RC64,
A RAD-HARD MANY-CORE HIGH-PERFORMANCE DSP
FOR SPACE APPLICATIONS

Ran Ginosar, Peleg Aviely, Tuvia Liran, Dov Alon,
Alberto Mandler, Fredy Lange, Reuven Dobkin and Miki Goldberg
Ramon-Chips Ltd., 5 HaCarmel Street, Yokneam 2069201, Israel
Email: [ran, peleg, tuvia, dov, tito, fredy, reuven, miki]@ramon-chips.com

Abstract
RC64, a novel rad-hard 64-core signal processing chip targets DSP performance of 75 GMACs (16bit), 150 GOPS and 20 single precision GFLOPS while dissipating less than 10 Watts. RC64 integrates advanced DSP cores with a multi-bank shared memory and a hardware scheduler, also supporting DDR2/3 memory and twelve 2.5 Gbps full duplex high speed serial links using SpaceFibre and other protocols. The programming model employs sequential fine-grain tasks and a separate task map to define task dependencies. RC64 is implemented as a 300 MHz integrated circuit on a 65nm CMOS technology, assembled in hermetically sealed ceramic CCGA624 package and qualified to the highest space standards.

RC64 Progress
In 2012 and 2013 we presented in DASIA early description and progress of the RC64 project [3][4]. Both reports assumed 130nm technology, which imposed many limitations. Those papers described the development of the RC64 architecture, starting with 64 simple RISC cores and discussing preliminary performance of the ESA NGDSP benchmark #5 (modem) on RC64 [1][2].

Meanwhile, the RC64 project has made significant progress. An advanced technology node of 65nm was selected together with a 624-pin ceramic CCGA package. Sixty-four advanced DSP cores were integrated in a simulator and as a FPGA implementation. Both platforms demonstrate several applications. The DSP core tools were used for providing a state-of-the-art integrated development environment for software development and performance analysis.

RC64 Architecture
The new package and new process technology enable the inclusion of twelve integrated full duplex high speed serial links (HSSL) using CML SERDES interfaces on chip at 2.5Gbps rate each, with an aggregate 60Gbps throughput. Several protocols such as SpaceFibre are considered for HSSL, aiming at efficient connectivity among multiple RC64 chips and other FPGAs, ASICs. HSSLS minimize PCB complexity, eliminate the need for multiple external SERDES components on board, save power and reduces the number of pins required. The HSSLS in RC64 support off-board and long range communication, as well as multi-lane channels when higher data rates are required.

The advanced technology and package also enable supporting faster and denser DDR3 SDRAM interface while keeping DDR2 as an option. Reed-Solomon ECC is employed to protect from DDR2/3 SEFI and SEE. The 32-bit wide DDR2/3 interface supports up to 25Gbps throughput.

Other I/O interfaces in RC64 include SpaceWire for control, Parallel LVDS interfaces for advanced ADC and DAC devices connectivity and interface to flash memory.

The on-chip shared memory system of RC64 is based on each core having its own write-through data cache, an instruction cache, and a private store, supporting the unique task-oriented programming (TOP) model. All cores access the single shared memory with 256 ports and a 64-to-256 ports multistage interconnection network, enabling simultaneous access of all processors to shared memory with very little conflicts. Thanks to the data cache, access to shared memory happens either for fetching a complete cache line (the interfaces and the interconnection network are optimized for transferring complete cache lines rather than individual words) or for writing a single word, thanks to the write-through mechanism. While write-through may result in higher traffic rate to memory than write-back, it eliminates the need for complex inter-core cache coordination mechanisms such as snooping, locking and directories. Instead, the programming model minimizes memory conflicts and prevents software from relying on shared memory synchronization.
The on-chip 4MByte shared memory acts as a local-store memory. Access to off-chip DDR2/3 memory is facilitated by software-controlled DMA. This approach simplifies software development and it is found to be very useful for DSP applications, which favor streaming over cache-based access to memory.

The many-core architecture is depicted in Figure 1. A central scheduler assigns tasks to processors. Each processor executes its task from its cache storage, accessing shared memory only when needed. When task execution is done, the processor notifies the scheduler, which can subsequently assign a new task to that processor. Access to off-chip streaming channels, DDR2/3 memory, and other interfaces happens only via programmable DMA channels.

When a single RC64 has insufficient processing power for the application, multiple RC64 chips can be accommodated, as shown in Figure 2. The four chips are interconnected with high-speed channels. RC64 has been designed for integration with tens or hundreds of other RC64 chips, enabling very powerful digital signal processing in space.

Figure 1: RC64 many-core architecture

Figure 2: Single board DSP using 4 interconnected RC64 chips

The architecture of Figure 2 can be extended to a large number of RC64 chips. For instance, highly compute-bound applications such as high-speed digital beam forming while receiving signals from tens of high-speed sensors may require tens of RC64 chips. Another sample application relates to high-performance digital switching of many hundreds of data channels in advanced communication satellites, also calling for a very large number of RC64 chips. A many-chip system is planned with multiple boards, each board containing about 10 RC64 chips with memory and control processors, interconnected by multiple HSSL links to create a tightly-interconnected system. A conceptual example of such a system, including also a multiple ADC and DAC card, is shown in Figure 3.
Figure 3: Multi-card high-performance RC64 system

RC64 Programming Model

The ESA NGDSP benchmark #5, a small portion of a modem [1][2], is depicted in Figure 4. The input stream is separated into even and odd sub-streams. The sub-streams are demodulated by FIR filters, producing I and Q channels. The channels are then up-sampled at a 1-to-L rate, low-pass filtered, and decimated at a M-to-1 rate.

This modem computation is mapped on RC64 to a structure of tasks as shown in the task map in Figure 5. The map comprises an infinite loop. Each loop iteration comprises three concurrent activities. First, the results of demodulation and decimation of the previous iterations are produced on some HSSL channels by the OUT-previous task. Second, a block of new samples for the next iteration is read from some HSSL channels into the on-chip memory by the IN-next task. Third, the present block of samples is processed by the multiple DEMOD_I, DEMOD_Q, DECIM_I and DECIM_Q tasks. Typically, a very large number of parallel copies of these four tasks are created, to exploit potential data parallelism that is inherent in the algorithm. The hardware scheduler issues as many tasks as there are free processors (64 processor cores or fewer), and when some processors complete their tasks they are automatically re-assigned new tasks. The graph in Figure 5 indicates to the schedule that, for instance, all copies of DEMOD_I must be completed before any copy of DECIM_I can be activated.

The code of the computational tasks of Figure 5 is shown (somewhat simplified) in Figure 6. All filter coefficients are pre-loaded. The infinite input stream is divided into large blocks (8,000 elements in the example). Each instance of DEMOD_I (or DEMOD_Q) computes a single element of I (or Q), so there are 4,000 instances of DEMOD_I and 4,000 instances of DEMOD_Q. Similarly, due to up-sampling and decimation, there are 3,200 instances of DECIM_I and 3,200 instances of DECIM_Q. The code accounts for the odd/even and plus/minus structure of the demodulator and for interpolation and decimation in DECIM_I and DECIM_Q.

Figure 4: Computational flow of ESA NGDSP benchmark B5

Figure 5: RC64 task map for benchmark B5
```c
#define NUM_INPUTS 8000
set quota demod_I NUM_INPUTS/2
set quota demod_Q NUM_INPUTS/2
set quota decim_I (NUM_INPUTS/2)*4/5
set quota decim_Q (NUM_INPUTS/2)*4/5
duplicable task demod_I
  // demod_Q is similar
  x = # * 4;
  demod_output_I [(x<<1)+69] =
    - input [x] * demod_coef [14]
    + input [x+2] * demod_coef [12]
    - input [x+4] * demod_coef [10]
    + input [x+6] * demod_coef [8]
    - input [x+8] * demod_coef [6]
    + input [x+10] * demod_coef [4]
    - input [x+12] * demod_coef [2]
    + input [x+14] * demod_coef [0];
demod_output_I [(x<<1)+4+69] =
    + input [x+2] * demod_coef [14]
    - input [x+4] * demod_coef [12]
    + input [x+6] * demod_coef [10]
    - input [x+8] * demod_coef [8]
    + input [x+10] * demod_coef [6]
    + input [x+14] * demod_coef [2]
    - input [x+16] * demod_coef [0];
}
duplicable task decim_I
  // decim_Q is similar
  sum = 0;
i = # * 5 + 4;
  for(t=(i % 4) ; t < 70 ; t +=4) {
    sum += demod_output_I [69-t+i]*decim_coef [t];
  }
decim_output_I [#] = sum;
```

Figure 6: RC64 Processor code example for ESA DSP benchmark B5

All computations are performed on 10-bit fixed point numbers (as specified by the NGDSP benchmark [1][2]) and are executed on single-cycle MAC ALUs in the DSP cores. With four parallel MAC ALUs per core, the B5 benchmark code achieves more than 150 GOPS (Giga operations per second). RC64 is capable of processing input samples of the B5 benchmark at the rate of close to 500 Msps.

This task-oriented programming model (TOP) is designed to relieve the programmer of the need to map computations to processors. The programmer designs tasks that are inspired by the inherent parallelism of the algorithm and data, regardless of how many processors there are and regardless of which processor is active when. Load balancing is achieved automatically by the hardware scheduler, as long as there are many more tasks than processors. Thanks to efficient design of the hardware scheduler, the overhead of switching a processor from one task to another is minimal, taking only a very small number of clock cycles.

**RC64 related FP7 projects**

RC64 program is currently promoted by two FP7 programs.

**QI2S**: Quick Image Interpretation System, [www.qi2s.eu](http://www.qi2s.eu)

The QI2S project develops an ITAR-free technology for quick onboard earth observation, hyper-spectral image interpretation systems. A demonstration platform, including RC64 simulator, FPGA and SW development tools are developed to perform hyperspectral image interpretation algorithms and measure its potential performance in future RC64 based systems. The SW
application include radiometric and atmospheric correction followed by image interpretation and material detection and identification. A mission definition command language will enable fast-turnaround reconfiguration of the application to avoid elaborate time-consuming testing.

**MacSpace**: Many-core space DSP architecture. [www.macspace.eu](http://www.macspace.eu)

MacSpace will research, develop and validate a High-Performance ITAR-Free, ManyCore rad-hard DSP for Computation-Intensive Space Applications. The project include SW applications requirements, on-board system interfaces definitions, customizing a DSP core with its SW development tools and demonstration of the platform and SW applications on a simulator, FPGA and ASIC prototype platforms. The results of the project will affect the architecture of RC64, including its selected DSP IP, interfaces, packaging and SW development environment.

*The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 313105 ([www.QI2S.eu](http://www.QI2S.eu)) and under grant agreement n° 607212 ([www.MACSPACE.eu](http://www.MACSPACE.eu)).*

### References


