A Real-Time Processor For The Danish C-Band SAR

Dall, Jørgen; Jørgensen, Jørn Hjelm; Christensen, Erik Lintz; Madsen, Søren Nørvang

Published in:
International Geoscience and Remote Sensing Symposium

Publication date:
1991

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Dall, J., Jørgensen, J. H., Christensen, E. L., & Madsen, S. N. (1991). A Real-Time Processor For The Danish C-Band SAR. In *International Geoscience and Remote Sensing Symposium: Remote Sensing: Global Monitoring for Earth Management*. (Vol. Volume 1, pp. 279-282). IEEE.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

A REAL-TIME PROCESSOR FOR THE DANISH C-BAND SAR

Jørgen Dall, Jan Hjelm Jørgensen, Erik Lintz Christensen, and Søren Nørvang Madsen

Electromagnetics Institute
Technical University of Denmark
B 348, DK-2800 Lyngby, Denmark
Telephone: 45-42 88 14 44
Fax: 45-45 93 16 34
Telex: 37529 dthdia

ABSTRACT

This paper presents an airborne real-time processor currently under development for the Danish C-band SAR, which has been flown since 1989. The main purpose of the processor is to assist the operator in using the SAR system, but since the imagery produced has a very high quality it can substitute off-line products in many cases. The real-time processor implements the range-Doppler algorithm as a pipeline of elements interconnected by a dedicated data path and a control bus. Basically only three different types of elements are involved: a programmable signal processing element, a multi-purpose memory element and a multi-purpose interface element. The hardware design will be finished in the first half of 1991, and a first reduced version of the processor is expected to be operational early 1992.

KEYWORDS: SAR, DSP, real-time processor, algorithm, architecture, implementation.

INTRODUCTION

In January 1986 the Electromagnetics Institute at the Technical University of Denmark (TUD) launched a remote sensing activity called "Coherent Radar and Advanced Signal processing" (KRAS). The goal is to develop an airborne C-band Synthetic Aperture Radar (SAR) [1]. So far data have been recorded on High Density Digital Tapes (HDDT) and processed in the laboratory using a general-purpose computer, but now an airborne Real-Time Processor (RTP) is under development.

The Danish SAR is a strip map SAR with a very high degree of flexibility achieved through extensive use of digital electronics. In particular both the signal generation and the signal processing are digital.

Part of the signal processing is already implemented in real time. The SAR sensor includes a Digital Pre-Processor (DPP) performing Doppler centroid estimation, first order motion compensation and azimuth pre-filtering.

The RTP is intended to serve many purposes. Firstly, it should assist the operator in using the SAR system (error detection, in-flight target inspection, etc.). Secondly, since the imagery provided by the RTP is of a high quality, and since it can be recorded in a digital form, it can often substitute imagery produced off-line by ground based processors. Finally, the RTP can be used on the ground in an off-line mode too. This is convenient if the processing has to be repeated after the flight in order to take into account calibration data for instance, or if the onboard processed data were not recorded.

The remainder of this paper describes the requirements to the RTP and presents the SAR processing algorithm adopted. The implementation is discussed in terms of the overall architecture, the control system, and the most important of the building blocks from which the processor is constructed. Finally a few concluding remarks are given.

PROCESSOR REQUIREMENTS

Table 1 lists the KRAS system parameters of relevance to the processor.

Frequency	5.3 GHz
Platform speed	≤ 300 m/s
Pulse repetition frequency	≤ 800 or ≤ 1600 Hz
Pre-filter ratio	4 or 8
Pulse length	from 0.64 to 20 μs
Bandwidth	≤ 100 MHz
Azimuth resolution	variable 2, 4, 8 m
Range resolution	variable 2, 4, 8 m
Slant range swath	variable 12, 24, 48 km
Range	≤ 80 km
Effective PRF	≤ 50, ≤ 100 or ≤ 200 Hz
Azimuth sample spacing	variable 1.5, 3.0, 6.0 m
Range sample spacing	variable 1.5, 3.0, 6.0 m
Range samples	8192
Pulse length	≤ 2048 samples
Doppler history length	≤ 1024 samples
Range curvature	≤ 2.4 samples
Depth of focus	32 samples

Table 1. System parameters, basic and derived.

The RTP is able to process the entire swath into full resolution imagery in real time. In order to attain this resolution, a comprehensive motion compensation is required and therefore the RTP complements the first order compensation performed by the digital pre-processor. The motion compensation is based on data from the Inertial Navigation System (INS) mounted next to the antenna.

Only one channel can be processed, but any of the four channels in the future polarimetric system can be chosen as input for the real-time processor.

Due to the three axis stabilization of the antenna which is controlled by the INS and the Doppler estimator, the processing assumes a nominally zero Doppler centroid. The motion compensation corrects for the undesired transversal aircraft displacements, and the azimuth processing only corrects for the remaining range curvature.

If desired the imagery processed by the RTP can be radiometrically corrected. The correction takes into account the range attenuation and the non-uniform antenna elevation pattern. In order to correct for the latter a flat terrain is assumed. Also data from the internal calibration of the SAR can be incorporated in the radiometric correction.

The RTP output can be presented either in the slant range projection or in the ground range projection. A flat terrain is assumed here too. An optional multi-looking can be accomplished by spatially averaging the full resolution imagery after the square law detection.

The RTP is supplied with aircraft position data originating from the INS or a GPS receiver. These navigation data are synchronized with the radar data so that the position of a pixel pointed out in the processed image can be given together with the associated backscatter coefficient.

It is possible to record the processed data either before or after the detection and in either the slant range or the ground range projection. In the present single polarization system, recording the (possibly complex) data focused by the RTP does not prevent the raw pre-filtered data from being recorded simul-

taneously. The HDDT has the capacity to record both, and the HDDT interface developed is able to merge the two data streams.

ALGORITHM

The RTP is based on the range-Doppler algorithm. Other algorithms have been considered, but they have been found less suitable. The RTP includes a range processor, an azimuth processor and a post processor as shown in Fig. 1. Basically the range and azimuth processors perform a matched filtering implemented in the frequency domain via the fast Fourier transform (FFT) and its inverse (IFFT). This approach is preferred to a time domain approach because of the long filters involved, cf. Table 1, and because it enables the Hybrid Algorithm [2] to be used for the range curvature correction. In azimuth the overlap-save block convolution algorithm is applied with an overlap of 50 per cent. With a Doppler history length up to 1024, this gives a maximum block length of 2048.

The motion compensation is split into a range migration correction and a phase correction [1]. In turn the latter is split into a first order compensation (part of the digital pre-processor), a local phase shift and, if needed, a global phase shift, see Fig 1. A right hand coordinate system is defined with the x-axis in the direction of the desired track (straight line or great circle) and the z-axis pointing downwards. Then a left looking SAR displaced from the desired track by Δ_y and Δ_z , measures a range exceeding the desired range, R , by

$$\Delta R \approx \Delta_y \sin \theta - \Delta_z \cos \theta, \quad \theta = \text{Arccos}(H_{\text{nom}}/R) \quad (1)$$

H_{nom} is the nominal altitude, and θ is the incidence angle as seen from the desired track. The range shift, ΔR , varies over the swath, so it cannot be eliminated by multiplying the range spectrum by a phase ramp. Instead the RTP implements the range shift using a cubic interpolator.

The phase shift associated with the displacement from the desired track is given by $-4\pi\Delta R/\lambda$. The first order motion compensation applies a range independent phase shift corresponding to the value of ΔR at

Fig. 1. RTP algorithm.

the middle of the swath. This phase shift, which ensures that the Doppler spectrum is centred within the pass-band of the azimuth pre-filter, is subtracted in the subsequent motion compensation.

For large scenes as those dealt with in real time strip map processing, Δ_y and Δ_z may eventually grow quite large, and hence the correction phase may change appreciably from range sample to range sample. The change is π for $\Delta_y = \Delta_z = 120$ m, $H_{nom} = 12500$ m and $\theta = 45^\circ$, for instance. This is undesirable because it complicates the processing where azimuth signals from neighbouring range gates are combined, e.g. the range curvature correction and the ground range projection (cubic interpolation).

In the RTP the problem is solved by defining a local track for each block in the azimuth processing. This local track is a straight line parallel to the desired track, but displaced so that it intercepts the actual track at the midpoint. Referring to the local track, Δ_y and Δ_z are small, and so the phase shift does not cause problems to the remaining processing. Unfortunately the "local phase shift" in Fig. 1 cannot be combined with the range shift (which still refers to the desired track) because of the block overlap. This means that each pixel is phase corrected twice. If phase continuity is crucial, the discontinuities at the edges of adjacent blocks referring to different local tracks can be eliminated. This is accomplished by the "global phase shift" module which of course must follow the critical ground range projection.

ARCHITECTURE

The hardware architecture of the real-time processor (RTP) is essentially a direct mapping of the algorithm structure (Fig. 1.) into hardware. This leads to a simple pipeline structure of elements where the main design objective is to minimize the number of different element types. This results in the definition of three element types, a programmable signal processing element (PSPE), a multi-purpose memory element and a multi-purpose interface element. Table 2 defines the three elements.

Element type	Functions
PSPE	Signal processing: FFT/IFFT, interpolation, detection etc.
Memory Element	Corner turning functions & display image buffer.
Interface Element	Data path interface: DPP->RTP, RTP->HDDT, RTP-unit->RTP-unit.

Table 2. Real-time processor elements.

By restricting the system to these three element types a modular and cost effective design is ensured. The flexibility gained by using a modular design also prepares the system for future enhancements.

A dedicated data path connects the elements of the pipeline. The data path is simply a unidirectional connection over which data are transferred between elements. The data path includes a 24 bit data bus and various control signals. Data words are transferred synchronously between elements using a clock

generated by the transmitting element. The clock is nominally 8 MHz. Data are transferred in blocks, the length of which are restricted to multiples of 1024 words and a maximum 16384 words. Block transfer is controlled by a handshake mechanism between the transmitting and the receiving element. The handshake lets the data ripple through the system whenever the elements are ready, thereby allowing the processing to proceed at the fastest possible speed. When the RTP is operating as an on-line real-time processing facility the speed is limited by the system pulse repetition frequency, however when used as an off-line processing facility, where the input data rate can be greater than real-time, a speed advantage will be found.

The data path is augmented by an intelligent buffer scheme incorporated into all elements. Each element has an input buffer, an output buffer and a dedicated buffer controller. The buffer controller handles the transfer of data between the data path and the buffers. Each buffer contains two memory blocks of 16384 words (24 bit). Access to each of the memory blocks can be switched between the data path and the element interior. This type of buffer is often referred a double buffer.

The employment of double buffers relieves the elements from the tedious task of performing trivial I/O operations on the data path. Double buffering especially improves the processing capability of the PSPE. Another advantage of double buffering is that it allows the synchronous data transfer rate on the data path to be minimized thereby simplifying the data path design and improving reliability.

The buffering will increase the RTP system latency, but this increase is negligible compared with the inherent latency caused by the large synthetic aperture. Buffering will also give rise to an increase in circuitry but this is considered to be a small price for the above-mentioned benefits.

A 24 bit integer word format is chosen for two reasons. Firstly, off-line processing of SAR data has shown that 16 bit precision is adequate, however this requires a careful and time consuming control of the gain throughout the processing in order to avoid arithmetic over/underflow. Using a 24 bit word format allows the requirements for gain control to be loosened thereby simplifying the processing. Secondly, the PSPE is based on a digital signal processor (Motorola DSP56001) offering a 24 bit word format.

CONTROL

The control of the RTP is based on the VME bus. All elements include an intelligent VME slave interface based on the Motorola MC68302 16 bit microprocessor and Performance Technologies VME Slave Interface chip. Among the facilities supported are interrupts, mailbox registers, dualport memory, and dynamic memory mapping.

The RTP is built into standard 19" card cages allowing 21 single-width boards per cage. The real-time processor will initially require 2 cages. This leaves enough surplus space to allow for the planned future enhancements. Each cage is known as an RTP unit.

Each unit contains an off-the-shelf general-purpose VME based computer board used to control the operation of the RTP elements. This computer is denoted the RTP controller. The RTP controller performs the initial setup of the RTP elements and the tasks associated with the operation of the RTP such as the distribution of auxiliary data associated with motion compensation, data collection, image display control, processing gain control and error handling.

The problem of synchronizing the auxiliary motion compensation data with the radar data at the PSPEs that are performing motion compensation processing tasks is solved by using a simple FIFO synchronization technique. Auxiliary data and radar data are treated as two separate data streams. The data path is viewed as a FIFO storage device for the radar data. The intelligent VME slave interface on the PSPE uses local memory to establish a FIFO memory structure for the auxiliary data. Both FIFO structures are empty before processing begins. The consumption of data from the two FIFO structures proceeds in a locked and synchronized fashion when the processing commences.

PSPE

The programmable signal processing element (PSPE) performs all the signal processing tasks required in the RTP. The variety of different signal processing tasks requires the PSPE to be very flexible which is why the PSPE design is based on an array of standard digital signal processors. An assessment of available digital signal processors resulted in the use of the Motorola DSP56001 (27MHz) due to various performance and cost considerations.

The signal processing tasks of the RTP differ greatly in respect to the amount of required computations compared with the amount of required I/O operations to be performed. The PSPE design takes into account that some tasks are highly compute bound (FFT/IFFT) while others are I/O bound (detection, interpolation etc.). Figure 2 shows the resulting PSPE design.

Fig. 2. PSPE block diagram.

The design uses 8 digital signal processors organized in two identical arrays. A double buffer is used to separate the two arrays. Furthermore, each array has two busses in order to increase I/O capability.

The signal processors are connected via the host interface to the MC68302 microprocessor in the VME interface. The MC68302 is used to initiate and control the operation of the signal processors.

The utilisation of the digital signal processors varies with the processing task type. Highly compute bound tasks offer 100% (8/8) processor utilisation while I/O bound tasks only offer 25% (2/8). Many of the PSPEs performing I/O bound tasks can however utilize the idle processors as co-processors thereby reaching a high degree of utilizational efficiency. Also, in some cases it is possible to combine compute bound and I/O bound signal processing tasks thereby increasing the overall processor utilization.

A special data path bypass feature is included on the PSPE (Fig. 2). The bypass feature allows data to be passed from the data path input to the data path output of the PSPE without involving the digital signal processors. The bypass feature allows PSPEs to work together in a parallel fashion when performing a task that is too big for a single PSPE and not practically distributive on a pipelined connection of PSPE's. The azimuth FFT and IFFT are such tasks. Since the PSPEs are physically situated in a pipeline but logically working in parallel, this is called "pseudo" parallel operation. Figure 3 illustrates the pseudo parallel operation.

Fig. 3. Pseudo parallel PSPE operation.

Finally, the bypass feature can also be used to enhance RTP system fault tolerance by including redundant PSPEs and using the bypass feature to switch PSPEs into and out of the data path.

CONCLUDING REMARKS

A real-time processor for the Danish high resolution SAR has been presented in terms of its functional performance, algorithm, architecture and implementation. The hardware design will be finished in the first half of 1991. The first version of the processor is expected to be operational early 1992, but it will not include comprehensive software for the motion compensation, the radiometric correction and the image presentation. The completed system will also include hardware and software enabling the processor to be used in a ground based off-line mode offering polarimetric and multi-look processing as well.

REFERENCES

- [1] S.N. Madsen, E.L. Christensen, N. Skou, J. Dall, "The Danish SAR System; Design and Initial Tests", Accepted for IEEE Trans. on Geoscience and Remote Sensing.
- [2] C. Wu, K.Y. Liu and M. Jin, "Modeling and a Correlation Algorithm for Spaceborne SAR Signals", IEEE Trans., Vol. AES-18, No. 5, Sep. 1982.