>
<td>Fix-floating</td>
<td>Latest modern DSP family: Ultra wide world instructions (256 bit)</td>
<td>live</td>
</tr>
<tr>
<td>TMS320C8x</td>
<td>Floating point</td>
<td>First multimedia processor: 4 pieces of TMS320C4x + RISC (too expensive, not sold in large quantities)</td>
<td>military only</td>
</tr>
</tbody>
</table>

Parameters of the TMS320C66x

- 1GHz – 1.5GHz Clock Rate
- Power consumption @1GHz ~ 1.5W
- 841-Pin Ball Grid Array
- On-chip L2 cache: 1 - 8.8MB
- On-chip 4 MB SRAM
- Number of DSP cores: 1-8

Architecture of the C66x I: Register banks

- Two register banks (A,B)
- 32 pieces of 32bit registers in each bank
- Three register types
  - Condition registers (3 / register bank)
  - Circular address registers (c/d/register bank)
  - Data registers

Architecture of the C66x II: Processor units

- L1, L2 units
  - Logic and arithmetic unit
  - 32/40-bit arithmetic and compare ops
  - 32-bit logical operations
  - Byte shifts
  - Dual / quad arithmetic operations
  - Dual / quad min/max operations
  - Quad 8-bit subtract with absolute value
- S1, S2 units
  - Arithmetic, comparisons, branches
  - 32-bit arithmetic and logical operations
  - 32/40-bit shifts
  - Branches
  - Dual and quad compare operations
  - Dual and quad saturated arithmetic operations
- M1, M2 units
  - Multiply add (MAC) unit
  - 16 pieces of 16x16 bit fixed MAC
  - Each 4 can be Reconfigure to one single precision fixed MAC
- D1, D2 units
  - Address arithmetic unit
  - 32-bit add, subtract, linear and circular address calculation
  - Loads and stores single and double words with constant offset
  - 32-bit logical operations
  - Dual 16-bit arithmetic operation
**Instruction bus**

- 32 bit instruction bus for each processor unit→256 bit wide instruction bus
- One instruction queue, no individual branches for the units
- A regular C code uses 2-3 units at a time
- Hand optimized assembly codes uses 6-8 units parallel
  - Convolution, FFT, matrix and vector operations etc.
  - Optimized function library

**Cache memories**

- Separate data and program paths
- L2 cache configurable

**Memory access speed increase with cache memory**

Fast processor units need fast data access.

High-speed memories (SRAMs) occupy large silicon space (expensive silicon real estate), therefore small blocks can be built.
What happens if the data is not in the cache?

- If it is in L1, it is transferred to the register in the same cycle.
- If it is not in L1 but in L2, then it will be transferred to L1 in 6-8 data cycles. The execution (in each processor units) is halted for this time.
- If more than one data is missing from the same data block, then it requires 6-8 cycles as well, because the data is moved in blocks.
- If the data is not in L2 a DMA transfer is needed from the external memory, which takes hundreds of cycles. All the processor units are halted for this time.

C6000 DSP family

- DSP & DSP+ARM
  - Single DSP
  - DaVinci video processor
- Multi-core DSP
  - Keystone

DaVinci video and media processors

- DSP + RISC + special purpose processors
  - ti 64x DSP (general computing)
  - ARM processor (communication)
  - Front End video processing unit
    - Analog Video signal interface (CCD)
    - Digital (HD) video interface
    - Histogram, resize, preview
  - Back End video processing unit
    - Special purpose digital video processors
    - Analog/digital video output, overlay (OSD)
  - Video compression unit
    - MPEG, H.264

TMS320DM6446 (Da Vinci)

Keystone:
TMS320C6678
Multicore DSP

- 8 processor core
- 1.25GHz
- 512kb L2/cores
- 4Mbyte shared cache
- 841 pin BGA

Reconfigurable processor architectures
FPGA
FPGA

Logic circuits are not completed in the fabrication time
- Unconfigured logic, arithmetic, storing, and connection circuits are provided
- Final configuration is made in the development time, and it might even changes during run-time
- Special purpose execution units can be built optimized for the given task
- Optimal on what level?
  - Implemented logic circuit level:
    - Maximally, depends only on the invested effort (developer time and coffee)
  - On the electrical circuit level:
    - Significant redundancies and unused circuit components (this is the price of the reconfigurability)

Building blocks of the modern FPGAs

- IO blocks
- CLBs
- Interconnections
- Memory blocks
- DSP Slices
- Processors
  - Soft
  - Hard
- High speed serial interfaces
  - Transceivers
    - Ethernet, PCI express, SATA, USB
- Mixed signal modules
  - AD converter
- IP blocks

Basic FPGA architecture

- Three basic building blocks:
  - I/O cells on the periphery (50-1200)
  - Reconfigurable interconnection network
  - Configurable logic blocks (CLB)

Configurable logic blocks

- 2 Four-input function generators (Look Up Tables)
- 1 Three-input function
- 2 Registers:
  - Pos. or Neg. edge trig.
  - Synchronous and asynch. Set/Reset
- Possible functions:
  - any fct of 5 var.
  - two fcts of 4 var, + one fct of 3 var.
  - some fct of 9 var.

(Most modern FPGA applies 6 bit Func. Gen. Block (LUT). However, their architecture is not released by the manufacturers.)

Programmable connection types

3 types:
- Fast Direct Connections
- General Purpose Connections with Switching Matrix
- Horizontal/Vertical Long Lines

Types of lines:
- Single length (6)
- Double length (6)
- Global lines (6)

IO blocks, CLBs

- IO blocks
  - Single ended
  - Differential
  - 1.866 Mb/s (DDR3 memory support)
  - Up to external 200 pins
- CLB (6 input LUT + FF)
  - Up to 2 million
Memory blocks

- 36 kbit memory blocks
- 20-2360 pieces
  - max: 9.44 Mbyte
- Can be configured as FIFO 638MHz
- Can be found in every modern FPGAs

DSP slices

- Multiple-add unit (MAC)
- 638 MHz
- Xilinx:
  - 25x18 bit
- From a few dozens to thousands

Chip with Columns of Embedded Multipliers and RAM Blocks

- RAM blocks
- Multipliers
- Logic blocks

Figure 4-11

Embedded microprocessors

- Hard
  - One or more processors are physically implemented
  - Dual ARM (925MHz)
    - Zynq (Xilinx)
    - Cyclon V SOC (Altera)
- Soft
  - Implemented in the reconfigurable fabric in development time
    - Xilinx: MicroBlaze, PikoBlaze
    - Altera: Nios
  - Configurable components
    - max: 210MHz

The soft cores are significantly weaker than hard core processors. Dual ARMs are quite powerful, but still weaker than a TI C6x DSP. However, the entire system can be more powerful, as the computational heavy parts are implemented in the optimized special purpose processors implemented in the FPGA fabric.

Configured for Performance (MicroBlaze v5.00)

<table>
<thead>
<tr>
<th>FPGA</th>
<th>Size</th>
<th>Clock Frequency</th>
<th>Dhrystone 2.1</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtex-5 (VLX300-1)</td>
<td>960 LUTs</td>
<td>210 MHz</td>
<td>240 DMIPS</td>
<td>1.15 DMIPS/MHz</td>
</tr>
<tr>
<td>Virtex-4 (VLX25-12)</td>
<td>1,709 LUTs</td>
<td>160 MHz</td>
<td>184 DMIPS</td>
<td>1.15 DMIPS/MHz</td>
</tr>
</tbody>
</table>

MicroBlaze Settings

- Base Shifters: ☐
- ROM Multiplier: ☐
- Cache Unit: ☐
- RAM: ☐
- HDL Exception: ☐
- JTAG: ☐
- Debug Logic: ☐

Performance Optimized Subsystem

- Local Instruction Memory: Local Data Memory
- UART: ☐
- GPIO, Timer: ☐

Configuration Options

- All options are selected except for ROM, HDL exception, and debug logic.
- Processor can be configured faster by selecting only the base shifters, ROM, and base instruction memory.
Mixed signal module
- 2 pieces of AD converters
  - 17 differential inputs
  - 1 MSPS

Serial interfaces
- Transceivers
  - Max 28Gbit/sec serial lines
  - Max 96 db
- Ethernet interface
  - 10/100/1000 Mbit/sec
- PCI Express
  - 4 db

JTAG chain
- One can stop the clock and read or write (set) the I/O pins

Programming the FPGAs
- Program languages
  - HDL (Verilog, VHDL) (most common)
  - C (Vivado, Xilinx)
- Compiler
- Partitioning tool
- Wiring tool

Steps of the FPGA programming
- HDL or C code writing
- Synthesis
  - Result is the Netlist, which is an equivalent logic description with the original code
    - Based on CLBs
- Mapping
  - Placement of CLB (physical location in the FPGA fabric)
- Packing
  - Configuring CLBs via filling out LUTs
- Routing
  - Wiring the CLBs

Simple HDL-based FPGA Flow
Chip scope

- Virtual logic analyzer
- Embedded in the compilation time (~10% extra logic)

Technology – Hierarchic Packaging

FPGA families at Xilinx

- Series 6
  - Spartan 6 series (Low cost economic)
  - Virtex series 5-6 (high-end)
  - LX (strong in logic circuits)
  - SX (strong in DSP slices and memory)
  - FX (embedded Power PC and Rocket IIO)
- Series 7
  - ARITIX-7 (Low cost economic)
  - KINTEX-7 (best price/performance)
  - VIRTEX-7 (high-end)
- Zinq
  - Dual ARM

IP* blocks

- IP blocks with different functionalities can be bought
  - DRAM interface
  - SATA interface
  - Special purpose processors (MPEG, JPEG, etc)
  - Data format converters
  - Communication blocks
  - etc
- Soft or Hard macro
  - Soft: no placing and routing
  - Hard: fix placing and routing

Reconfigurabilty in run-time

- Programming a large FPGA may requires tens of seconds
- Traditionally, the LUTs and the routing is constant in run-time
- Latest models enable partial reconfiguration in run-time

Trend: Embedded system in an FPGA
**FPGA roadmap**

- Similarly to microprocessors, not the clock speed but the complexity is increasing with the scaling down
  - Increasing the number of CLBs, memory blocks, serial channels, DSP slices,
  - New functional blocks (e.g.: AD, ARM)
  - Significant reduction of power consumption

**FPGA-DSP coproduction**

- In which cases a DSP is needed next to the FPGA?
  - When large computational tasks are not implementable efficiently in FPGA.
    - Both regular/structured (FPGA) and irregular (DSP) computational loads are relevant
  - e.g.: Smart camera
    - The regular parts are implemented in FPGAs, the others are in the DSP

**Why to use FPGA?**

- More economical than designing a chip (ASIC)
  - Both in time and in costs
- More efficient for structured computations than DSP
- Low power consumption and high computational power
- Entire embedded system can be built
  - Reduces chip count
  - Reduces circuit size and cost

**Which one to use?**

**FPGA**

- Existing CLBs are filled up with program
- Large overhead due to the reconfigurability
- Significantly shorter development time
- Significantly smaller prototype cost
- Similar design process

**ASIC**

- Full chip designed by us
- All the components are entirely free and free to place (CAD)
- No reconfigurability, no overhead, but if it does not work, or small changes are needed, than redesign and re-fabrication is needed (3-5 months, $5-$50k)
- High development cost, but very low fabrication costs (per unit)

**Positioning FPGAs**

- For a given computational task, it is cheaper than DSP for large volume
- The development time/cost of the FPGA implementation is significantly larger than for a DSP, but significantly cheaper than for ASICs

**Price problem**

- A mobile phone, GPS, ADSL or cable modem, digital camera, MP3 player, DVD player requires huge computational power, what can be provided by a $50 FPGA
  - However, the electronics of these devices should be covered from $5- $50 .
- What to do?
  - FPGA → Structured ASIC → ASIC migration
  - $50 /pcs $100 /pcs $2 /pcs  
  - >10,000 pcs/year >300,000 pcs/year
Price and power consumption reduction in the ASIC migration steps

- **RTL (Pre-synthesis)**
- **Gate (Post-IPQ)**
- **Gate (Post-layout)**

**Level of Design Abstraction**

- 1x
- ~10x
- ~100x

**Relative cost of timing analysis and debug**

- RTI
- Gate (Post-IPQ)
- Gate (Post-layout)

Cost of Timing Analysis and Debug

Example: Given a complex computational task

1. solution:
   - 3 pcs of DSP implementation
     - Fabrication cost of 1 pc: $150
     - Development costs: $30,000
2. solution:
   - FPGA implementation
     - Fabrication cost: $30
     - Development costs: $150,000
3. solution:
   - Custom-designed chip (ASIC)
     - Fabrication cost of 52 (above 50,000 pcs/year)
     - Development costs: $2,000,000