

技術論文

DOI: <http://dx.doi.org/10.5139/JKSAS.2012.40.9.827>

T-방식과 M-방식을 이용한 저궤도위성용 반작용 휠의 속도측정보드 설계

이재녕*, 박성훈*, 허수진*, 이윤기**

The Implementation of the Speed Measurement Board for the Reaction Wheel on the LEO Satellite using the T, M-Method

Jae-Nyeung Lee*, Sung-Hun Park*, Su-Jin Heu* and Yun-Ki Lee**

ABSTRACT

In this paper, we will design the speed measurement board of LEO Satellite's reaction wheel which has two speed measuring methods as M-Method type and T-Method type. therefore we can use the advantage of two methods. and we will verify the availability of design on the on-board computer at the real LEO Satellite(KOMPSAT-3). In the reaction wheels satellite that can change the satellite's attitude is one of the leading drivers by the rotational inertia of the motor will perform attitude control. Reaction methods for detecting wheel rotation speed generated during a certain period T internal reaction wheel tacho pulse counting M-Method to detect wheel speed and wheel tacho pulses are generated by measuring the time between the detection rate can be divided into T-Method. M-method is simple to implement and benefit measurement time is constant, but slow fall in the velocity measurement accuracy is a disadvantage. In contrast, the time between tacho pulses to measure the T-Method to measure the precise speed at low speed and to measure the time delay is small, has the advantage. However, this method also in the actual implementation and the complexity of the operation at different speeds depending on the speed of operation has the disadvantage.

초 록

본 논문에서는 저궤도 위성의 반작용 휠 속도 측정을 위해 M-방식과 T-방식 모두 검출 가능한 보드를 설계하여 두 방식의 장점을 이용할 수 있도록 하고, 실제 위성탑재컴퓨터에 장착하여 시험을 수행하고 이를 통하여 그 설계의 유용성을 검증한다. 위성에서 반작용 휠은 위성의 자세를 변화시킬 수 있는 대표적인 구동기의 하나로 반작용 휠 구동모터의 회전에 의해 발생하는 회전관성에 의해 자세 제어를 수행하게 된다. 반작용 휠의 회전 속도를 검출하는 방법으로는 일정한 주기 T 동안 발생된 반작용 휠 내부 타코 펄스를 세어 휠의 속도를 검출하는 M-방식과 휠에서 발생하는 타코 펄스들 간의 시간을 측정해서 속도를 검출하는 T-방식으로 나눌 수 있다. M-방식은 구현이 간단하고 측정 시간이 일정하다는 장점이 있으나, 저속에서는 속도 측정 분해능이 떨어진다는 단점이 있다. 그에 반해, 타코 펄스간의 시간을 측정하는 T-방식은 저속에서 정밀한 속도를 측정할 수 있으며 측정에 따른 시간 지연이 작다는 장점이 있다. 그러나 이 방법 역시 실제 구현에 있어서 연산이 복잡하고 속도에 따라 연산 속도가 달라진다는 단점이 있다.

† 2012년 3월 13일 접수 ~ 2012년 8월 20일 심사완료

* 정회원, 한국항공우주산업

** 정회원, 한국항공우주연구원

교신저자 E-mail: ca7ch_22@koreaaero.com

경남 사천시 사남면 유천리 802번지

Key Words : LEO Satellite(저궤도 위성), Reaction Wheel (반작용 휠), T-Method (T-방식), M-Method(M-방식), Tacho Pulse(타코 펄스), Satellite On-board Computer(위성탑재 컴퓨터)

I. 서 론

위성에서 반작용 휠은 위성의 자세를 변화시킬 수 있는 대표적인 구동기의 하나로 반작용 휠 구동모터의 회전에 의해 발생하는 회전관성에 의해 자세 제어를 수행하게 된다. 반작용 휠의 회전 속도를 검출하는 방법으로는 일정한 주기 T 동안 발생한 반작용 휠 내부 타코 펄스를 세어 휠의 속도를 검출하는 M-방식과 휠에서 발생하는 타코 펄스들 간의 시간을 측정해서 속도를 검출하는 T-방식으로 나눌 수 있다. M-방식은 구현이 간단하고 측정 시간이 일정하다는 장점이 있으나, 저속에서는 속도 측정 분해능이 떨어진다는 단점이 있다. 그에 반해, 타코 펄스간의 시간을 측정하는 T-방식은 저속에서 정밀한 속도를 측정할 수 있으며 측정에 따른 시간 지연이 작다는 장점이 있다. 그러나 이 방법 역시 실제 구현에 있어서 연산이 복잡하고 속도에 따라 연산 속도가 달라진다는 단점이 있다.

본 논문에서는 저궤도 위성의 반작용 휠 속도 측정을 위해 M-방식과 T-방식 모두 검출 가능한 보드를 설계하여 두 방식의 장점을 이용할 수 있도록 하고, 실제 위성탑재컴퓨터에 장착하여 시험 및 검증을 수행하고자 한다.

II. 본 론

2.1 반작용 휠의 속도 측정 방법

반작용 휠의 속도를 측정하는 방법은 일정시간 동안 타코 펄스를 세는 M-방식과 두 타코 펄스간의 시간 간격을 측정하는 T-방식으로 나눌 수 있다.

2.1.1 M-방식을 이용한 속도 측정

반작용 휠의 속도를 측정하는 방법은 일정시간 동안 타코 펄스를 세는 M-방식과 두 타코 펄스간의 시간 간격을 측정하는 T-방식으로 나눌 수 있다.

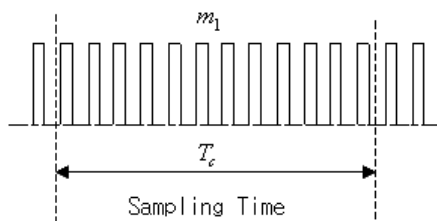


Fig. 1. M-방식

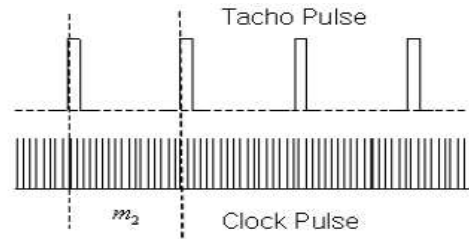


Fig. 2. T-방식

이 방식의 기본 원리는 Fig. 1에서 보는 바와 같이, 일정한 샘플링 시간 T 마다 발생하는 타코 펄스의 개수를 세어 전동기의 회전 속도를 측정한다. 측정되는 반작용 휠의 속도를 w , 한 바퀴 회전 시 발생하는 펄스의 개수를 P , 측정 주기를 T_c , 측정주기 동안 발생하는 펄스의 개수를 m_1 이라 하면, 반작용 휠의 속도는 아래 식(1)과 같이 주어진다.

$$w = \frac{60 \cdot m_1}{P \cdot T_c} (rpm) \quad (1)$$

2.1.2 T-방식을 이용한 속도 측정

이 방식은 타코 펄스 간의 시간 간격을 측정하는 것으로, 펄스와 펄스 사이의 간격에 $f_c(\text{Hz})$ 의 클럭 펄스가 m_2 개 들어간다고 했을 때, 측정되는 반작용 휠의 속도를 w , 한 바퀴회전 시 발생하는 펄스의 개수를 P 라하면, 반작용 휠의 속도는 식(2)와 같이 주어진다.

$$w = \frac{60 \cdot f_c}{P \cdot m_2} (rpm) \quad (2)$$

2.1.3 M-방식의 분해능

M-방식의 분해능은 측정 주기 동안 발생하는 펄스의 개수가 한 개일 때와 같으므로 식(3)과 같이 표현된다.

$$R = \frac{60}{P \cdot T_c} (rpm) \quad (3)$$

수식(3)에서 보는 것과 같이 샘플링 주기 T_c 가 일정한 상태에서, 휠이 한 바퀴 돌 때의 펄스의 개수인 P 값이 일정하므로 그 분해능은 수식에서 보는 바와 같이 일정하게 된다.

2.1.4 T-방식의 분해능

T-방식의 분해능은 임의의 속도에서 클럭 펄스의 개수를 실제보다 한 개 놓칠 때 발생하는

속도의 차이와 같으므로 아래 식(4) 와 같이 표현될 수 있다.

$$R = \frac{w^2 \cdot P}{60 \cdot f_c + w \cdot P} \quad (4)$$

여기서 오차율 R 은 m_2 가 하나의 펄스를 놓치는 경우 발생하는 속도의 차이로 정의 되어 있는 바와 같이 $R = w - w'$ 라고 하면(여기서 $w = m_2$ 일 때의 속도, $w' = m_2 - 1$ 일 때의 속도), 식(2)를 통해 m_2 를 w 에 대한 식으로 정리한 다음 $R = w - w'$ 에 적용하면 식(4)를 유도 할 수 있다. T-방식의 속도 측정 방법은 회전속도가 증가할수록 분해능이 저하되고, 회전 속도가 줄어들면 분해능이 향상된다[1].

2.2 반작용 휠 속도 측정 보드 설계

본 절에서는 두 방법의 로직 설계와 그에 따른 로직 시뮬레이션 및 보드 설계에 대해 설명하고자 한다.

전체 로직 Block은 Fig. 3에서 보는 것과 같이 Command 디코딩부와 Tacho 펄스 카운터부, 그리고 카운터 Result Latch 레지스터로 구성이 된다.

2.2.1 커맨드 디코딩 로직 설계

Local Bus로부터 CS, Address, Data를 기준 클럭과 함께 입력을 받게 되고, 보드 내부에 할당된 카운터 레지스터 값에 대한 요청이 있을 경우, 현재 저장된 카운터 Latch 레지스터 값을 Bus에 실어주는 역할을 한다.

2.2.2 M-방식 로직 설계

M-방식 로직은 Fig. 4와 같이 타코 펄스에 대해 8Hz 샘플링 시간을 기준으로 펄스 카운팅을 수행하며, 샘플링 시간 기준으로 Ping-Pong Up-

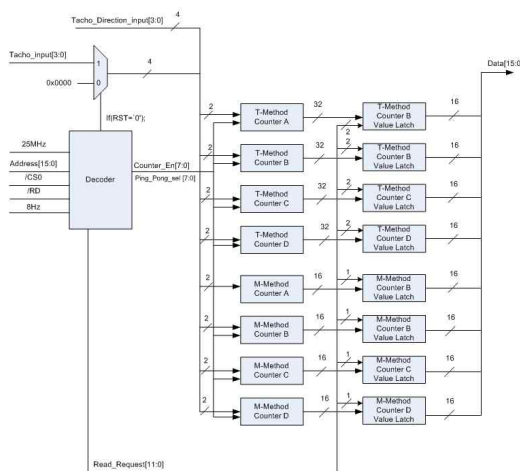


Fig. 3. T&M-방식의 로직 블록도

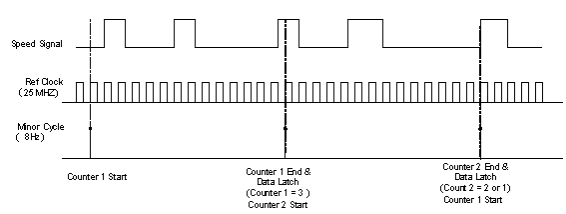


Fig. 4. M-방식 타이밍

```

01 -- M-Method.vhd
02
03 M_TD_CNT_LAT:process(B8HZ,m_cnt_en,BRST)
04 begin
05     if(BRST = '0') then
06         temp_lat<='0';
07     end if;
08
09     m_td_cnt_latch <= (others => '0');
10     end_m_latchd1 <= '0';
11     end_m_latchd2 <= '0';
12     elsif(B8HZ event and B8HZ = '1') then
13         if(m_cnt_en = '1') then
14             temp_lat<='0';
15         end if;
16
17         m_td_cnt_latch <= m_td_cnt1;
18         end_m_latchd1 <= '1';
19         end_m_latchd2 <= '0';
20     elsif(m_cnt_en = '0') then
21         temp_lat<not temp_lat;
22     end if;
23
24     m_td_cnt_latch <= m_td_cnt2;
25     end_m_latchd1 <= '0';
26     end_m_latchd2 <= '1';
27 end if;
28 end process;
29
30
    
```

Fig. 5. M-방식 카운터 로직 코드

카운터를 설계함으로써 카운팅 결과 값의 요청 여부에 상관없이 카운팅을 계속 수행하도록 하였다. Ping 또는 Pong 카운터가 카운팅을 시작하면 반대편 카운터는 클리어 된다.

M-방식 로직은 위성탑재컴퓨터의 제어 주기가 내부 Double Buffered 레지스터를 갱신하게 되어 실제의 요청에 대해 Data 충돌이 없도록 로직 설계를 수행했다. M-카운터 결과 값을 저장하는 레지스터는 Direction bit를 포함한 16bits 레지스터로 설계했다.

2.2.3 T-방식 로직 설계

T-방식 로직은 Fig. 6에서와 같이 타코 펄스에 대해 25Hz 클럭 펄스 카운팅을 수행하며, 타코 펄스의 rising edge를 기준으로 Ping-Pong Up-카운터를 구동한다. Ping 또는 Pong 카운터가 카운팅을 시작하면 반대편 카운터는 M-카운터와 마찬가지로 클리어 된다.

T-방식 로직도 M-방식과 마찬가지로 데이터 Arbitration은 M-방식과 마찬가지로 위성탑재컴

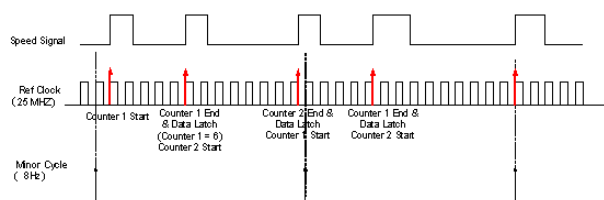


Fig. 6. T-방식 타이밍

```

-- T-Method.vhd
-- Counter
T_TA_COUNT_PROCESS(CLK, t_ta_cnt_en, BRST, t_tacha_end)
begin
    if (BRST = '0') then
        t_ta_cnt1 <= (others => '0');
        t_ta_cnt2 <= (others => '0');
    elsif (rising_edge(CLK)) then
        if (t_ta_cnt_en = '1') then
            if (t_ta_cnt1 <= T_CNT_MAX) then
                t_ta_cnt1 <= t_ta_cnt1 + '1';
            end if;

            if (t_tacha_end = '1') then
                t_ta_cnt2 <= (others => '0');
            end if;
        elsif (t_ta_cnt_en = '0') then
            if (t_ta_cnt2 <= T_CNT_MAX) then
                t_ta_cnt2 <= t_ta_cnt2 + '1';
            end if;
        end if;

        if (t_tacha_end = '1') then
            t_ta_cnt1 <= (others => '0');
        end if;
    end if;
end process;
end entity;
    
```

Fig. 7. T-방식 카운터 로직 코드

퓨터의 제어 주기마다 내부 Double Buffered 레지스터를 갱신하게 되어 실제의 요청에 대해 데이터 충돌이 없도록 로직 설계를 수행했다. T-카운터 결과 값을 저장하는 레지스터는 Over flow와 Direction bit를 갖는 32 bit 레지스터로 설계했다.

2.2.4 M-방식 로직 시뮬레이션

시뮬레이션은 Mentor Graphics사의 ModelSim을 이용하여 수행하고, 시뮬레이션 단계는 Pre Synthesis 시뮬레이션을 통하여 Functional 검증을 수행하고, Post Synthesis 시뮬레이션, Post Layout 시뮬레이션을 통하여 실제 Target FPGA Timing 시뮬레이션을 수행했다.

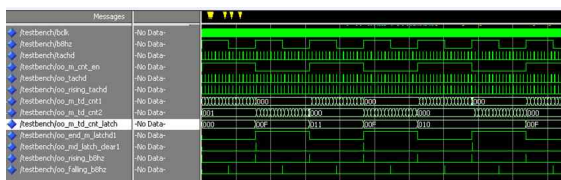


Fig. 8. M-방식 Pre-Synthesis 결과

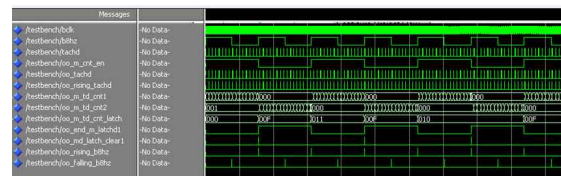


Fig. 9. M-방식 Post-Synthesis 결과

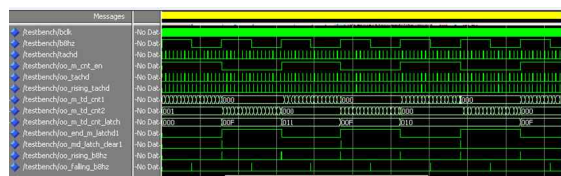


Fig. 10. M-방식 Post-Layout 결과

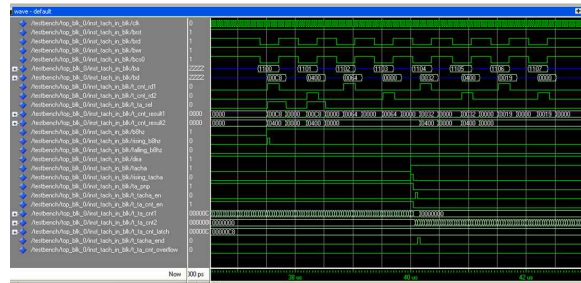


Fig. 11. T-방식 Pre-Synthesis 결과

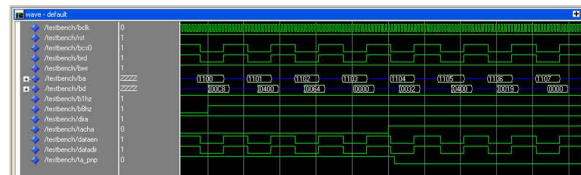


Fig. 12. T-방식 Post-Synthesis 결과

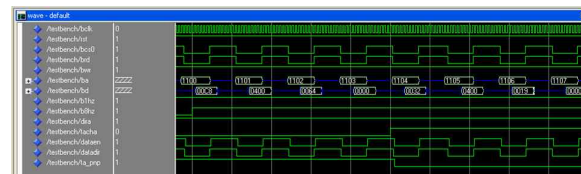


Fig. 13. T-방식 Post-Layout 결과

2.2.5 T-방식 로직 시뮬레이션

T-방식에 대한 로직 시뮬레이션 역시 M-방식 로직 시뮬레이션과 동일한 방법으로 수행했다.

실제 위성의 타코 펄스는 40us의 펄스폭을 가지고 있다. Post Layout 이 후, Worst Timing에 대해 Libero IDE Smart Timing Analysis Tool[7]을 사용하여 Worst Case Analysis를 수행해 보면, 다음 Fig. 14와 같이 로직 Delay는 33ns 정도로 FPGA 기준 클럭 25Mhz의 1 클럭보다 작은 값을 가지므로 로직 Timing Delay는 8Hz 제어 Cycle 기준에서 볼 때, 거의 없다고 볼 수 있다.

2.2.6 보드 설계

보드는 타코 펄스 입력 Circuit으로부터 FPGA에 타코 펄스와 Direction을 제공하고, FPGA는 버스 인터페이스를 위해 Address, CS, Data 그리고, Reset, 클럭 입력을 받는다.

Design		TOP_RLK		Max Operating Condition: WORST				
Family		S4624A		Min Operating Condition: BEST				
Die		R7S45X25		Voltage: COM				
Package		208 CGSP		Temperature: COM				
				Speed Grade: STD				
				Relaxation Epilogue: 0				
Clock Detail:								
Name	Period (ns)	Frequency (MHz)	Required Period (ns)	Required Frequency (MHz)	External Setup (ns)	External Hold (ns)	Max Clock (MHz)	Min Clock (MHz)
BCLK	40.000	25.000	N/A	N/A	1.007	0.368	24.976	25.024
inst_TACH_IN_0[0:1]	N/A	N/A	N/A	N/A	N/A	N/A	14.089	23.903
inst_V2CH_IN_0[0:1]	N/A	N/A	N/A	N/A	N/A	N/A	14.366	4.589
IO Detail:								
Name	Min Delay (ns)	Max Delay (ns)						
Input to Output	2.174	33.762						

Fig. 14. Worst Timing 해석 결과

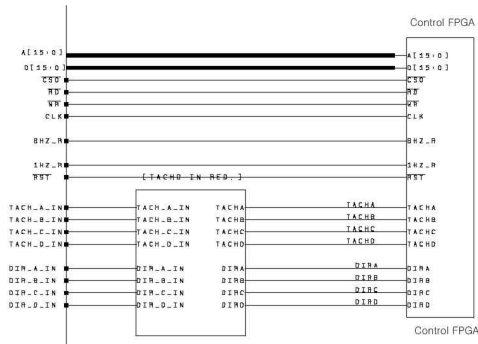


Fig. 15. 회로 블록도

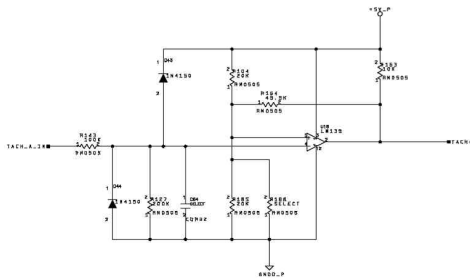


Fig. 16. 타코 펄스 입력 회로

반작용 휠로부터 입력되는 타코 펄스는 반작용 휠의 회전 속도를 검출하기 위한 것으로, FPGA내부에서 카운터 로직을 통하여 회전 속도를 측정하게 된다. 아래 Fig. 16은 Noise를 고려한 타코 펄스 입력 회로의 구성이다.

실제 보드 클럭은 25Mhz를 사용하고, FPGA는 Space level의 Actel사 RT54SX32a-CQ208을 사용한다. 보드 Circuit Block Diagram은 Fig. 15과 같고, 타코 펄스 입력 회로는 Fig. 16과 같다. 실제 제작된 보드는 Fig. 17, 18에 보인 바와 같다. 보드 설계에 있어서 Part Stress Analysis 및 FMECA, 구조/열 해석, 신뢰도 해석을 수행하였으며, 이 부분은 우주 운용환경에 맞도록 설계에 반영하였다.

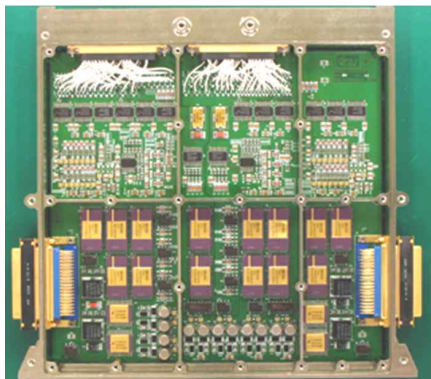


Fig. 17. 보드 사진 Top

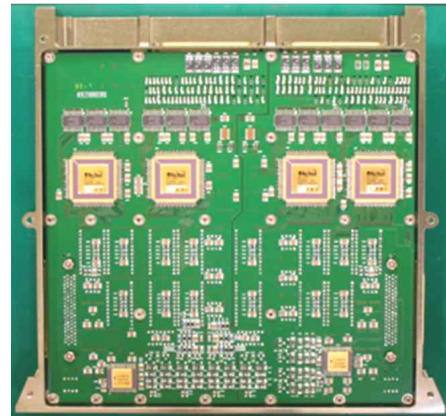


Fig. 18. 보드 사진 Bottom

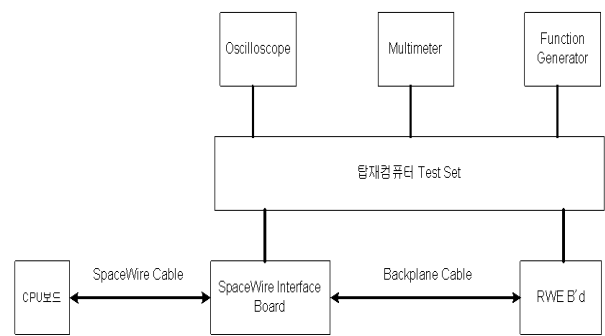


Fig. 19. 보드 시험 환경

2.3 시험 결과

시험 및 검증은 실제 위성시스템에 적용을 위해, Function Test와 환경시험을 통하여 위성 시험환경 조건에서 그 기능 검증을 수행했다.

2.3.1 시험 환경 구성

시험 환경은 탑재컴퓨터 전용 검증장비를 사용하며, 모사된 반작용 휠 타코 펄스는 Function Generator를 사용하여 보드로 전송하도록 한다. 보드에서 계산된 카운터 값은 CPU의 요청이 있을 때, 전송하도록 한다.

2.3.2 성능 시험 결과

성능 시험은 Table 1과 같이 4가지 타코 펄스 입력 경우에 대해 시험을 수행하고 추가적으로 Overflow 로직 및 Direction 부분도 검증하도록 했다. 세부적인 Variation 설계 요구조건은 다음과 같다.

- 1회전 당 타코 펄스 수 = 108
- M-Count 샘플링 시간 = 8Hz(0.125s)
- T-Count 샘플링 클럭 주파수 = 25Mhz
- 타코 펄스 폭 : 40us

Table 1. 시험 결과

Tacho Pulse	T-Count 요구 값	T-Count 측정 값	속도	T방식 속도분해능	M-Count 요구 값	M-Count 측정 값	속도	M방식 속도분해능
96Hz	260417(±14)	260416	53.33346987	0.0002048	12(±1)	12	53.33333333	4.444444444
544Hz	45956(±3)	45956	302.2214485	0.00657618	68(±1)	68	302.2222222	4.444444444
1KHz	25000(±2)	25000	555.5555556	0.02222133	125(±1)	125	555.555556	4.444444444
4KHz	6250(±1)	6250	2222.222222	0.35549868	500(±1)	500	2222.22222	4.444444444

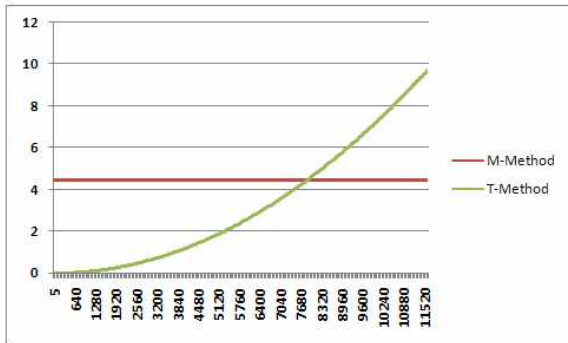


Fig. 20. M-방식과 T-방식의 속도 분해능

성능 시험은 보드 단독 및 탑재컴퓨터 조립 후 시험을 수행하였고, 그 검증 결과는 동일하였다. Fig. 20에서 보는바와 같이 T-Count의 샘플링 주파수를 25Mhz로 했을 경우, 7840rpm 이하에서는 T-방식이 속도 분해능이 더 좋음을 알 수 있다.

2.3.3 환경 시험 결과

환경시험은 우주 환경을 모사하여 환경적인 특성을 검증하는 것으로 MIL-STD-1540B (Test Requirements for Space Vehicles), PSS-01-801 (Test Requirements Specification for Space Equipment), ECSS-E-10-0 (Space Engineering, Testing)의 표준 시험요구조건을 적용하여 시험을 수행한다[2],[3],[4].

기능적인 특성은 성능 시험결과와 동일한 시험 결과를 얻었으며, 환경시험 조건 및 결과는

Table 2. 환경 시험 결과

구분	Test Condition	Test Case	Test Results
Random Vibration Test	10grms/min	X,Y,Z axis	No function fail
Thermal Cycle Test	-25°C to +55°C	8cycle	No function fail
Thermal Vacuum Test	10 ⁻⁶ -5torr	8cycle	No function fail
EMI/EMC Test	CE,CS,RE,RS	CE,CS,RE,RS	No function fail

Table 2와 같다. 시험 결과는 2.3.2의 결과와 동일하게 요구 규격 범위를 초과하지 않았다

III. 결론

본 논문에서 살펴본 바와 같이, 고속에서는 M-방식이 장점이 있고, 저속에서는 T-방식이 장점이 있다고 할 수 있다. 따라서 두 방법을 적절히 사용을 하면, 전체 속도 영역에서 분해능을 향상시킬 수 있을 것이다. 본 논문에서는 저궤도 위성의 반작용 휠 속도 측정을 위해 M-방식과 T-방식 모두 검출 가능한 보드를 설계하여 두 방식의 장점을 이용할 수 있도록 하고, 실제 위성 탑재컴퓨터에 장착하여 지상 검증시험 및 환경시험을 통하여 검증을 수행하였다. 본 논문에서 제안한 혼용 방식은 M-방식에 T-방식을 더하는 M/T 방식이 아닌 적절한 속도 오차율을 만족할 수 있는 범위 내에서 두 방식 중 적합한 방식을 S/W가 선택 할 수 있도록 설계했다. 따라서 M/T방식은 현재 S/W 추정 영역으로 남겨둔 상태이다.

현재 설계된 보드는 다목적실용위성 3호의 자세제어 인터페이스 보드로 적용이 되었으며, 3호 발사 후 우주궤도에서 안정적인 자세제어 관련 임무를 수행 하고 있다. 향후 과제로는 결론에서 언급한 바와 같이 M-방식에 T-방식을 더해서 속도를 검출할 수 있는 M/T 방식을 하드웨어로 구현하여 더욱 향상된 위성 반작용 휠 속도 검출기를 설계 하도록 해야겠다.

참고문헌

- 1) 오시환, 이해진, 이선호, 용기력, 반작용휠 속도측정방법의 오차 분석, 한국우주과학회지, 제 25권 4호, pp. 481~490
- 2) MIL-STD-1540B, Test Requirements for Space Vehicles
- 3) PSS-01-801, Test Requirements Specification for Space Equipment
- 4) ECSS-E-10-0, Space Engineering, Testing
- 5) www.actel.com