Processor Performance Profiles for RFID Sensor Data Fusion

Dr. Andrew Strelzoff[1]

Abstract – Software defined radio is an exciting new area of research in which the basic functionality of a radio device is defined in an embedded processor. This provides tremendous flexibility in programming a variety of Radio related algorithms. Field Programmable Gate Arrays (FPGA) offer the opportunity to not only change the code running on an embedded processor but also change the processor itself. Radio Frequency Identification (RFID) presents numerous practical problems which may be solved through the use of software defined FPGA-implemented radio technology. One of the most challenging of these problems is the accurate positioning of thousands of RFID-tagged objects using multi-lateration techniques. One of the first steps in developing a robust embedded FPGA based system which can triangulate thousands of moving objects is to select a soft processor or processors on which the algorithms will run. It is expected that additional application specific hardware will need to be developed, but the initial selection of a size-efficient processor is extremely important.

This paper presents initial profiling and performance metrics for Radio Frequency Identification [RFID] Positional Sensor Data Fusion using two soft processor architectures. Implementation size, code size and running time are compared for the TSK51 microcontroller and the TSK 165 RISC processor. The RISC architecture achieved faster run time performance but was less size efficient on a triangulation per logic unit used basis. This mixed result has motivated further study of the issue and the continued effort with more advanced NIOS II and TSK3000 processors. This effort and the first steps towards application specific architecture for RFID sensor fusion are discussed.

Keywords: Sensor Data Fusion, Active RFID, RFID Positioning, Application Specific Architecture

INTRODUCTION

Radio Frequency Identification (RFID) is widely used in inventory tracking using passive RFID tags. A passive RFID tag has no battery and uses the inductive field of the reader to gain just enough power to reflect a signal with a few imposed information bits. Passive RFID is cheap and effective at short ranges, usually no more than a few inches. Active RFID involves a battery powered tag which is capable of sensing interrogation at a greater distance and is able to originate a response to its base station. Location of a tag is accomplished through multi-literation, essentially the timing of signal and response between a tag and three or more base stations. Older active RFID systems have effective read ranges of between 8 and 20 feet and have been designed to locate up to 2000 tagged object simultaneously [7]. Newer systems have ranges of 80 to 300 feet and theoretically could position up to one hundred thousand items simultaneously. The aim is to replace dozens or hundreds of short range readers with just a few long range readers. Long range RFID positioning has been successfully deployed to locate relatively few high-ticket items such as firefighting equipment [1] but it has not been successful deployed in large scale inventory control. One barrier to the more general implementations of longer range positioning RFID systems is the computational complexity of multi-latertion calculations.

---

[1] Department of Computer Science, University of Southern Mississippi , 118 College Blvd. Hattiesburg MS. 39406, and strelz@orca.st.usm.edu

RFID embedded processor designs have typically relied upon a single processor to handle positioning, encryption, decryption, the detection and elimination of multiple images at overlapping station boundaries and communication with a remote database [8]. This paper focuses on the development of an embedded co-processor, a specialized processor which will handle up to 100,000 multi-lateration calculations per second, freeing the primary embedded processor to concentrate on other necessary tasks.

FPGA BASED RFID SENSOR FUSION

An FPGA, as shown in Figure 2, is an integrated circuit imprinted with thousands of logic cells or "gate arrays" which are "field programmable". Each logic cell is basically a multiplexer attached to the rest of the array through programmable memory switches. Pathways through the gate array are set by setting the bit pattern of the switches.

The basic architecture of an FPGA based RFID sensor fusion system is one or more radio antennas attached through Analog to Digital converters [ADC] to an FPGA. Several embedded processors on the FPGA will then cooperatively decode and decrypt signals, triangulate tagged objects, eliminate multiple images, encrypt results and send these results on to a centralized database where the positional results will be available for query and other secondary processing. Both co-processor throughput and FPGA resources expended are important considerations for selecting an embedded processor for multi-lateration.

Size is a complex issue. In addition to logical units used the co-processor will need memory for instructions, scratch space and for look-up tables that are used to avoid direct floating point calculation. Floating point is often not supported in compact embedded processors because it requires a great deal of space. RAM memory is available hosted on the FPGA board but in order to avoid conflicts with other processor units the multi-lateration co-processor should use only internal FPGA memory. Banks of un-used switches may be combined to form "register memory". However, this then makes the logical units attached to those switches unusable. For the Therefore compact program size is also an important consideration.

The aim of this research is to develop an FPGA based co-processor that can process 100,000 multi-literation calculations per second and requires only 1000 logic units and uses as few banks of register memory as possible.

For this paper the TSK51 8-bit microcontroller, shown in Figure 4, and the TSK165 8-bit RISC processor, shown in Figure 5, were considered for the task of multi-lateration calculation. The small bit width limits the accuracy of calculations to around a foot or more. This is sufficient for most inventory applications but may prove problematic for higher level sensor fusion. Open source triangulation code [4] was compiled and run on both embedded processors on an Altera Cyclone II FPGA board.

A diagram showing the functionality of the code is shown in Figure 3. Assembled code size, processor size and triangulations per second achieved were considered. Initial results are summarized in Table 1.

Processor	Assembled Code Size	Processor Size	Multi-Laterations per Second
TSK51	2114 lines	2192 logical units	6,250 per second
TSK165	2345 lines	921 logical units	16,340 per second

Table 1 - Preliminary Results for Embedded Processor Performance on RFID triangulation

The larger code size for the TSK165 is more problematic in an FPGA than it would be in a conventional CPU as discussed above. However the minor size disadvantage in code size is more than made up by the TSK165's smaller footprint and much more efficient execution. Much of the

speed advantage can be traced to the TSK's superior 2 clock memory access compared with the 6 clocks per memory access for the TSK51.

When comparing our target metric of (Triangulations per second) / (Logical units engaged) the TSK165 fares even better as shown in Table 2.

Processor	(Laterations per second)/(LUTS utilized)
TSK51	2.85
TSK165	17.74

Table 2 – Processor space efficiency for mult-lateration

The above results were achieved with the Cyclone II FPGA running with a 50MHz clock. The Cyclone II FPGA can be set with speeds up to 260 MHz [9] that would result in nearly 90,000 mulit-lateration calculations per second just short of the 100,000 required.

One possible approach to reach the target of 100,000 multi-lateration calculations per second would be to add specialized hardware to the TSK165 to support frequently repeated tasks such as calculating the sum of squares. The primary savings would be in parallelism, the three squares, X, Y and Z would be calculated in parallel, with secondary savings resulting from fewer instruction fetches since the single SUMSQUARES super instruction would replace dozens of smaller instructions (essentially a CISC approach). Smaller code size would also be a benefit. Unused switch memory in the Cyclone II is organized in banks of 4 kilobytes. If our program size could be reduced to 500 eight bit instructions it would fit efficiently in a single bank of register memory. Another approach is to consider more complex processors that may be able to calculate the results more quickly and accurately with the same clock speed.

OTHER EMBEDDED PROCESSORS

Two other embedded processors, the MIPS based 32-bit NIOS II and the 32-bit RISC TSK3000 remain to be tested. The above research was done with an 8 bit version of the code. In order to locate objects more precisely larger 16 or 32 bit numbers are required. A fundamental question is whether the 32 bit architecture is sufficiently more efficient when dealing with 32-bit numbers to justify the extra size. The NIOS II requires 3300 logic elements compared to 921 for the TSK165.

Another question is the use of embedded hardware cores in FPGA for this project. Several FPGA manufacturers produce chips with actual processors integrated into the FPGA fabric. An example would be the Virtex 4 LC line of FPGA from Xilinx which includes one or more Power PC cores. These cores do not take up any FPGA space and may prove to be faster than purely FPGA defined soft processors.

CONCLUSION

The major finding of this study was that the TSK165 approaches the requirements for a fast and compact embedded co-processor for multi-lateration calculations. Research continues both to make improvements in the TSK165 size and speed performance and to consider more complex processors which may have significant speed and accuracy advantages.

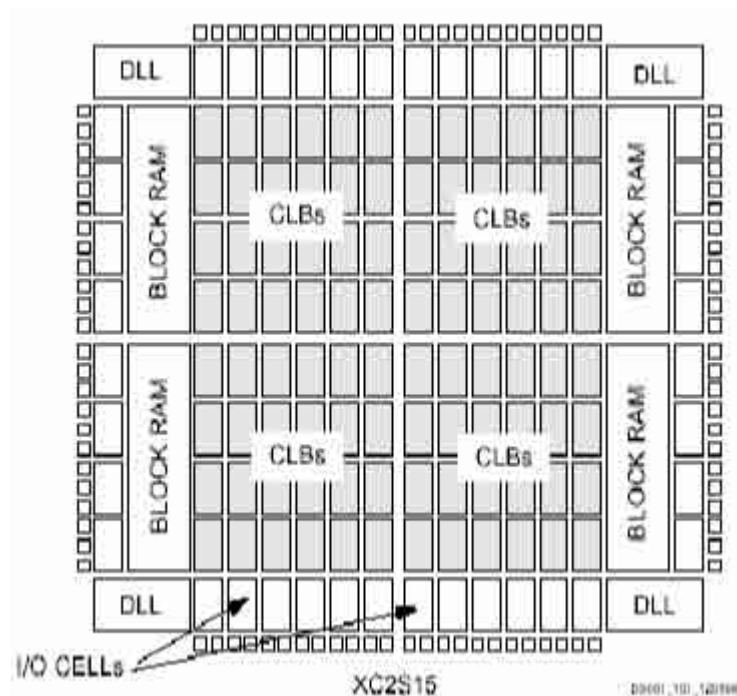**Figure 1 - SFF SDR from Lyrtech [1] which features an Xilinx SX35 [2] user programmable FPGA.**

Figure 2 - FPGA fabric showing arrays of logical units. The FPGA shown is a relatively small Xilinx Spartan II. The FPGA on available with the Lyrtech software defined radio kit is an SX35 with more than 32,000 logical units.
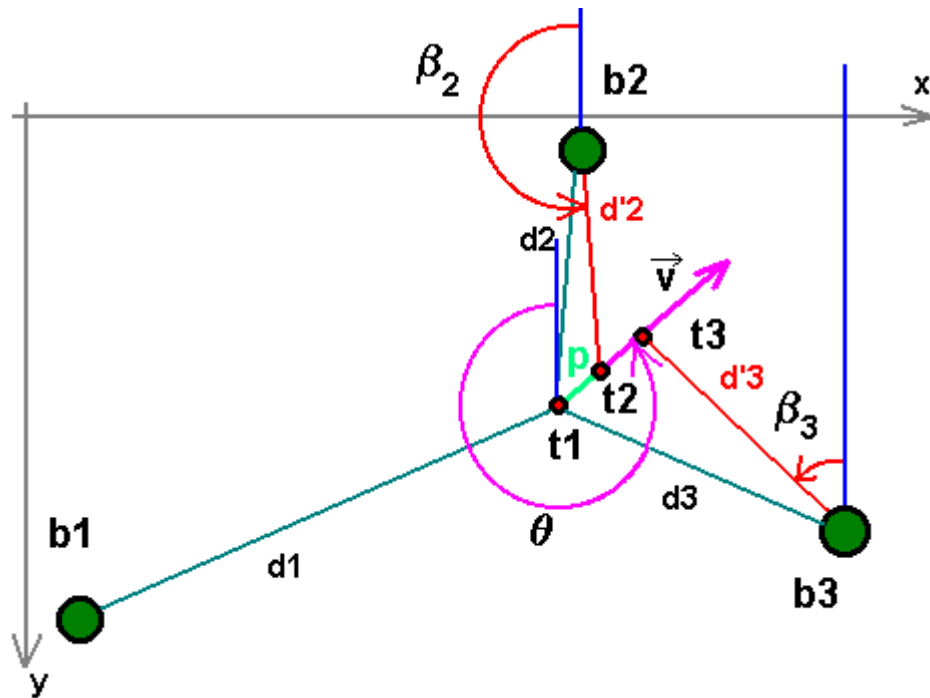
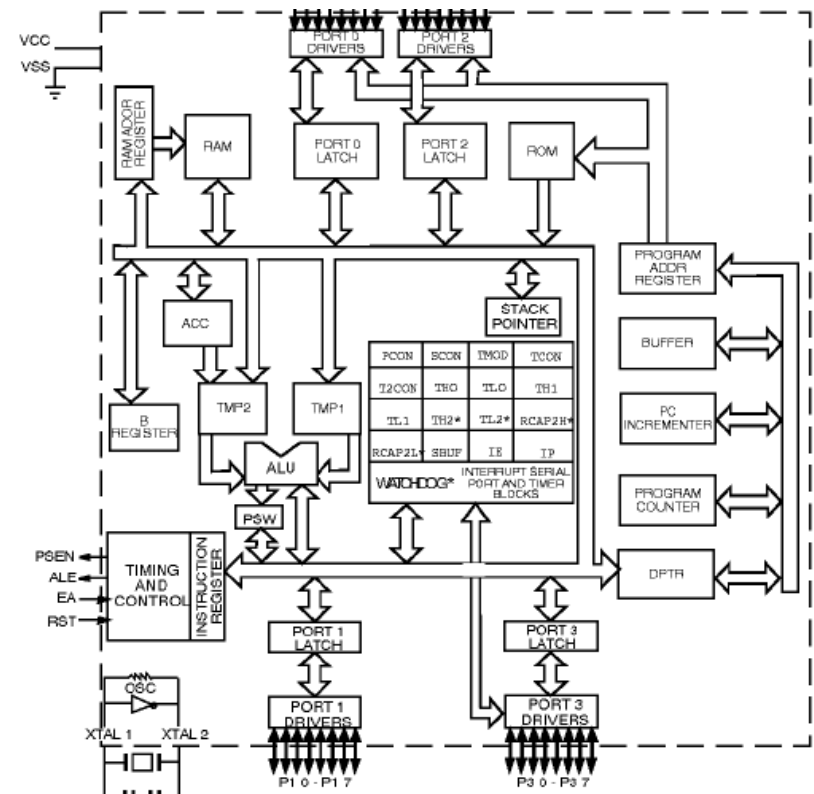**Figure 3 - A diagram showing the basics of multi-lateration [3].**

Figure 4 - The TSK51 8-bit microcontroller[4].

ALU STATUS W
ramdatao
a b wout statusout

CONTROL_UNIT INSTRUCTION FETCH
ROMDATAI
ROMDATAO

RAM

ADDRESS_CONTROL
PC
STACK1
STACKN
FSR

ramdatai
ramaddr
ramwe

ROMADDR
ROMWE
RAI
RBI
RCI
RDI
REI
RFI

RST
POWERUP
DEVICE RESET TIMER — drtclk
CLK

PORTS
PORTA PORTB PORTC PORTD PORTE PORTF
TRISA TRISB TRISC TRISD TRISE TRISF

RAO
RBO
RCO
RDO
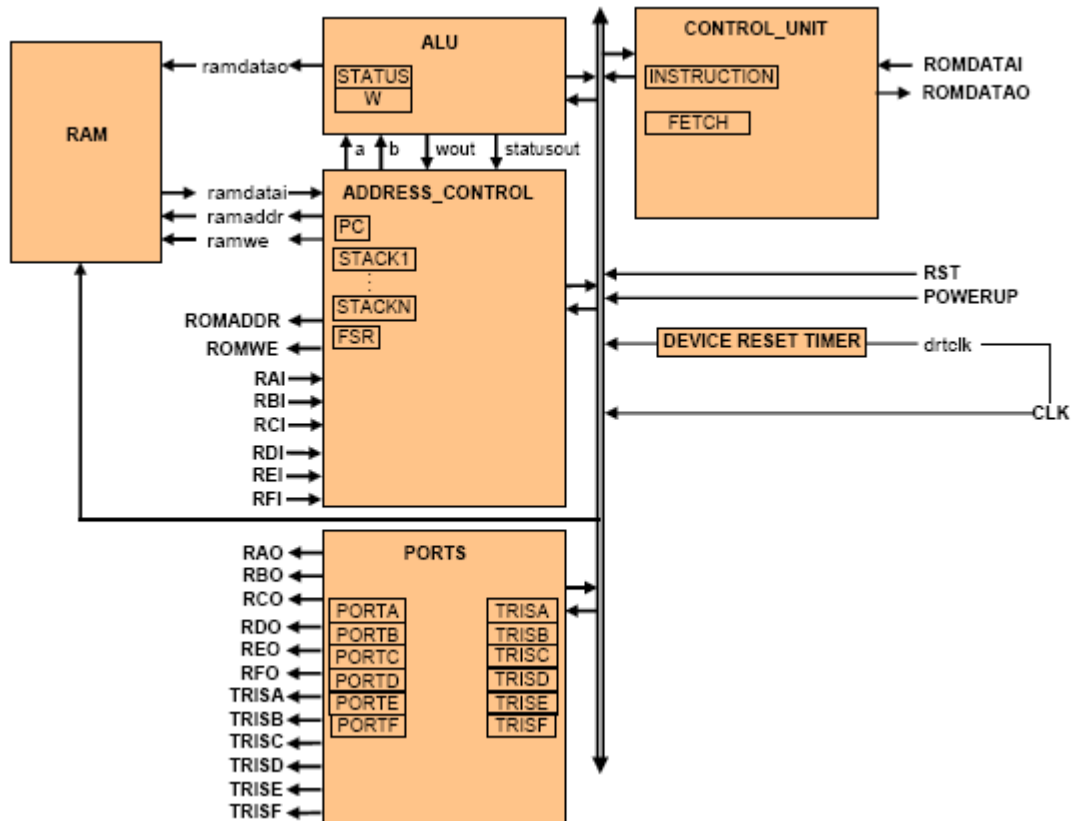REO
RFO
TRISA
TRISB
TRISC
TRISD
TRISE
TRISF

Figure5 - The TSK165 8-bit RISC processor[5].

REFERENCES

[1]   Scanpak, www.scanpak.com
[2]   SFF SDR, Lyrtech, www.lyrtech.com
[3]   SX35, Xilinx, www.xilinx.com
[4]   Boulette Institute of Technology, "Beacon Positioning", Convict Episcopal de Luxembourg, http://www.restena.lu/convict/Jeunes/beacon/Positioning.htm, undated technical paper
[5]   TSK51, Altium, www.altium.com
[6]   TSK165, Altium, www.altium.com
[7]   V652M Mobile Tag Reader, Multispectral Solutions
[8]   M.Reynolds and C. Weigand, "Design Considerations for Embedded Software Defined RFID", Emerging Wireless Technology, 2007
[9]   Altera, Cyclone II Device Family data sheet, http://www.altera.com/literature/hb/cyc2/cyc2_cii5v1_01.pdf

Dr. Andrew Strelzoff
2004 Ph.D. Computer Science, University of California, Santa Barbara
2004 Assistant Professor, Computer Science, University of Southern Mississippi
Areas of Research: FPGA computing, Radio Frequency Identification (RFID)
.