# FPGA Evolution for SDR

**Ian Scott**
**ZL4NJ**

arb.xfm@gmail.com

1 February 2017

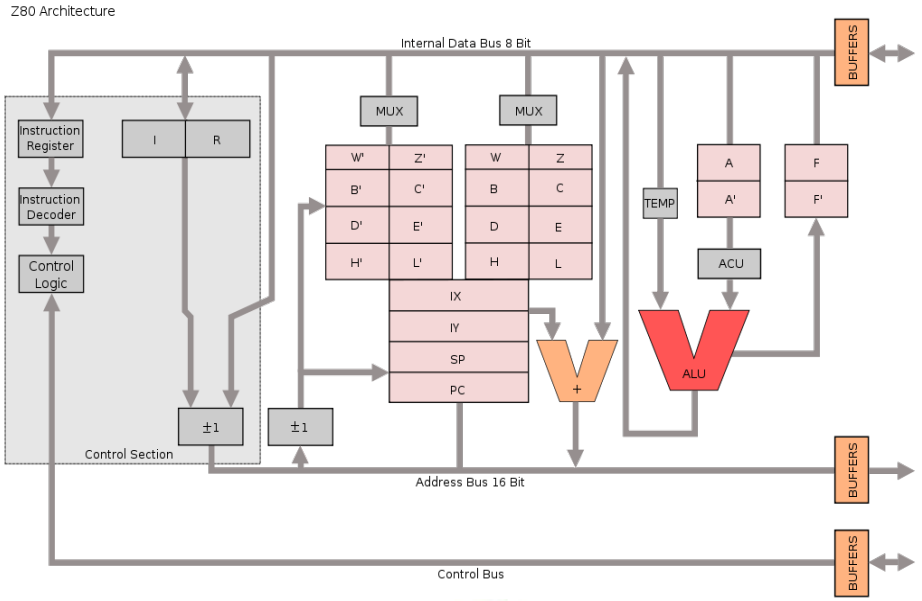
# FPGA Evolution for SDR

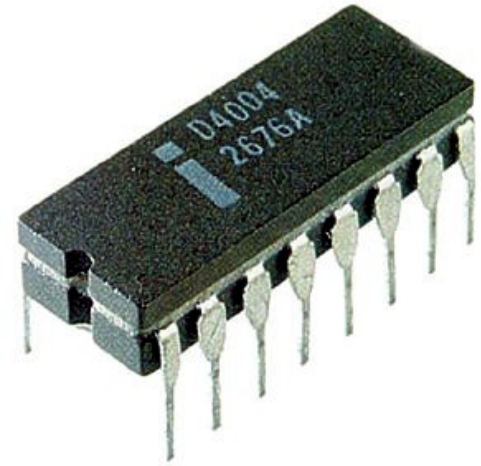
## Three Chapters

- **Chapter 1 – Evolution from Microprocessor to FPGA**
- **Chapter 2 – Examples of FPGA Structures**
- **Chapter 3 – High Level VHDL Code Examples**

# FPGA Evolution for SDR

- **Chapter 1 – Evolution from uP to FPGA**

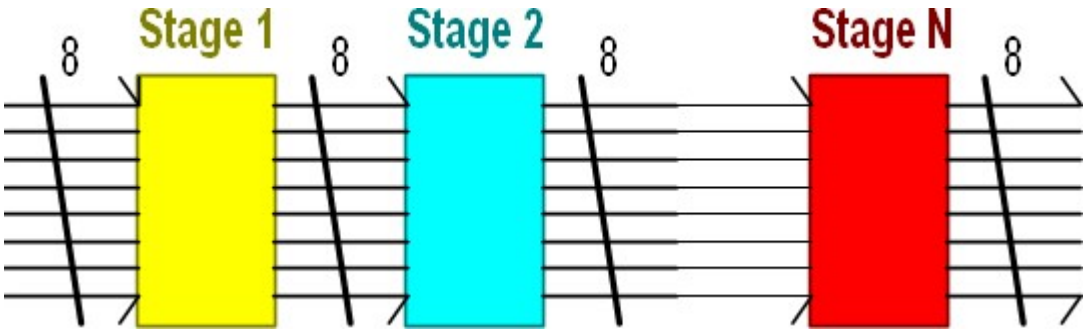


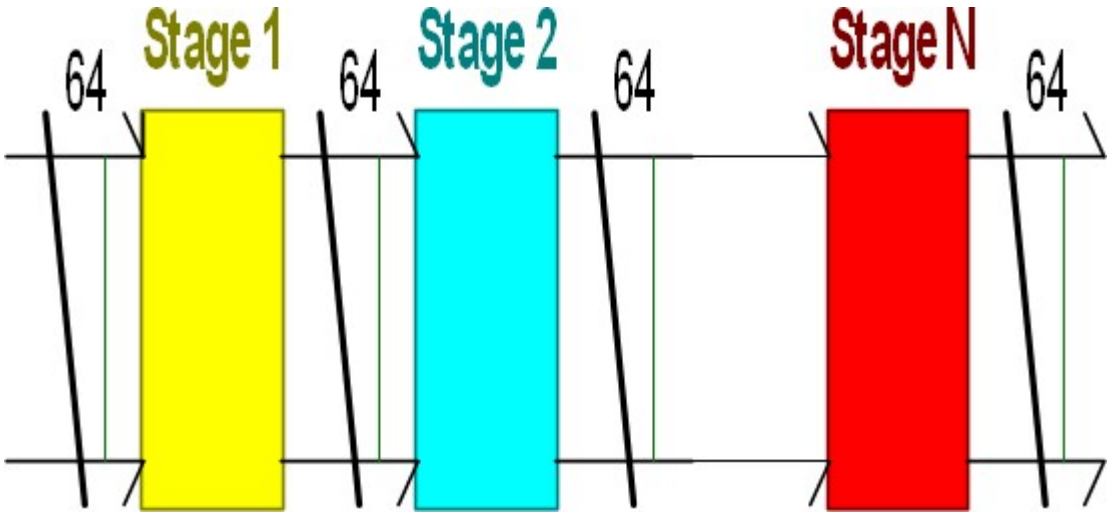




- First Intel Microprocessor (4004) developed by Stanley Mazo and Busicom engineer Masatoshi Shima, ~1969
- Later Z80 uP shown below
- All uP operate sequentially
- All uP have fixed bit size

# FPGA Evolution for SDR

- **Chapter 1 – Evolution from uP to FPGA**

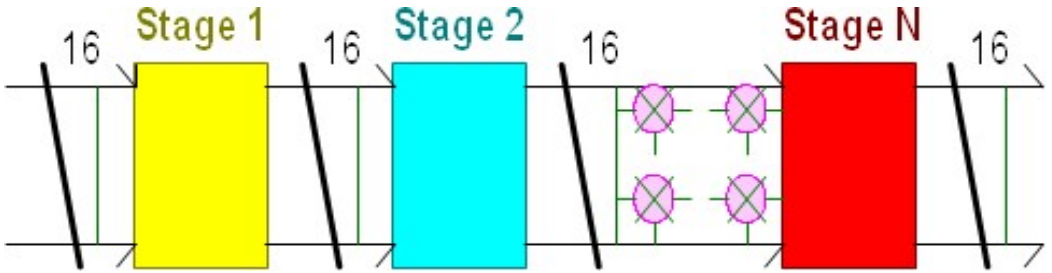


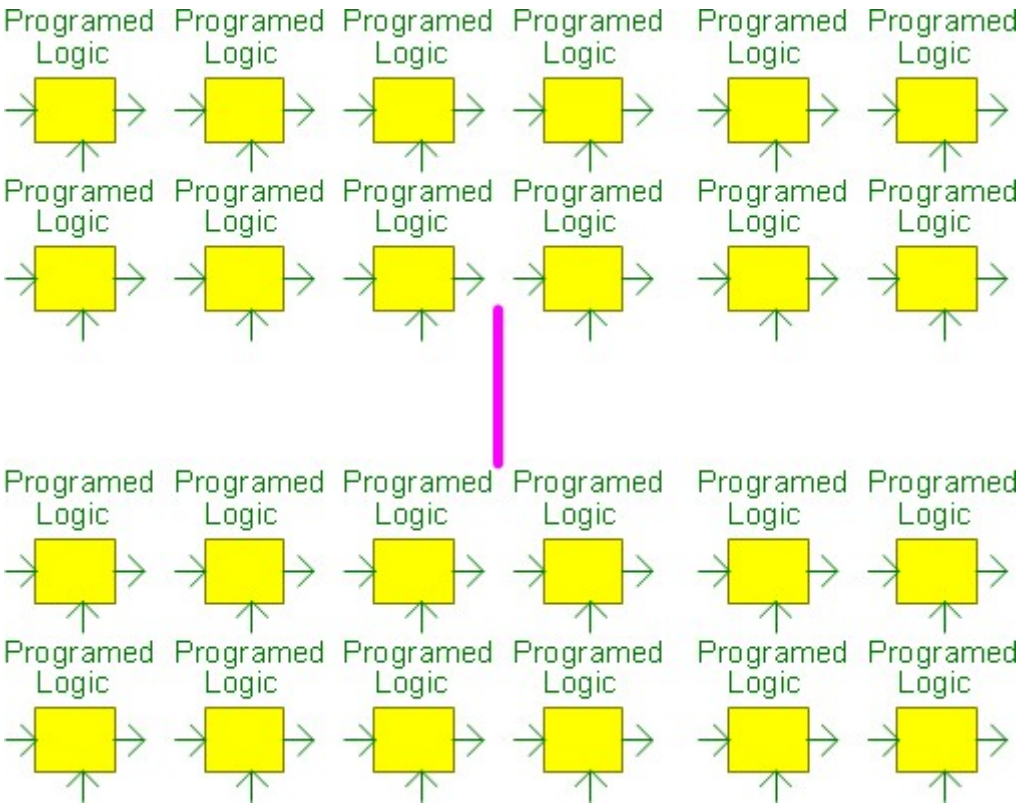




- 8 Bit uP Example
- Modern Computer CPU now 64 Bit
- Bits are like soldiers crossing a bridge, side by side
- In principle, 64 bits should be 8 times faster than 8 bits

# FPGA Evolution for SDR

- **Chapter 1 – Evolution from uP to FPGA**




- Digital processing (DSP) applications often need dedicated high speed M x N Bit integer multipliers
- DSP IC devices were developed to meet the needs for exact audio processing, e.g. to replace analog filters (FIR, IIR)
- Here is a DSP concept with 4 multipliers
- DSP are faster but still sequential
- Field Programmable Gate Arrays were developed to be parallel or sequential
- Arbitrary Gate configuration, any bit size (e.g. 800 bits OK) and processing is completely parallel (and/or sequential)

# FPGA Evolution for SDR

- **Chapter 1 – Evolution from uP to FPGA**

- FPGA devices used for very high speed signal processing – Video, Advanced Communications
- Reprogrammable, used to develop code → **ASIC**
- Code written on a PC → **bit stream to FPGA**
- Logic elements programmable → AND, OR,...
- All gates “wired together” from PC software
- Low cost versions up to 150,000 logic elements

# FPGA Evolution for SDR

- **Chapter 1 – Evolution from uP to FPGA**

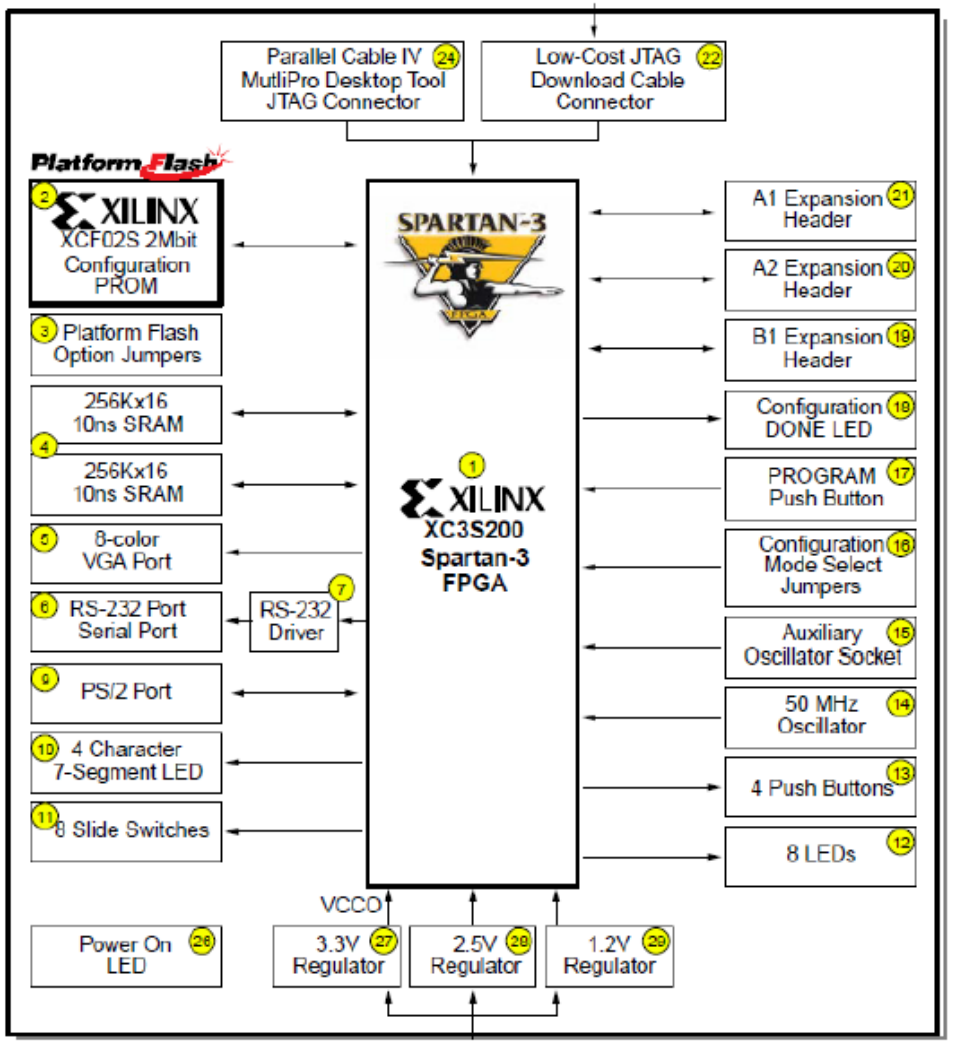
## Low Cost 7-Series Offerings From Xilinx (Alterra also)

*Table 1:* **7 Series Families Comparison**

| Max. Capability | Spartan-7 | Artix-7 | Kintex-7 | Virtex-7 |
|---|---|---|---|---|
| Logic Cells | 102K | 215K | 478K | 1,955K |
| Block RAM(1) | 4.2 Mb | 13 Mb | 34 Mb | 68 Mb |
| DSP Slices | 160 | 740 | 1,920 | 3,600 |
| DSP Performance(2) | 176 GMAC/s | 929 GMAC/s | 2,845 GMAC/s | 5,335 GMAC/s |
| Transceivers | – | 16 | 32 | 96 |
| Transceiver Speed | – | 6.6 Gb/s | 12.5 Gb/s | 28.05 Gb/s |
| Serial Bandwidth | – | 211 Gb/s | 800 Gb/s | 2,784 Gb/s |
| PCIe Interface | – | x4 Gen2 | x8 Gen2 | x8 Gen3 |
| Memory Interface | 800 Mb/s | 1,066 Mb/s | 1,866 Mb/s | 1,866 Mb/s |
| I/O Pins | 400 | 500 | 500 | 1,200 |
| I/O Voltage | 1.2V–3.3V | 1.2V–3.3V | 1.2V–3.3V | 1.2V–3.3V |
| Package Options | Low-Cost, Wire-Bond, Quad Flat Package | Low-Cost, Wire-Bond, Lidless Flip-Chip | Lidless Flip-Chip and High-Performance Flip-Chip | Highest Performance Flip-Chip |

# FPGA Evolution for SDR

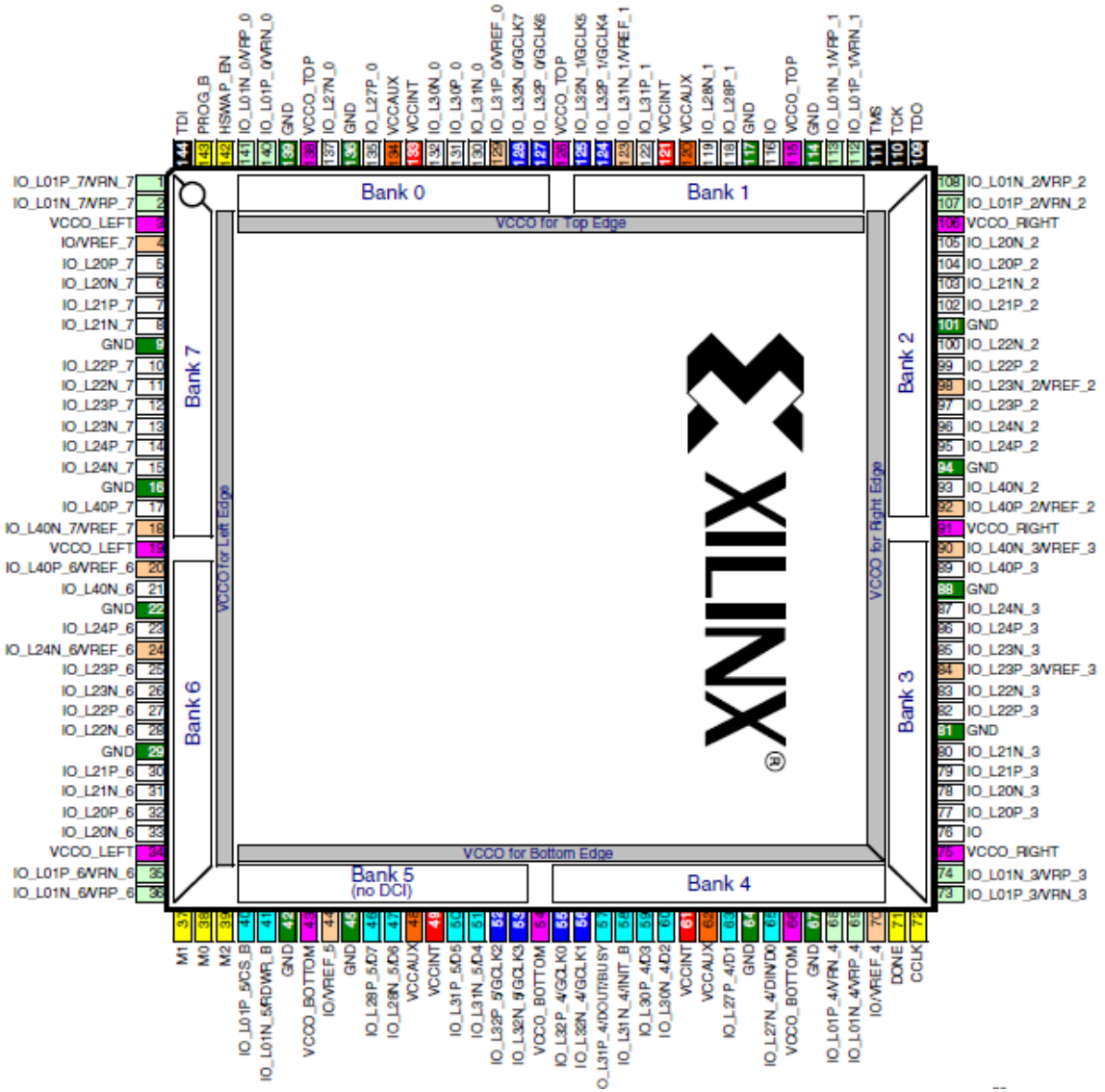
- **Chapter 1 – Evolution from uP to FPGA**




- **Example Low Cost FPGA**
- **Good Peripheral Support**
- **Reasonable Gate Count**
- **Clock ~50 MHz, max 133 MHz**
- **Some FPGA Clock >> 1 GHz**
- **Unlike CPU, bit size unlimited, set per variable**
- **A = [1077,0] quite OK**

# FPGA Evolution for SDR

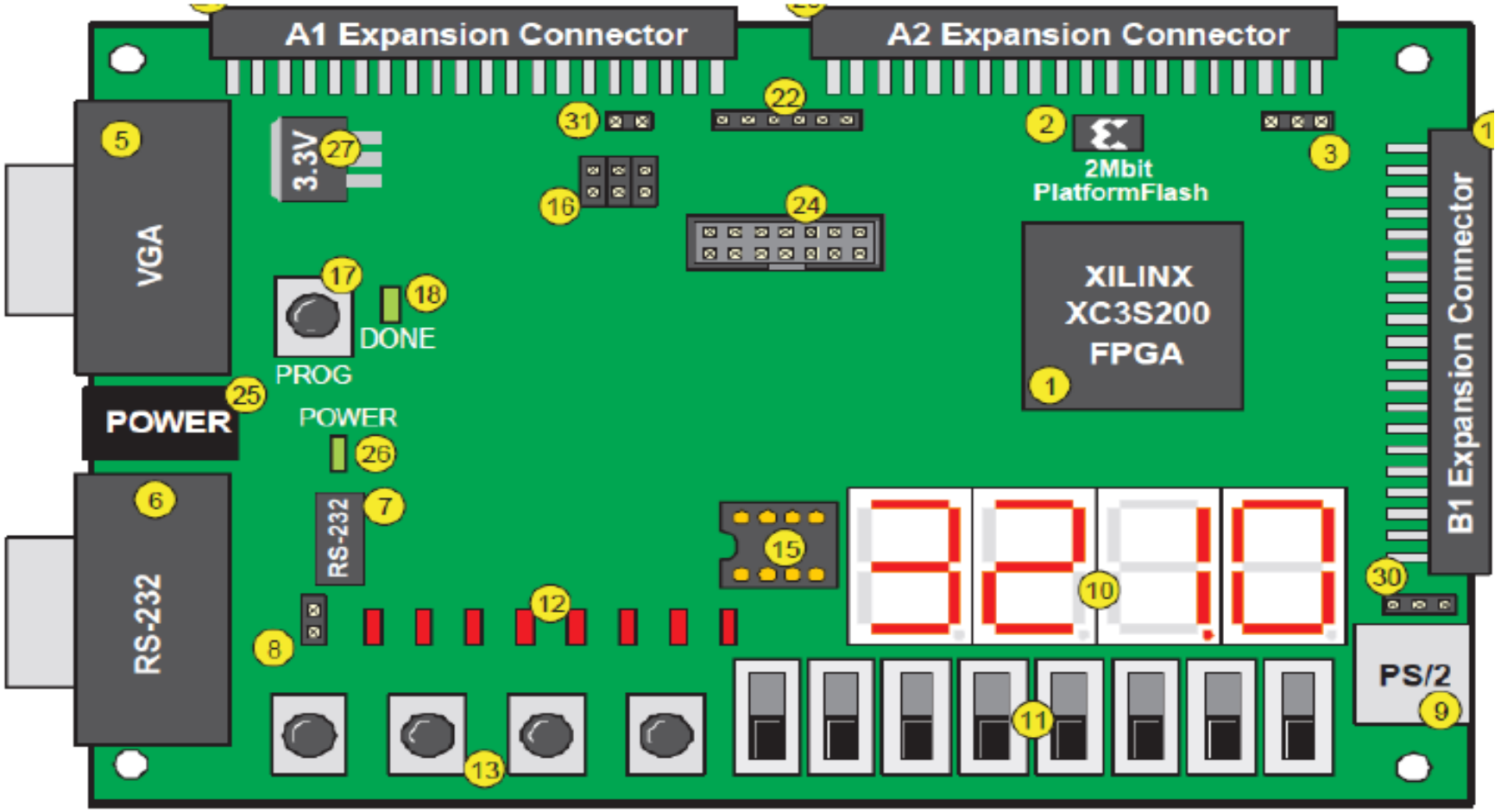
- **Chapter 1 – Evolution from uP to FPGA**



# FPGA Evolution for SDR

- **Chapter 1 – Evolution from uP to FPGA**

- **Example FPGA Evaluation PCB – Xilinx Example**



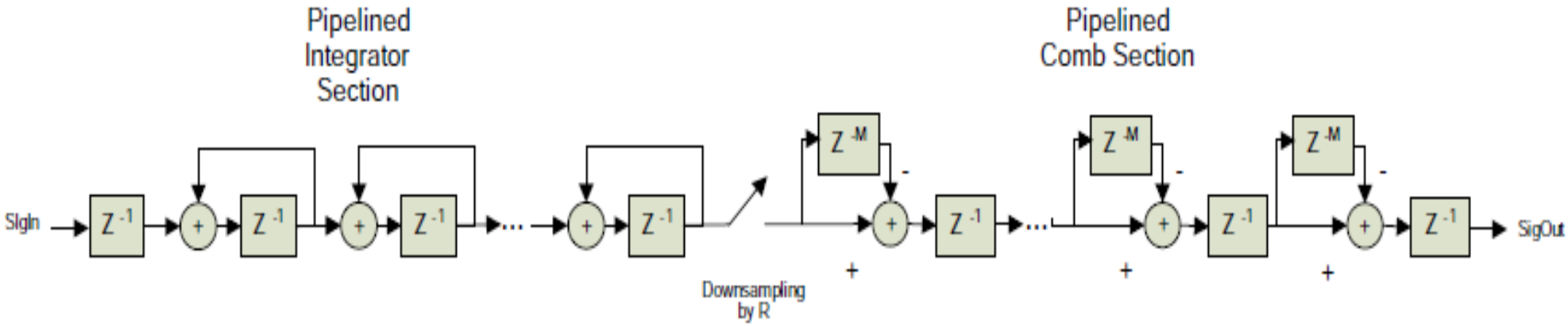
# FPGA Evolution for SDR

## Chapter 1 – Summary

- **Early Microprocessors Demonstrated Software Configurable Digital Logic (replaced gates)**
- **But Limited to low Bit size (e.g. 8 Bits, fixed)**
- **And Processing is Sequential – i.e. step by step**
- **Computer CPU Architectures now 64 Bit, fixed**
- **DSP provided high speed multipliers – for filters**
- **FPGA non sequential, arbitrary bit sizes, no limit**

# FPGA Evolution for SDR

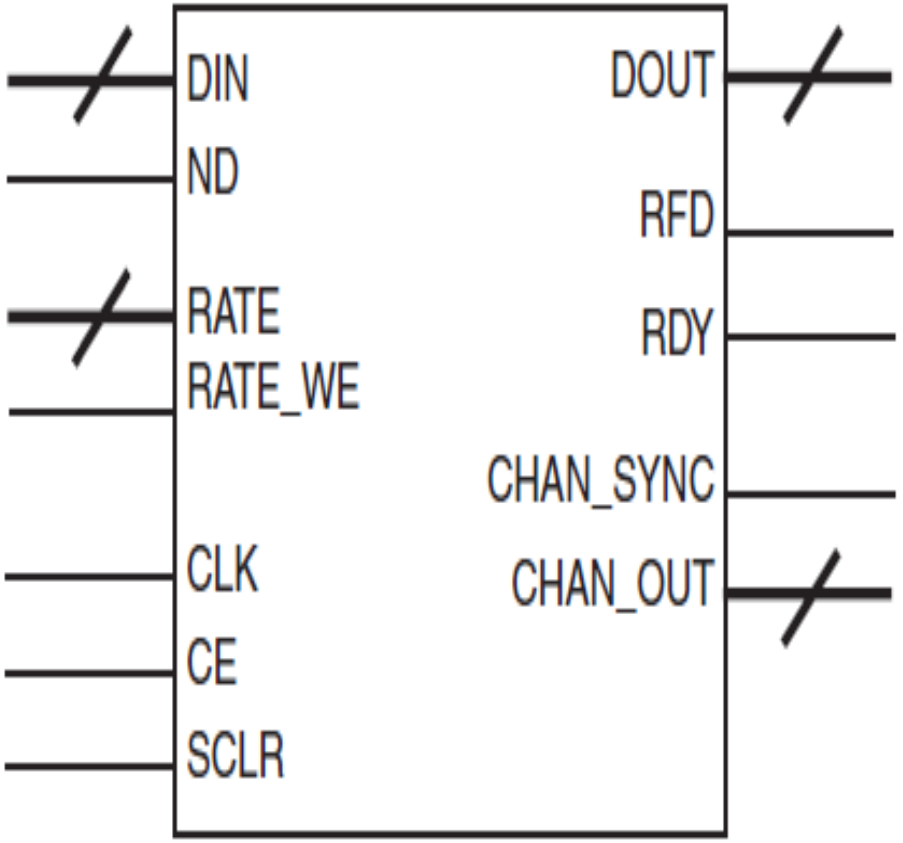
- **Chapter 2 – Examples of FPGA Structures**




- Example Cascaded Integrator Comb CIC Low Pass Digital Filter
- Used as a “Roofing LPF” in SDR – (similar to XTAL → Ceramic)
- Followed by more exact Finite Impulse Response (FIR) Filter or IIR

# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**

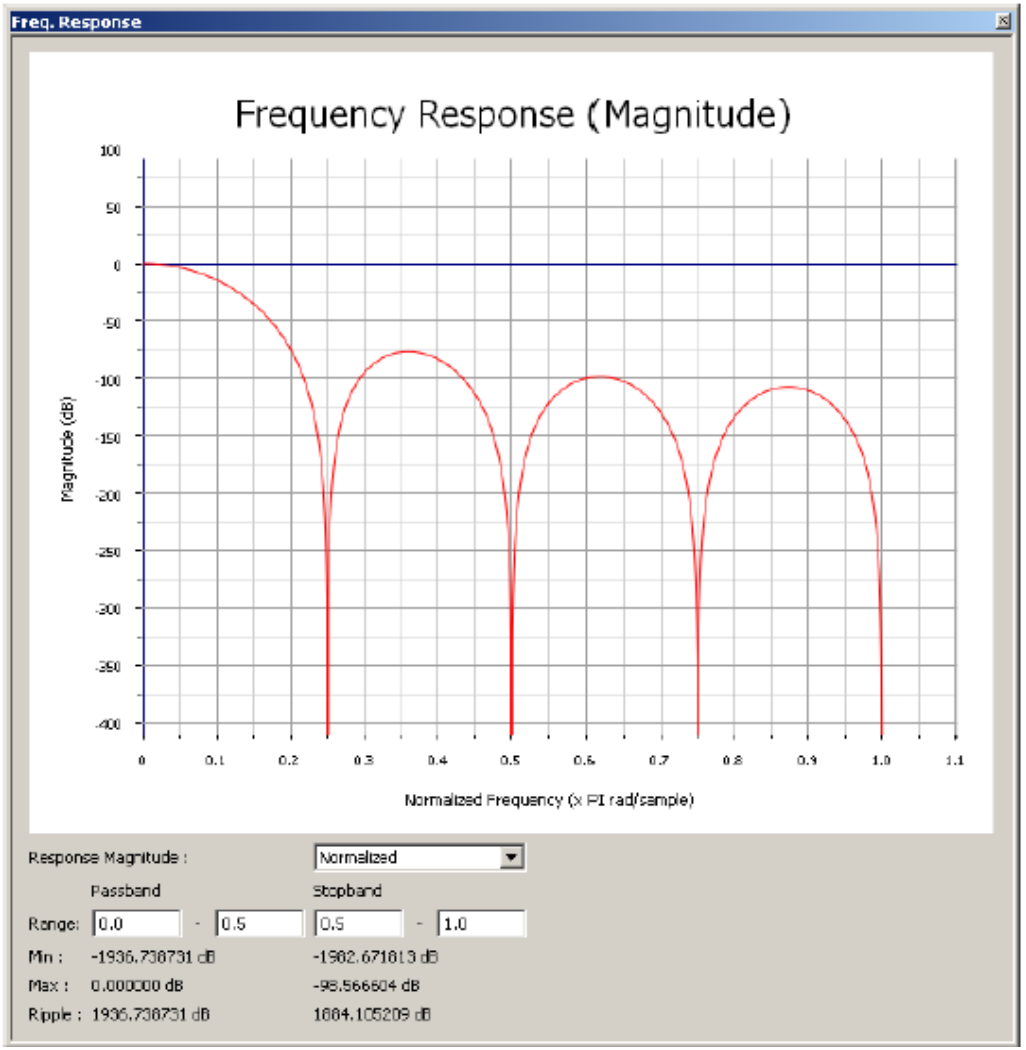



**Note – Bus lines are bold (N-Bit)**

- **FPGA Programming is simplified using Xilinx "LogiCORE™" modules**
- **Symbol shows data in (DIN), data out (DOUT) and control functions for CIC Filter**

# FPGA Evolution for SDR

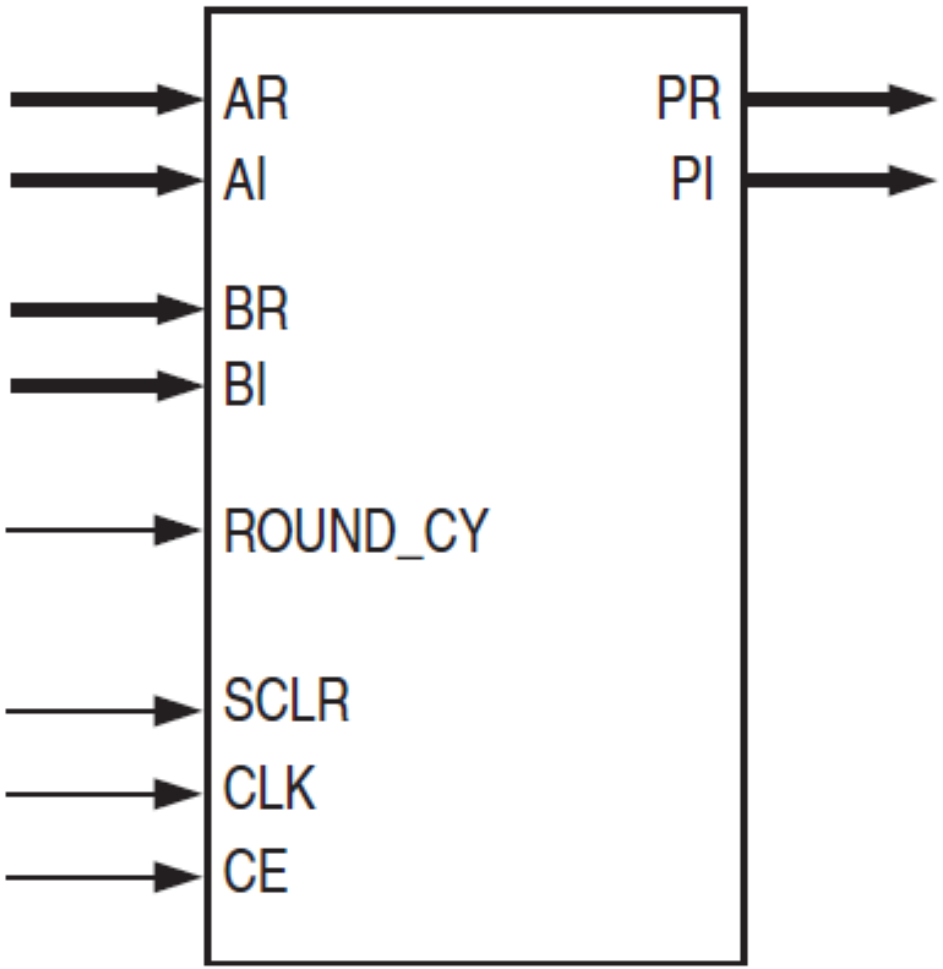
- **Chapter 2 – Examples of FPGA Structures**




- **Typical CIC Low Pass Frequency Response**
- **Stop band is “lazy”**
- **Sinc(x) notches used in filters strategically**

# FPGA Evolution for SDR

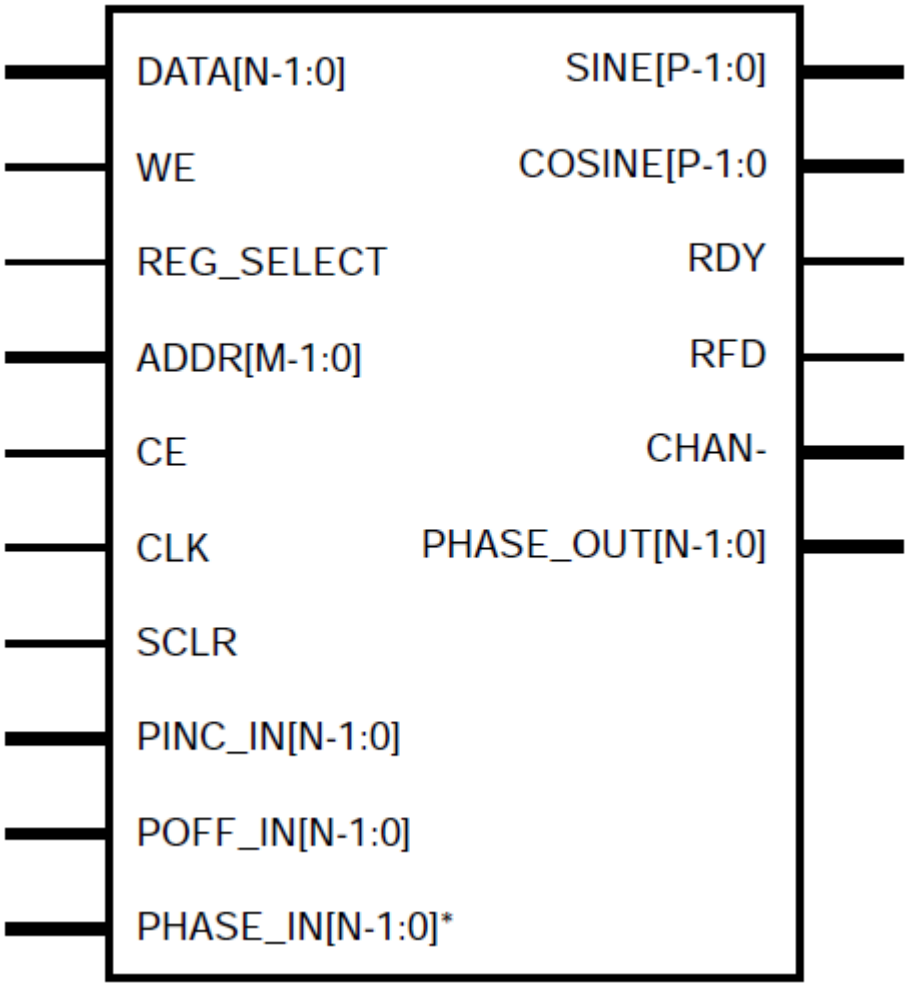
- **Chapter 2 – Examples of FPGA Structures**




- **LogiCORE™ IP modules simplify complex logic operations**
- **Example Complex Multiplier This module implements [PR + j PI] = [AR + j AI] x [BR + j BI]**
- **Extremely difficult to design using hard-wired logic gates!**
- **Modules perform Excellently!**

# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**

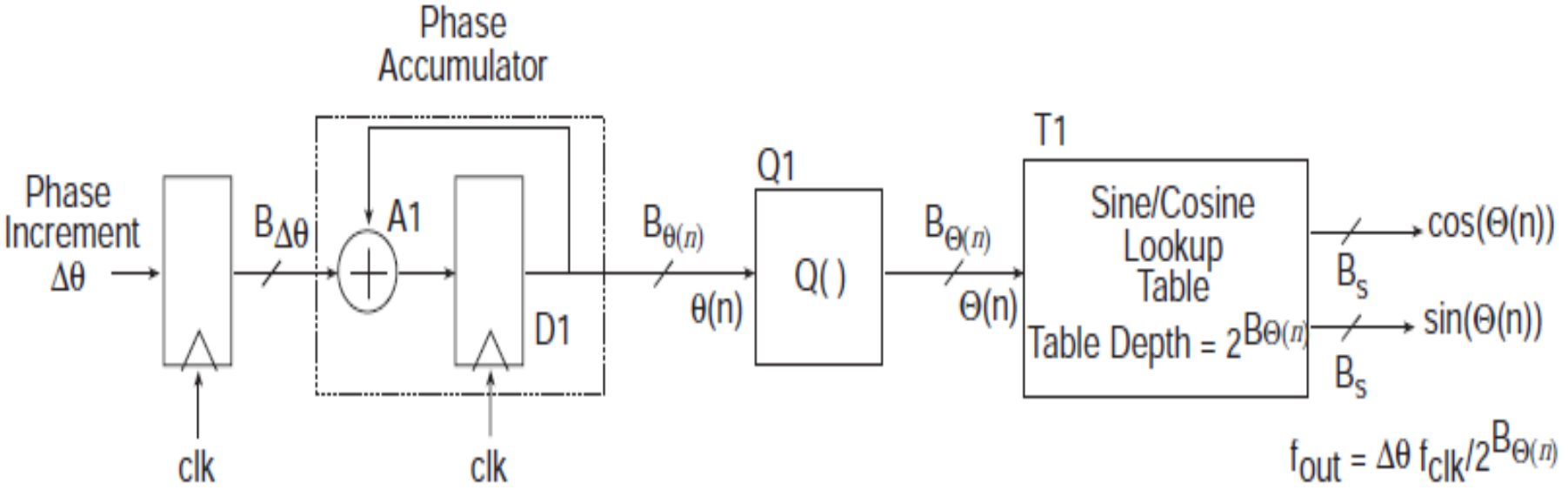



- **Example Direct Digital Synthesis**
- ***Extremely fine frequency resolution***
- **Phase quadrature out**

# FPGA Evolution for SDR

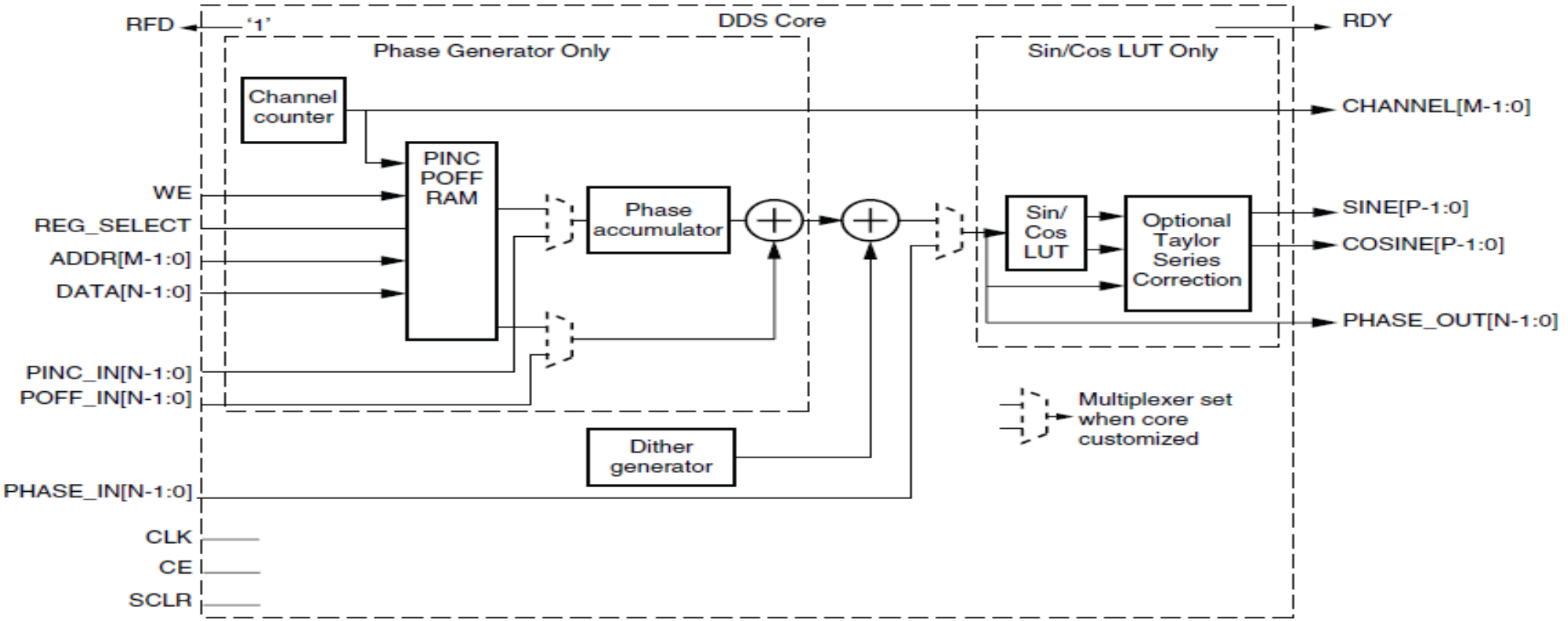
- **Chapter 2 – Examples of FPGA Structures**

## Generic DDS Structure



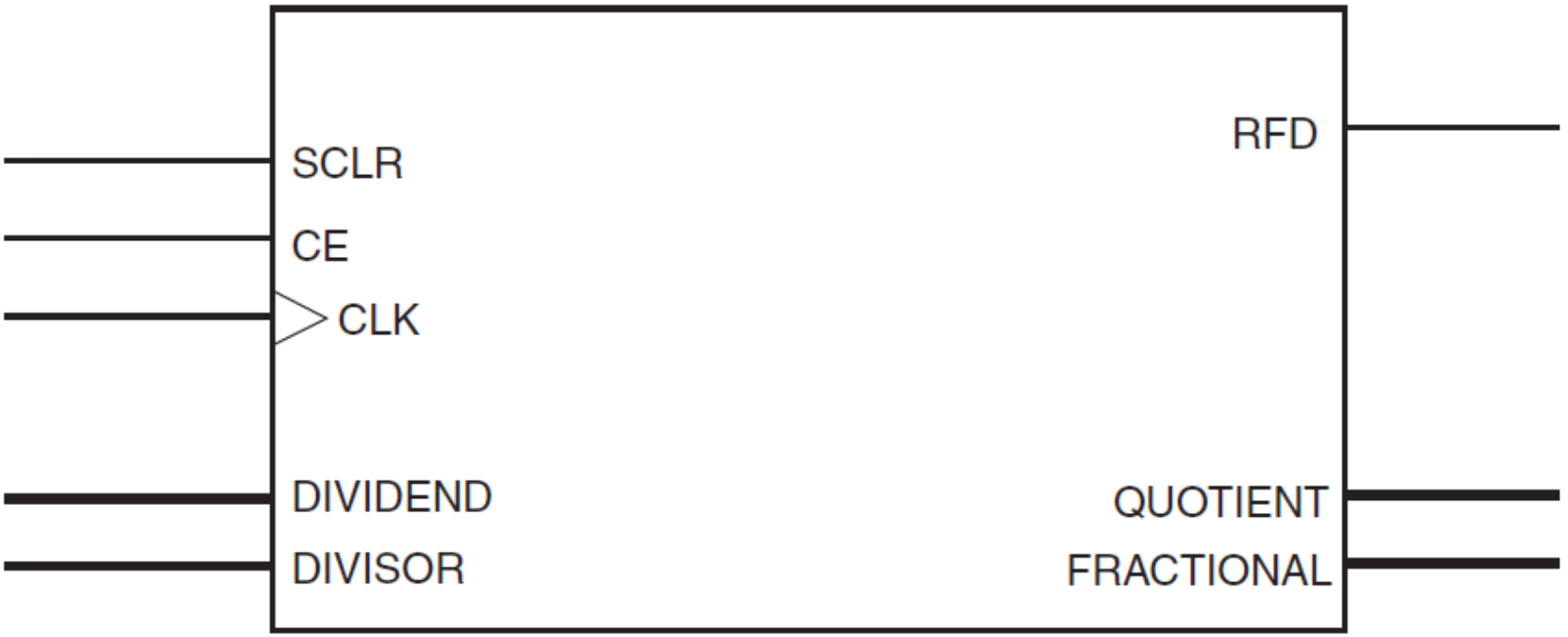
# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**
  - **Advanced DDS Structure**
  - **Dither & polynomial correction**



# FPGA Evolution for SDR

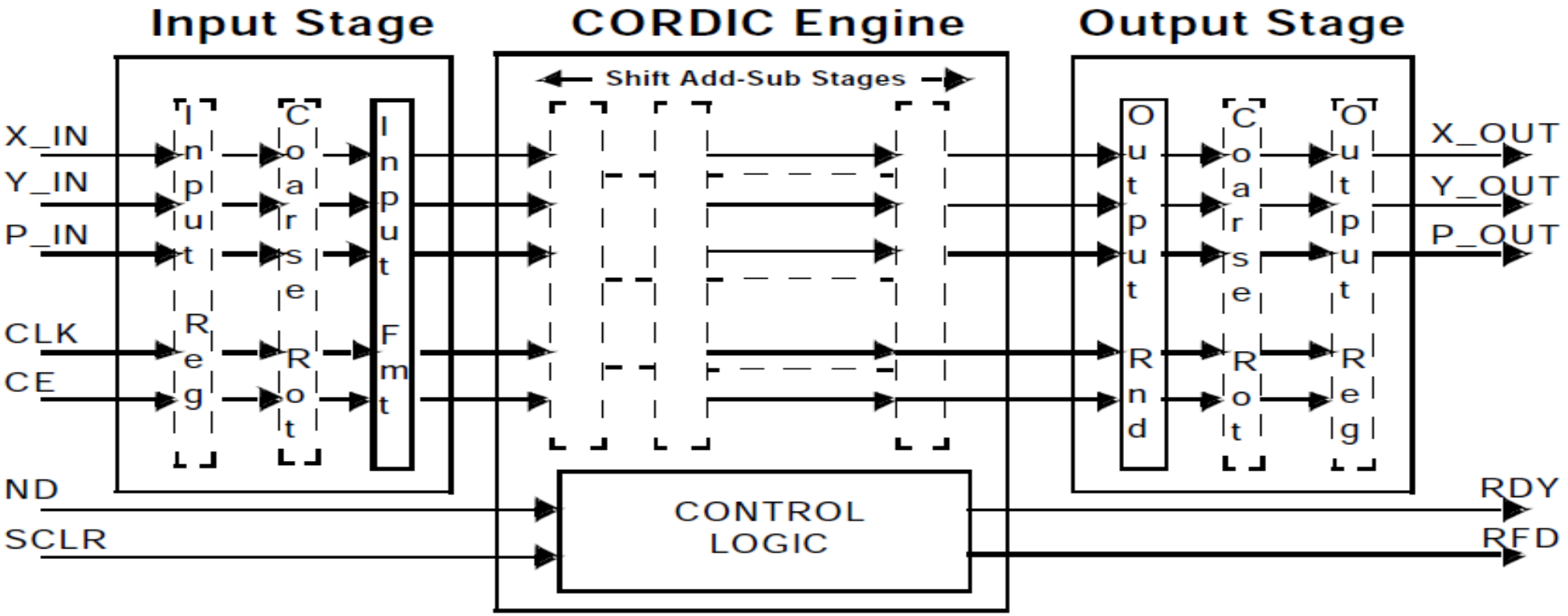
- **Chapter 2 – Examples of FPGA Structures**




- **Division Symbol, Dividend / Divisor = Quotient + Remainder**
- **Extremely hard to design with hard-wired logic gates!**
- **Remember doing "Long Division" in High School?**

# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**

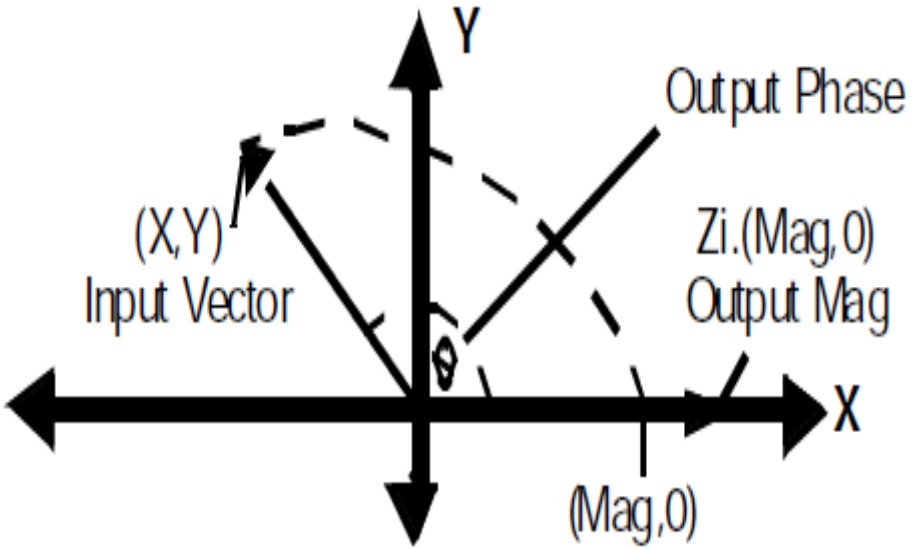


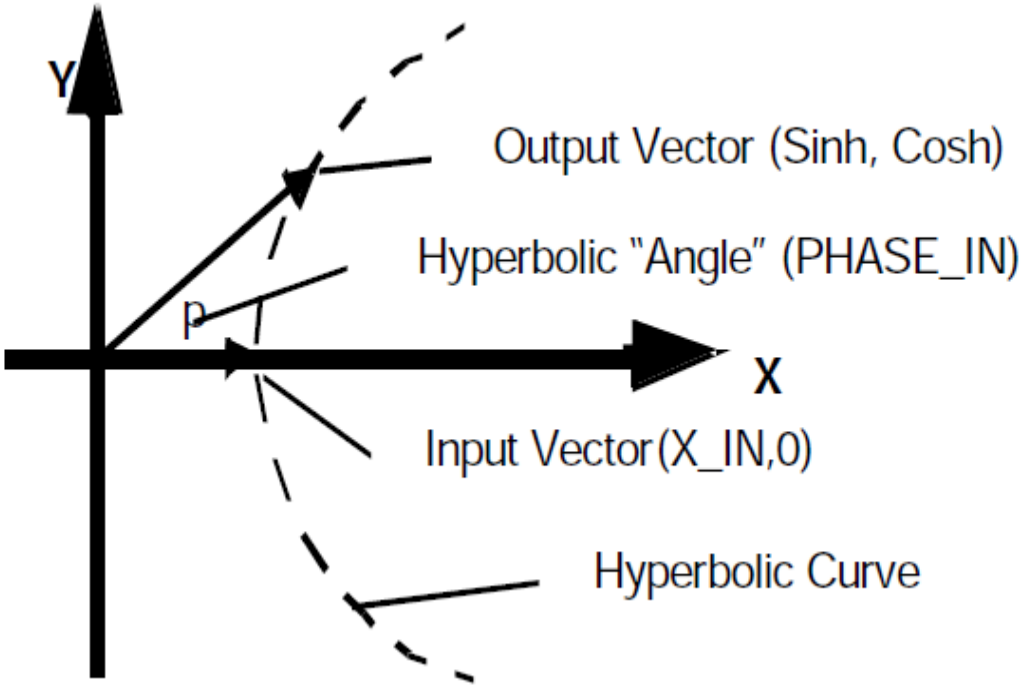

- **"CORDIC" Symbol – Used for rotation, trigonometry, logarithmic conversion etc – Impossible with gates!**

# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**

## • CORDIC Trigonometry Examples

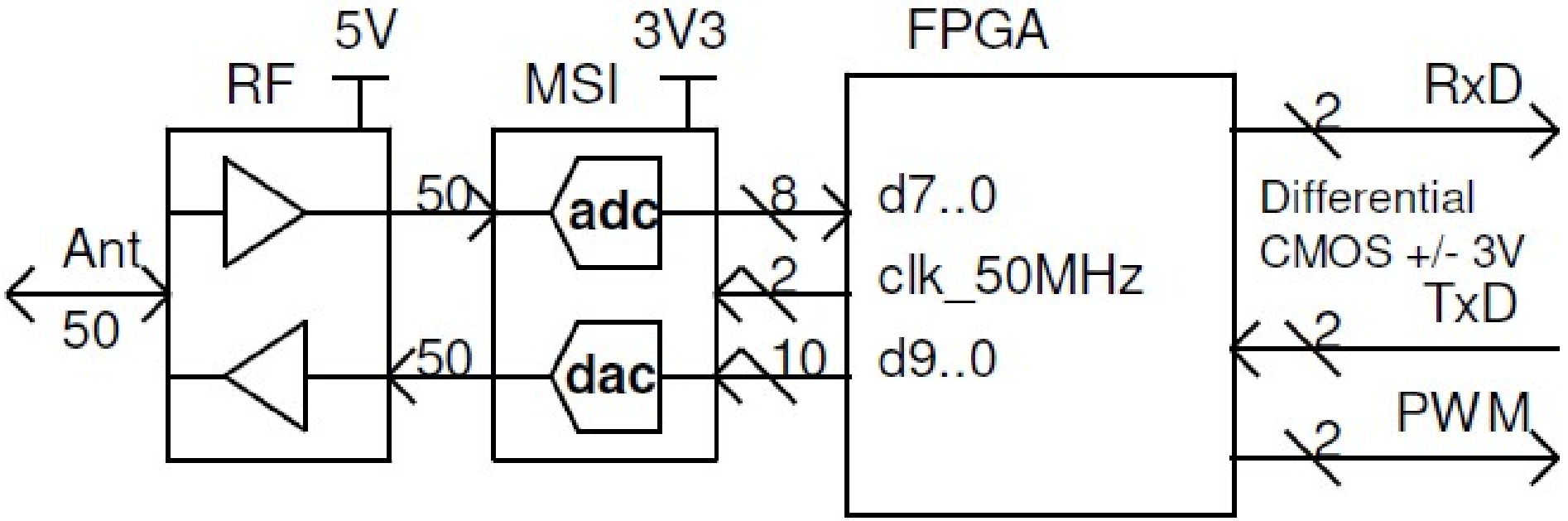



The Xilinx LogiCORE™ IP CORDIC core implements a generalized coordinate rotational digital computer (CORDIC) algorithm.

# FPGA Evolution for SDR

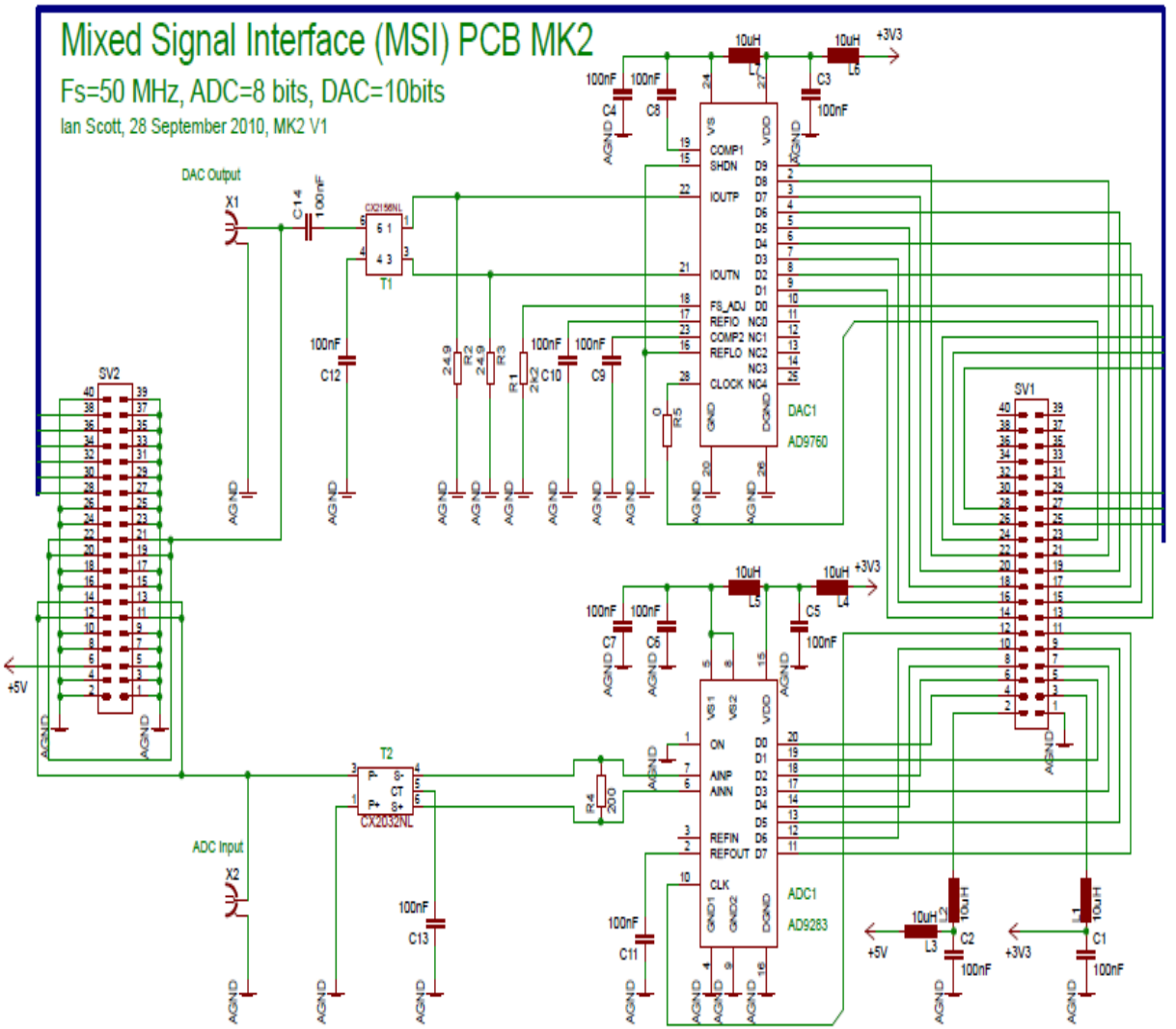
- **Chapter 2 – Examples of FPGA Structures**




- **Generic Software Defined Radio (SDR) System**
- **RF Signal Conveyance, Mixed Signal Interface, FPGA**
- **Generic Structure, Applicable to High Performance**

# FPGA Evolution for SDR

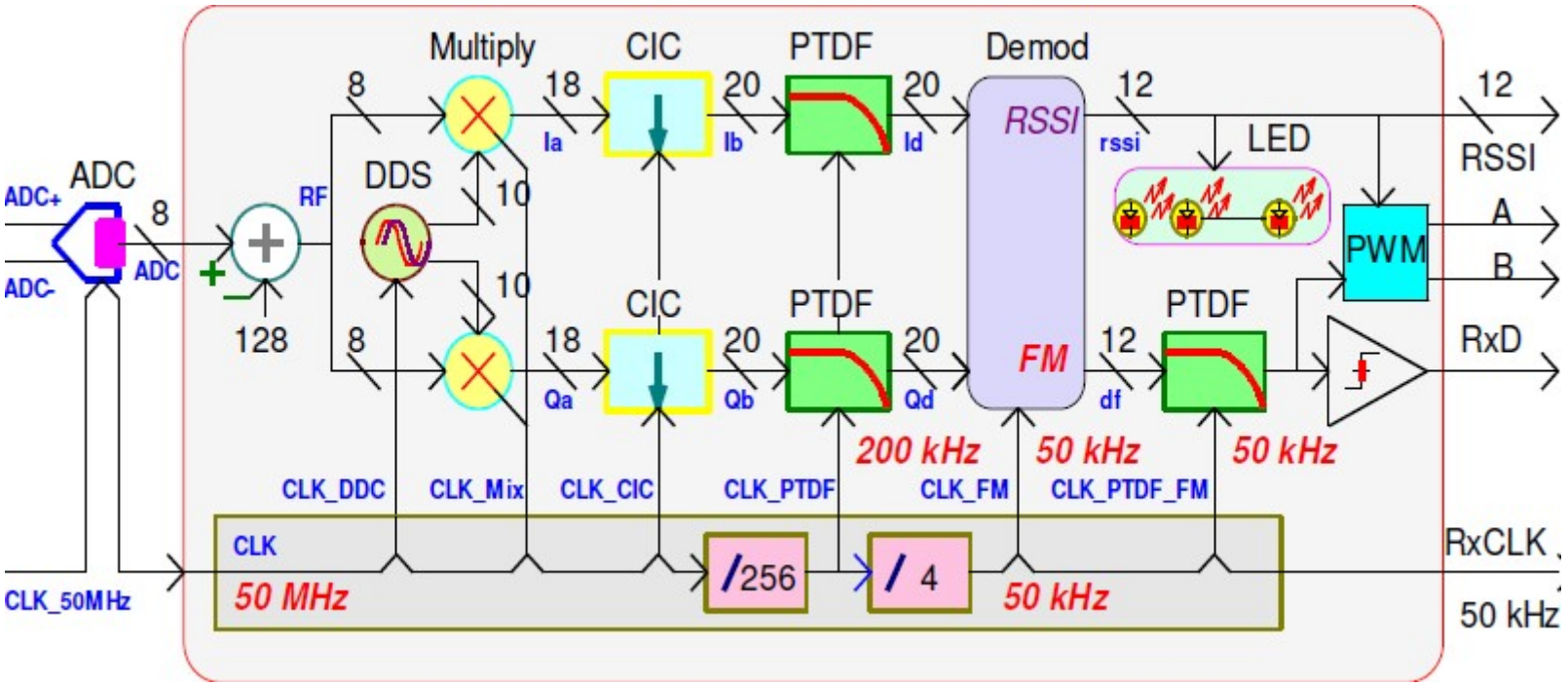
- **Chapter 2 – Examples of FPGA Structures**




- **MSI Transceiver**
- **Low Cost ~ $20 total**
- **8-Bit Rx ADC at Top**
- **Lower 10-Bit Tx DAC**
- **Sample $F_s$ = 50 MHz**
- **ADC BW ~ 500 MHz**
- **3.3 V Supply. Low $I_s$**
- **Non Critical FPGA Interface – see FPGA EVAL PCB connectors**

# FPGA Evolution for SDR

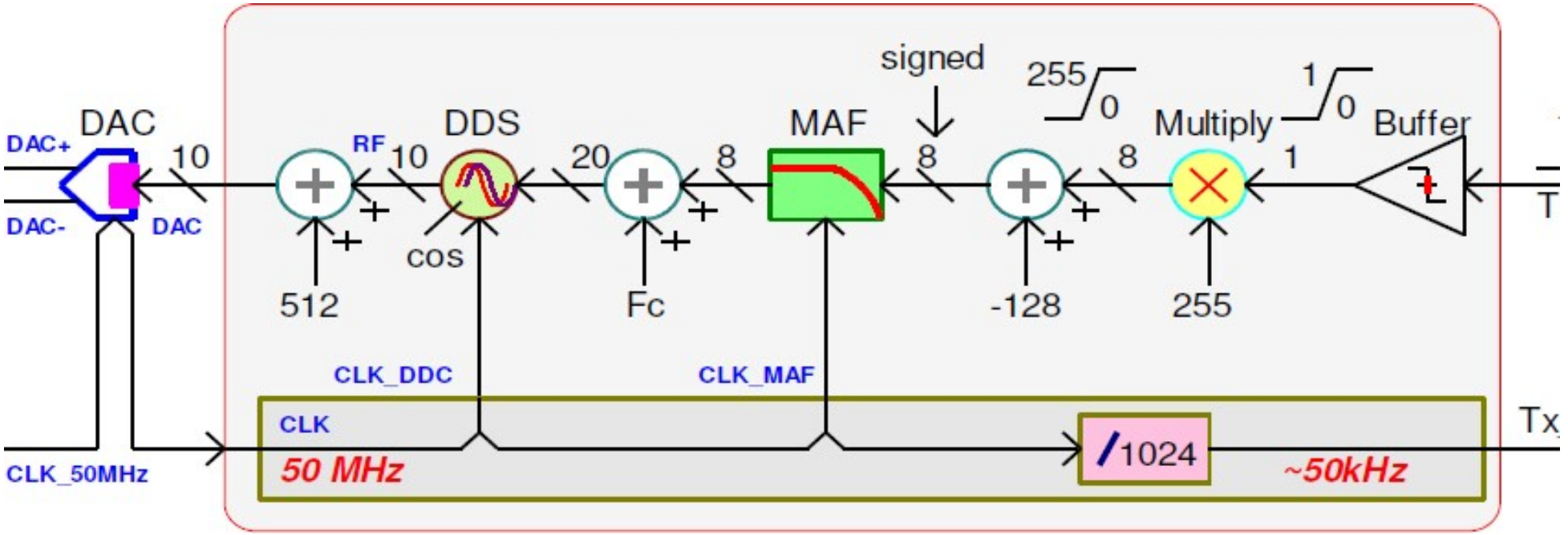
- **Chapter 2 – Examples of FPGA Structures**




- **Example SDR Receiver System Processing Steps**
- **ADC Signal Acquisition, I+j Q Bit Growth, CIC Filter, Detailed IQ LPF, Demodulation, Formatting**

# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**

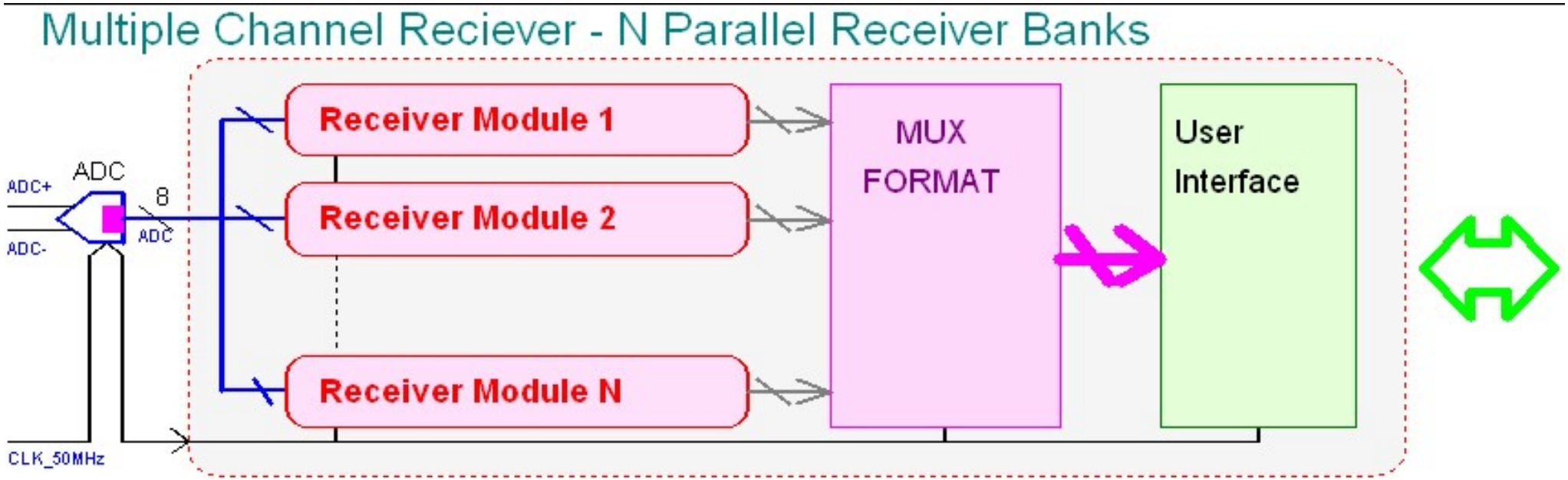



- **Example SDR Transmitter System Processing Steps**
- **Input on Right, Scale, Moving Average Filter, Direct DDS Modulation, DAC Output For Filtered FSK**

# FPGA Evolution for SDR

- **Chapter 2 – Examples of FPGA Structures**

## Example Multiple Channel Software Defined Receiver




**Parallel Receiver Bank is Far Superior to Scanning Receivers!**

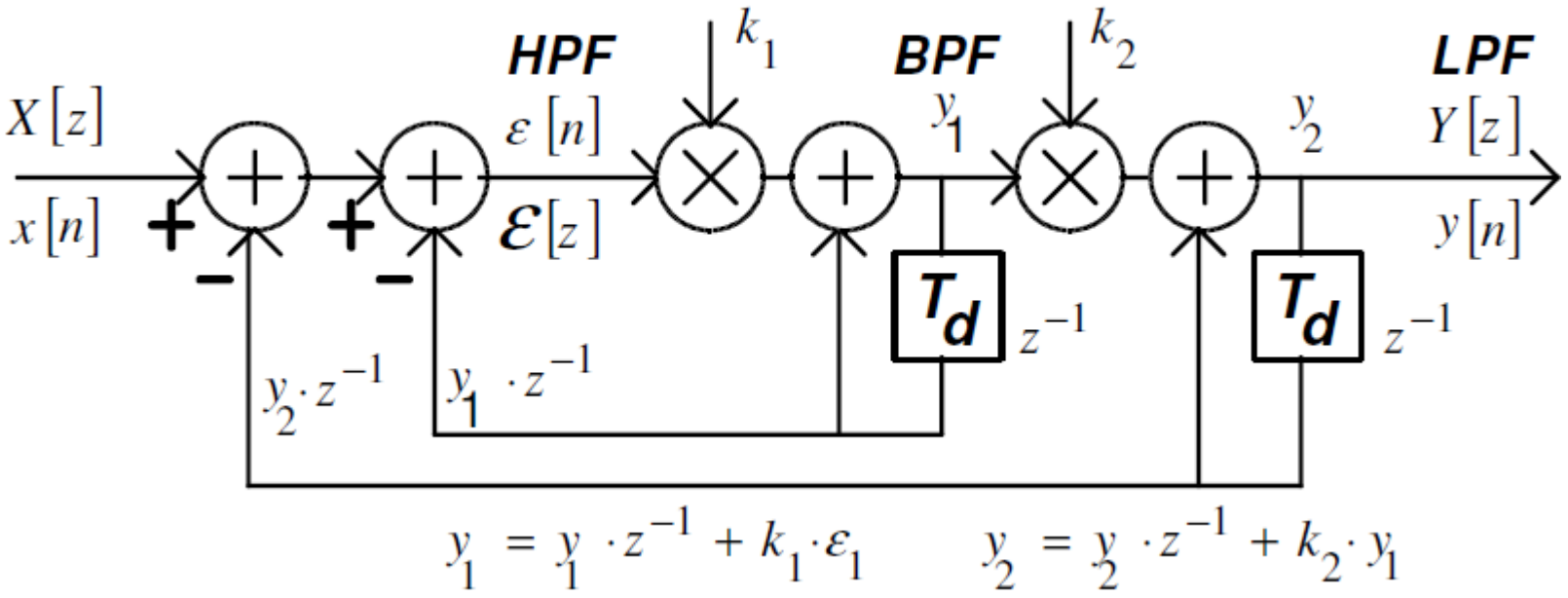
# FPGA Evolution for SDR

## •Chapter 2 – Summary

- Speed up Programming with LogiCORE™ Modules
- Example CIC Filters, Lazy “Roofing” with Sinc(x)
- Complex Multiplication, DDS and Enhanced DDS
- Avoid those Lashes of Logic Based Long Division!
- CORDIC – Rotation, Trigonometry, Logarithms
- MSI Interface to FPGA → RF, ADC / DAC, FPGA
- Parallel Processing Application – Muti-Channel Rx

# FPGA Evolution for SDR

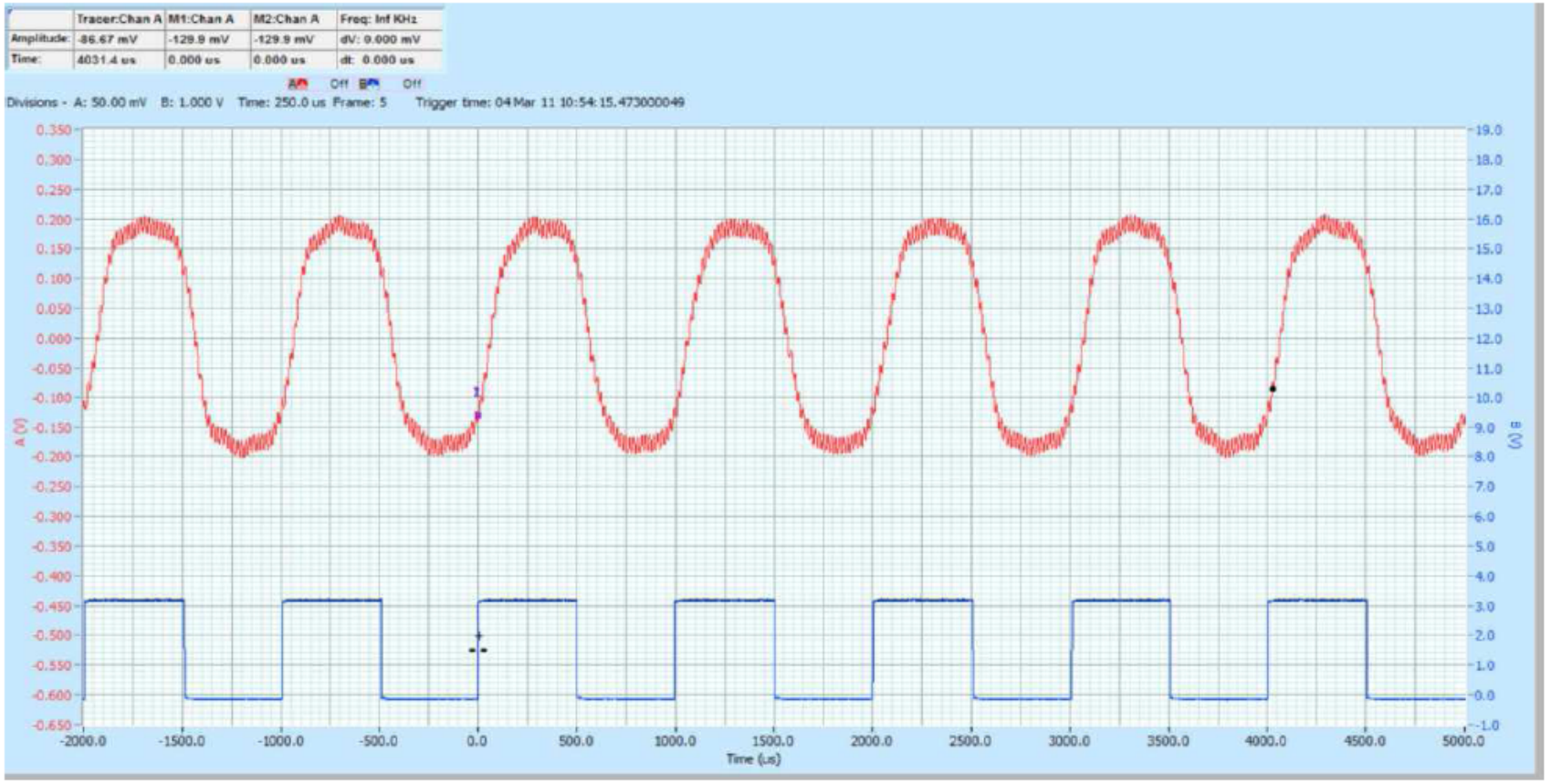
- **Chapter 3 – High Level VHDL Code Examples**




- **Selective Parameter Tuned Digital Filter Architecture**
- **Infinite Impulse Response (IIR) Architecture - Feedback**

# FPGA Evolution for SDR

- **Chapter 3 – High Level VHDL Code Examples**




**Measured SDR Receiver Demodulation Waveforms**

# FPGA Evolution for SDR

- **Chapter 3 – High Level VHDL Code Examples**

```
real a, b, c ; // a,b,c to be real


integer j, k ; // integer variable
integer i[1:32] ; // array of integer variables
```

```
module foo2 (cs, in1, in2, ns);
   input [1:0] cs;
   input in1, in2;
   output [1:0] ns;
   function [1:0] generate_next_state;
   input[1:0] current_state ;
   input input1, input2 ;
   reg [1:0] next_state ;
   // input1 causes 0->1 transition
   // input2 causes 1->2 transition
   // 2->0 illegal and unknown states go to 0
   begin
    case (current_state)
     2'h0 : next_state = input1 ? 2'h1 : 2'h0 ;
     2'h1 : next_state = input2 ? 2'h2 : 2'h1 ;
     2'h2 : next_state = 2'h0 ;
     default: next_state = 2'h0 ;
   endcase
   generate_next_state = next_state;
   end
   endfunction // generate_next_state

   assign ns = generate_next_state(cs, in1,in2) ;
endmodule
```

```
always @(rst)// simple if -else
   if (rst)
         // procedural assignment
         q = 0;
   else   // remove the above continous assign
         deassign q;

always @(WRITE or READ or STATUS)
   begin
   // if - else - if
         if (!WRITE) begin
               out = oldvalue ;
          end
         else if (!STATUS) begin
               q = newstatus ;
               STATUS = hold ;
          end
         else if (!READ) begin
               out = newvalue ;
          end
   end
```

# FPGA Evolution for SDR

- **Chapter 3 – High Level VHDL Code Examples**

```
wire signed [19:0] I, Q;
RF8_IQ20 rfiq(.clk(CLK_50MHz), .RF(ADC), .I(I), .Q(Q) );
//
wire signed [19:0] Ia, Ib, Qa, Qb;
MAF16_20B maf20(.clk(CLK256), .x(I), .y(Ia) );    // 4 stage MAF of 16 samples
MAF16_20B maf21(.clk(CLK256), .x(Q), .y(Qa) );       // 4 stage MAF of 16 samples
//
wire [11:0] p1, p2, p3, p4;  // target BW = +/- 4 kHz
assign p1 = 267;  assign p2 = 1333;       assign p3 = 1600; assign p4 = 4000;
PTDF_LP_4 ptdf0(.clk(CLK256), .p1(p1), .p2(p2), .p3(p3), .p4(p4), .x(Ia), .y(Ib) );
PTDF_LP_4 ptdf1(.clk(CLK256), .p1(p1), .p2(p2), .p3(p3), .p4(p4), .x(Qa), .y(Qb) );
//
wire [15:0] RSSI;
wire signed [15:0] FM;
FM_Demodulator fm0(.clk(CLK512), .I(Ib), .Q(Qb), .RSSI(RSSI), .FM(FM) );
//
wire signed [15:0] Filt_RSSI, Filt_FM;
```

# FPGA Evolution for SDR

- **Chapter 3 – High Level VHDL Code Examples**

## Some VHDL Code Excerpts for FM/FSK SDR Receiver

```
begin
II  <=  I;                  QQ  <=  Q;
III <=  II;                 QQQ <=  QQ;
dI  <=  II - III;       dQ  <=  QQ - QQQ;
sI  <= (II + III) / 2; sQ  <= (QQ + QQQ) / 2;
end
```

```
division div1(.clk(clk),
  .dividend(omega),
  .divisor(power),
  .quotient(quof),
  .fractional(FM),
  .rfd(rfdf) );
```

```
wire signed [divider_bits - 1:0] p0, p1, p2, p3; // 32 bit product = division IP limit
multiply mult0(.clk(clk), .a(sI / 16), .b(dQ / 16), .p(p0) );      // 20 bits / 16 -> 16 bits
multiply mult1(.clk(clk), .a(sQ / 16), .b(dI / 16), .p(p1) );      // 20 bits / 16 -> 16 bits
assign omega = p0 - p1;    // difference of product3 = 32 bits
//
multiply mult2(.clk(clk), .a(sI / 16), .b(sI / 16), .p(p2) ); // 20 bits / 16 -> 16 bits
multiply mult3(.clk(clk), .a(sQ / 16), .b(sQ / 16), .p(p3) );    // 20 bits / 16 -> 16 bits
```

# FPGA Evolution for SDR

## Executive Summary

- Discussed from uP, CPU, DSP to FPGA
- Demonstrated some Xilinx LogiCORE™ IP Modules that execute very Complex Logic Functions
- Typical SDR Architecture – RF, MSI, FPGA, Code
- Showed some examples of VHDL / Verilog Code

- **All Questions Welcome...**