

DSP를 사용한 위상정합 레이더용 시간 영역 디지털 펄스압축기 운용특성 분석

임중수, 박영철
백석대학교

jslim@bu.ac.kr, ycpark@bu.ac.kr

An Analysis of Operating Characteristics for Digital Pulse Compressor of Coherent Radar in Time Domain

Joong Soo Lim, Young Chul Park
Baekseok University

요 약

본 논문에서는 DSP를 이용하여 코히어런트 레이더에 사용될 수 있는 확장성이 용이한 디지털 펄스 압축기를 시간영역에서 설계하고 그 내용을 분석하였다. 수신 거리 셀 및 FIR 필터 탭(tap) 수에 따른 펄스압축 연산시간을 ADSP21060을 사용하여 분석함으로써, 설계하고자하는 레이더 시스템 요구 파라미터 들이 정해지는 경우 펄스압축기 구성에 소요되는 DSP의 수행 기능과 소요 수를 쉽게 예측할 수 있도록 운용특성을 분석하였다.

1. 서론

펄스압축(PC : pulse compression) 기법은 탐색 (surveillance) 및 추적 레이더에 주로 사용되는 기법으로 신호의 대역폭을 넓게 하는 주파수 또는 위상변조를 신호에 가함으로써 구현된다. 이러한 기능의 펄스압축 처리는 디지털 신호처리 분야의 급속한 발전과 더불어 최근에는 디지털 방식에 의해 구현되며, 아날로그 방식에 비해 큰 압축 비를 쉽게 구현할 수 있는 장점이 있다.^[1,2]

디지털 펄스압축 처리는 FIR(finite impulse response) 필터 알고리즘을 이용한 시간영역 또는 FFT(fast Fourier transform) 알고리즘을 이용한 주파수 영역에서 구현된다. FFT에 의한 주파수영역 연산은 반드시 거리 셀(N_R) 전체가 수신된 후 이를 블록(block) 단위로 처리해야 하지만, FIR 필터 알고리즘에 의한 시간영역 연산은 하드웨어 구성에 따라서는 거리 셀 데이터를 수신하고 있는 중에도 필터 연산을 시작할 수 있는 유리한 점이 있다.^[3]

본 논문에서는 DSP를 이용하여 확장성이 용이한 디지털 펄스압축기를 시간영역에서 구성하였다. 아날로그 디바이스(Analog Device)사의 ADSP-21060을 사용하여 C-언어와 어셈블리 언어에 의한 펄스 압축 연산시간을 분석함으로써, 시스템 파라미터가 정해지는 경우 다중 DSP 구조의 펄스압축 처리기 구성에 소요되는 DSP 수를 쉽게 예측할 수 있도록 하였다. 그리고 이러한 기술의 적용 가능성을 탐색 레이더를 대상으로 확인하였다.

2. 디지털 펄스 압축기 구성

레이더의 송신 파형을 $s(i)$, $i=0,1,\dots,N-1$ 와 같이 표현하는 경우, 이 신호에 대한 정합필터의 임펄스(impulse) 응답 $h(i)$ 는

$$h(i) = s^*(N-i-1), \quad i=0,1,2, \dots, N-1 \quad (1)$$

과 같이 주어진다. 식에서 *는 복소공액(complex conjugate)을 의미하며, N은 송신 파형에 대한 디지털

털 탭 수이다.

시간영역에서의 펄스압축 관계는 수신신호와 임펄스 응답 $h(i)$ 의 상승적(convolution) 임으로 식 (2)와 같이 표현될 수 있다.

$$y(k) = \sum_{i=0}^{N_T-1} x(k+i) h(i)$$

$$= \sum_{i=0}^{N_T-1} x(k+i) s^*(N-i),$$

$$k=0,1,2, \dots, N_R-1. \quad (2)$$

식에서 $y(k)$ 는 펄스압축기의 k 번째 출력신호, $x(i)$ 는 i 번째 수신신호, $h(i)$ 는 i 번째 임펄스 응답, N_T 은 FIR 필터 탭 수, N_R 은 수신신호의 거리 셀 수를 각각 의미한다.

3. 디지털 펄스 압축기 설계

디지털 펄스압축기 구현방식은 하드웨어 측면에서 보면크게 DSP를 이용하는 방식과 FIR 필터 또는 ASIC (application specific integrated circuit)과 같은 전용기능의 칩(chip)을 이용하는 2가지로 구분될 수 있다. DSP를 이용하는 경우에는 입력데이터가 연산 수행에 앞서 원하는 메모리로 분배되어야 한다.^[6] 따라서 펄스압축처리의 데이터 저장/분배/연산 타이밍(timing)은 그림 1과 같이 일반화 할 수 있다. 그림 1에 나타낸 데이터 저장 및 분배시간은 하드웨어 구조에 따라 무시될 수 있으므로, 이 경우에는 연산이 데이터 저장과 거의 동시에 수행 된다.

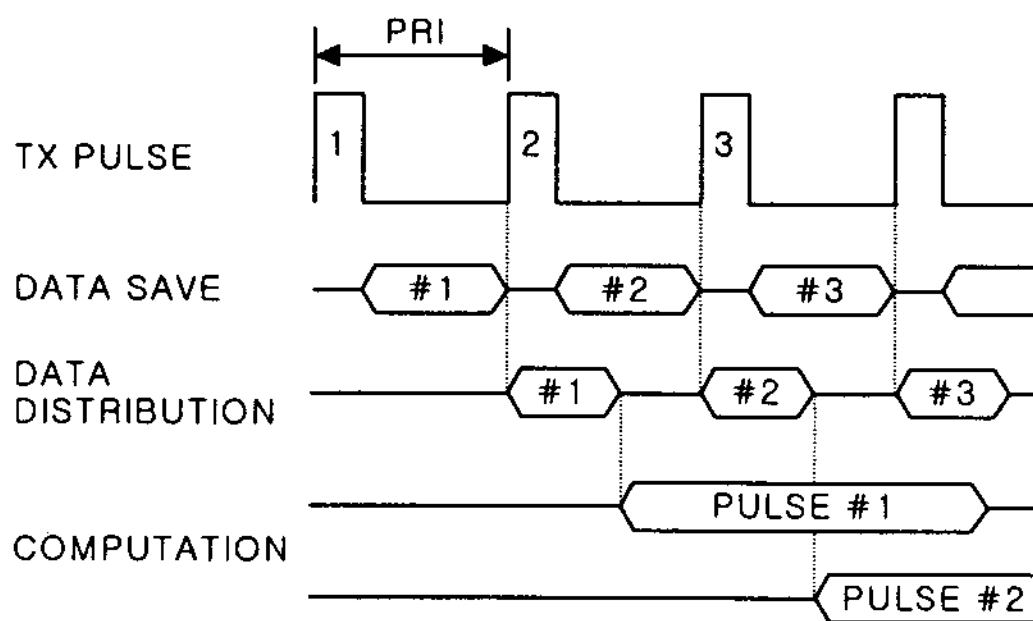


그림 1. 디지털 펄스압축기 연산 타이밍

위상정합 펄스 누적 레이더는 CPI(coherent processing interval) 내의 N_{CPI} 개의 펄스를 기본단위로 처리가 이루어진다. 또한 DSP를 이용하여 구

현되는 경우, 분배+연산시간이 일반적으로 PRI (pulse repetition interval)보다 길기 때문에 식(3)이 만족된다.

$$T_{pc} < N_{CPI} \cdot PRI \quad (3)$$

식 (3)에서 T_{PC} 는 한 개 펄스에 대한 펄스압축 연산시간이다. 결국 여러 개의 DSP를 이용하여야만 연산지연을 갖는 실시간 펄스압축기의 구현이 가능하며, 이를 구현하기 위하여 그림 2와 같이 N개의 DSP를 이용한 다중 DSP 구조의 디지털 펄스압축기를 구성한다.

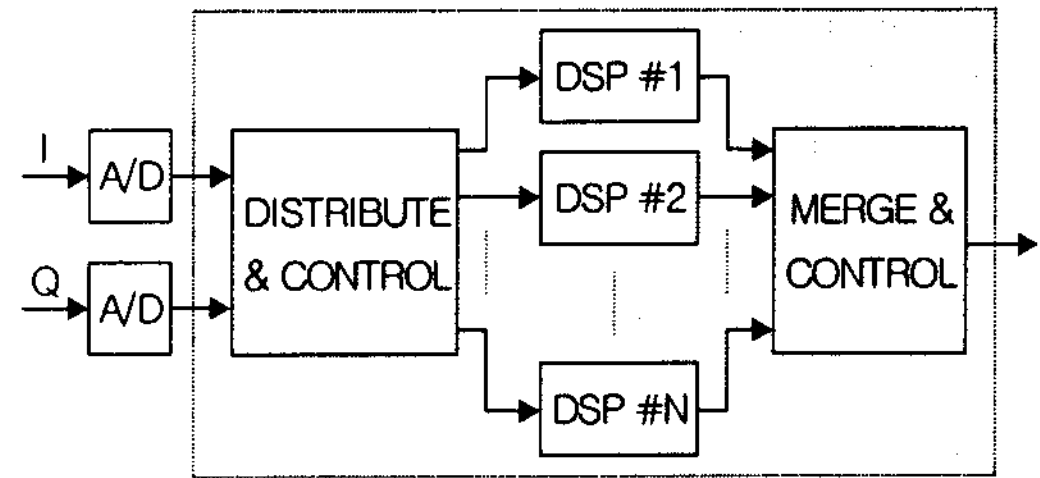


그림 2. 디지털 펄스압축기의 설계

그림 2에서 데이터 분배 및 콘트롤 회로가 A/D 보드로부터 입력되는 데이터를 첫 번째 DSP에서부터 순환(circulation)하여 분배한다면, 데이터 분배 및 펄스압축에 소요되는 연산시간이 정해지는 경우 시스템 구성에 필요한 DSP 수를 쉽게 예측할 수 있다. 즉, 시스템 구성에 소요되는 DSP 수 N_{DSP} 는 식 (4) 과 같다.

$$N_{DSP} = \frac{T_D + T_{PC}}{PRI - T_X} \quad (4)$$

식 (4)에서 T_D 은 데이터 저장 후 DSP 메모리로의 데이터 분배시간, T_{PC} 는 펄스압축 연산시간, PRI(pulse repetition interval)는 펄스반복시간, T_X 는 송신 펄스폭을 나타낸다. 일반적으로 MTI (moving target indicator) 탐색 레이더에 있어서는 $PRI \gg T_X$ 및 $T_{PC} \gg T_D$ 의 관계가 성립함으로 식 (4)은 식(5)와 같이 근사화하여 사용될 수 있다.

$$N_{DSP} \approx \frac{T_{PC}}{PRI} \quad (5)$$

식 (4)이 만족되지 않는 경우, 다중 DSP가 사용되어야 한다. 따라서 펄스압축 연산에 의한 지연시간을 줄이기 위해서는, 그림 2와 같이 다수의 DSP를 사용하여 거리 셀 데이터를 중첩처리 함으로써 DSP 상호간 데이터교환 없이 출력신호의 연속성을 유지할 수 있다.

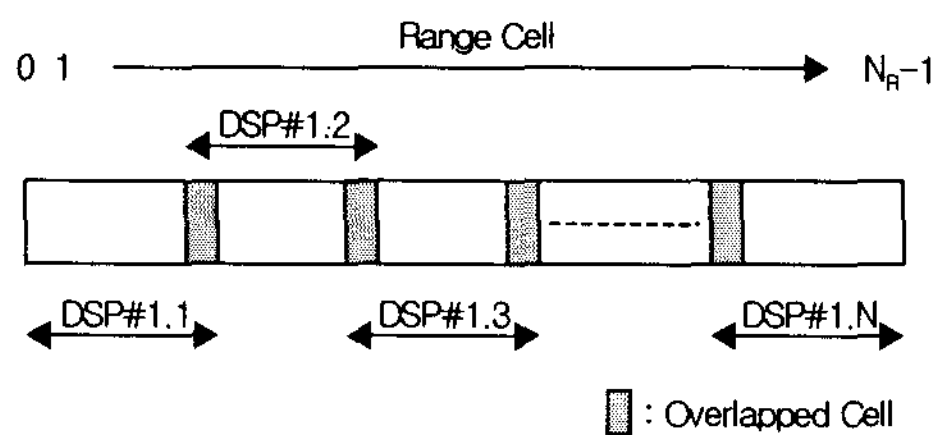


그림 3. 중첩처리를 위한 데이터 분배

4. 펄스압축 연산 시간

DSP를 이용하여 시스템이 요구하는 고속연산을 수행하는 경우, 운용 프로그램의 일반적인 제어(control) 부분은 C-언어와 같은 상위(high level) 언어로 개발되지만, 속도가 요구되는 주요 알고리즘은 어셈블리어언어를 사용하여 개발하는 것이 일반적이다.^[1] 본 절에서는 C-언어와 ADSP21060 어셈블리어언어에 대한 펄스압축 연산속도를 비교·분석하였다.

본 논문에서 사용한 아날로그 디바이스(Analog Device)사의 ADSP21060은 한 사이클에 3개까지의 다중연산을 수행할 수 있으므로 최대 120 MFLOPS의 부동소수점 연산을 수행할 수 있다. 특히 프로그램 및 데이터를 기억할 수 있는 프로그램 메모리와, 데이터만을 기억할 수 있는 데이터 메모리가 독립적으로 구성되어 있다. 따라서 데이터를 프로그램 메모리와 데이터 메모리에 분리하여 저장함으로써 한 사이클에서 2개의 데이터를 동시에 읽을 수 있는 장점이 있다.^[4]

프로그램 개발 및 디버깅은 아날로그 디바이스사의 ADI Visual DSP 4.1 환경 하에서 이루어졌다. 필터 계수 값 및 데이터는 한 사이클에서 2개의 데이터를 동시에 읽어올 수 있도록 프로그램 메모리 및 데이터 메모리에 각각 저장되었다.

표 1은 식 (2)의 펄스압축 연산수행을 위해, C-언어로 작성된 운용 프로그램에서 C언어로 된 펄스압축연산을 함수를 호출한 경우 거리 셀 및 FIR 필터 탭 수에 따른 소요 사이클 수를 나타내고 있으며, C-언어를 사용한 경우에는 최적화(optimization)

선택사항을 적용하였다. 표 1의 결과에서 첫 번째 항 52 및 26은 전체 사이클 수에 비해 매우 작기 때문에 C-언어 대 어셈블리어 언어 연산속도 비 R_T 는 식 (6)과 같고, 그림 4는 식(6)의 연산속도 비를 나타낸 결과로서 펄스압축 탭 수가 증가함에 따라 약 3배의 연산속도 차이를 확인 할 수 있다.

$$R_T \approx \frac{9 + 12 * N_T}{11 + 4 * (N_T - 1)} \quad (6)$$

표 1. 펄스압축 사이클 수.

언어	소요 사이클 수 (25ns/cycle)
C	$52 + (N_R - N_T + 1) * (9 + 12 * N_T)$
어셈블리	$26 + (N_R - N_T + 1) * \{11 + 4 * (N_T - 1)\}$

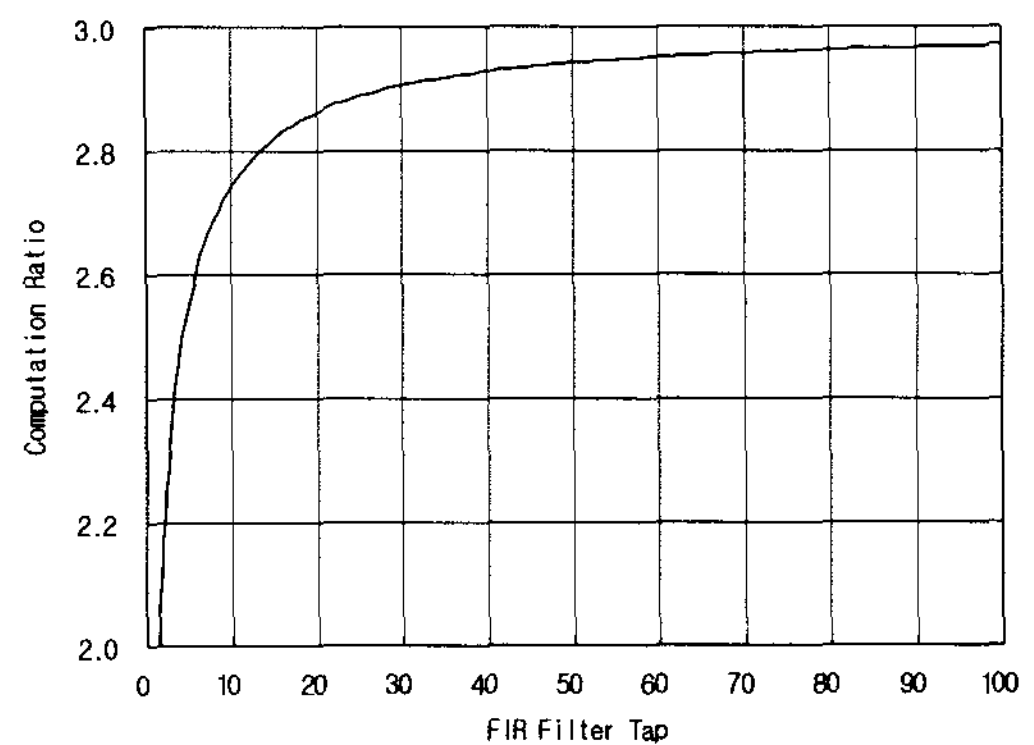


그림 4. C-언어 및 어셈블리어언어에 대한 펄스압축 연산시간 비.

5. 결론

본 논문에서는 DSP를 이용하여 시간영역에서 확장성이 용이한 디지털 펄스압축 처리기를 구성하였다. 수신거리 셀 및 FIR 필터 탭 수에 따른 펄스압축 연산시간을 아날로그 디바이스사의 ADSP 21060을 대상으로 C-언어와 어셈블리어언어로 비교/분석하였다. 본 논문에서 적용한 다중 DSP 펄스압축기 구성방법은 DSP 한 개를 사용한 경우의 펄스압축 연산시간만 정확히 예측할 수 있으면, DSP 종류에 관계없이 펄스 압축기 설계에 적용할 수 있다.

참고문헌

[1] M.I. Skolnik, *Introductio to Radar Systems*, McGraw-Hill, Inc., 1980.

- [2] Ren Peihong, "A Kind of High-Speed Real-Time Digital Pulse Compressor Implemented in Time Domain," *Radar, 2001 CIE International Conference on, Proceedings*, pp. 966~969, 2001.
- [3] Teng Long and Erke Mao, "High-Speed Real-Time Digital Radar Processor : Design and Implementation," *Radar, 2001 CIE International Conference on, Proceedings*, pp. 987~991, 2001.
- [4] "ADSP-2106X SHARC™ User's Manual," *Analog Devices, Inc.*, 2nd Edition, 1997.