

FPGA를 이용한 다기능 고속 카운터 구현에 관한 연구

이도향, 양 오  
 청주대학교 이공대학 전자공학과

Research about a multifunction high-speed counter implementation which uses FPGA

Do-Hyang Lee, Oh Yang  
 Dept. of Electronic Engineering, Cheong-Ju university

**Abstract** - In this paper, We used FPGA which was high speed counter implementation. It was Counting accurately rather fast so that there were we as a counter facility of the pulse implemented.

We constructed wide environment rather because we used H8/3672 with FPGA. This FPGA was synthesized by A54SX72A. FPGA programmed by VHDL for a 208pin PQFP package. The measurement the pulse is possible though it peels off a maximum 200kHz.

There is used at a stopping action movement and control of the body.

1. 서 론

기존의 카운터 기능은 단순히 업·다운(Up·Down) 카운터 기능만을 구현하여 단순히 사람이나 물건의 들어오고 나가는 수를 체크 하는 거였으나 그 기능을 다양화 하여 엘리베이터나 컨베이어벨트 등에 이용하여 원하는 곳에서 정지 및 위치 추적을 할수 있게 하였고 채널을 2개로 하여 2개의 채널이 동시에 동작을 하고 펄스의 주파수를 측정할 수 있게 하므로써 동작하는 기기의 속도도 측정할 수 있다.

펄스입력의 종류를 6개로 하여 여러 종류의 펄스를 카운터 할 수 있게 하였다.

카운터 기능을 Software적으로 선택을 할 수도 있으며 Hardware적으로도 할수 있게 하여 산업현장에서 직접 카운터 기능을 선택할 수도 있고 컴퓨터로 모니터링을 하며 기능을 선택할 수도 있게 하였다. 이렇게 여러 가지 기능 카운터를 구현하므로 여러 방면으로 쓰일 수 있게 하였다. 위와 같은 기능을 고속화하기 위해 VHDL을 이용하였다. FPGA는 ACTEL사의 A54SX72A를 이용하여 구현하였다.

2. 본 론

2.1 동작 및 기능

고속으로 카운터를 하기 위해 200kHz까지 카운터가 되도록 했으며 넓은 범위에 대해서 카운터하기 위해 32bit binary 값(-2147483648~2147483647)으로 카운터된 값을 버퍼메모리에 저장하였다. 채널을 2채널로 하여 2채널을 동시에 사용할수 있게 하였다.

전체적인 동작은 다음과 같다.

본 연구는 과학기술부·한국과학재단 지정 청주대학교 정보통신연구센터(RRC)의 지원에 의한 것입니다.

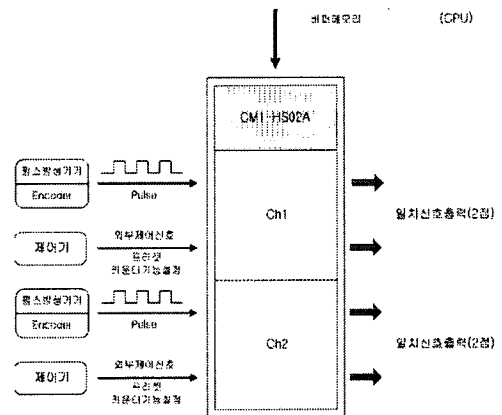


그림 1. 고속 카운터의 동작

펄스 입력단자에 들어오는 펄스를 카운트를 한다. 단순히 그냥 펄스를 카운트 하는게 아니라 여러 종류의 펄스를 카운트 할 수가 있다.

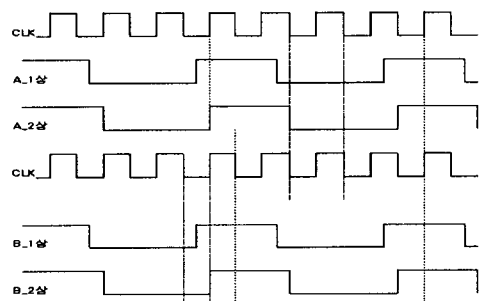


그림 2. A상과 B상의 펄스 입력

위의 그림 2에서와 같이 A\_1상은 외부 입력 펄스를 나타내는데 CLK이 상승 에지(Edge)일때 A상과 같은 펄스를 A\_2상을 만든다. 이렇게 하여 A상이 '1'이고 A\_1상이 '0'일때를 체크하여 입력 펄스의 상승 에지(Edge)를 체크할수 있게 하고 A\_1상이 '0'이고 A\_2상이 '1'일때 체크하여 입력 펄스의 하강 에지(Edge)체크를 할 수 있다. 외부 입력 펄스가 하나 더 있는데 그것을 B상이라 하여 A\_1상과 같이 상승,하강 에지(Edge) 체크할 수 있게 하였다. 이렇게 하여 A\_1상과 B\_1상의 에지(Edge) 및 Level 체크를 이용하여 펄스의 입력 종류를 만들 수가 있다. 여러 종류의 펄스를 입력 받을 수 있게 하여

사용자가 선택한 펄스가 들어왔을 때만 카운트 하게 할 수 있도록 다음과 같이 설계를 하였다.

```
A_1QN <= not(A_1Q);
A_2QN <= not(A_2Q);
B_1QN <= not(B_1Q);
B_2QN <= not(B_2Q);
```

```
A1 <=(A_1Q and A_2Q and B_1QN and B_2Q);
A2 <=(A_1Q and A_2Q and B_1Q and B_2QN);
A3 <=(A_1QN and A_2QN and B_1Q and B_2QN);
A4 <=(A_1QN and A_2QN and B_1QN and B_2Q);
A5 <=(A_1Q and A_2QN and B_1Q and B_2Q);
A6 <=(A_1QN and A_2Q and B_1Q and B_2QN);
A7 <=(A_1QN and A_2Q and B_1QN and B_2QN);
A8 <=(A_1Q and A_2QN and B_1QN and B_2QN);
```

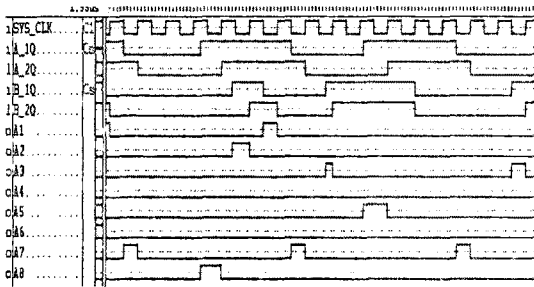


그림 3. 상승 하강 에지 체크 및 레벨 체크 시물레이션

위 소스를 시물레이션하면 그림 3과 같이 된다. 즉 A1는 A상이 '1'이고 B상이 하강에지(Edge)일때 set이 되고 A2는 A상이 '1'이고 B상이 상승에지(Edge)일때 set이 되고 A3는 A상이 '0'이고 B상이 상승에지(Edge)일때 set이 되며 A4은 A상이 '0'이고 B상이 하강에지(Edge)일때 set, A5은 A상이 상승에지(Edge)이고 B상이 '1'일때 set, A6은 A상이 하강에지(Edge)이고 B상이 '1'일때 set, A7은 하강에지(Edge)이고 B상이 '0'일때 set, A8은 A상이 상승에지(Edge)이고 B상이 '0'일때 set이 된다. 이렇게 다양하게 펄스를 체크 할수 있는 것을 조합하여 다음과 같이 하였다. X\_CW는 가산 signal이고 X\_CCW는 감산 signal로 하였다.

1상 1체배는

```
X_CW <= A8;
X_CCW <= A6
```

1상 2체배는

```
X_CW <= (A7 or A8);
X_CCW <= (A5 or A6);
```

CW/CCW 는

```
X_CW <= A8;
X_CCW <= A3;
```

2상 1체배는

```
X_CW <= A8;
X_CCW <= A7;
```

2상 2체배는

```
X_CW <= (A6 or A8);
X_CCW <= (A5 or A7);
```

2상 4체배는

```
X_CW <= (A2 or A4 or A6 or A8);
X_CCW <= (A1 or A3 or A5 or A7);
```

위와 같이 여러 가지 종류의 펄스 입력을 구현 할 수 있다.

카운터 형식에는 2가지 방식을 구현했는데 첫 번째로 리니어 카운터 이다. 두개의 계수 방식을 구분할 수 있는 Signal을 만들어 이를 모드를 선택을 할 수 있게 했다. 리니어 카운터 동작은 리니어 카운터 모드가 선택 되면 카운터 동작은 -2147483648~2147483647 사이에서 동작을 한다.

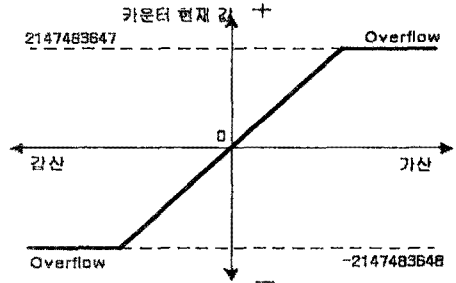


그림 4. 리니어 카운터 계수 방식

그림 4에서 보듯이 -2147483648, 2147483647를 초과했을 시 Overflow가 발생하며 카운터가 정지하게 하였다.

두 번째로 링 카운터는 링카운터 모드가 선택되면 비퍼 메모리의 링 카운터의 최소값과 최대값 사이의 값에서 반복적으로 동작이 된다.

POS\_LIMIT(최대값),NEGA\_LIMIT(최소값)이란 Signal을 만들어 이를 카운터 되는 값과 계속하여 비교 하여 카운터를 한다. 링카운터에서는 Overflow가 발생하지 않도록 구현하였다.

링카운터의 최대값과 최소값은 임의적으로 사용자가 정할 수 있도록 하였다

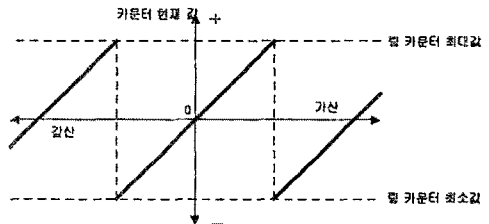


그림 5. 링 카운터 계수 방식

위의 그림 5에서

가산(UP) 카운터 인 경우 : 현재 값이 링 카운터의 최대값에 도달 한 경우 링 카운터의 최소값이 자동적으로 현재 값에 저장 된다.

감산(DOWN) 카운터 인 경우 : 현재 값이 링 카운터의 최소값에 도달 한 경우 링 카운터의 현재 값은 그 값을 보유하고 있다가 다음 감산 펄스가 들어오면 "링 카운터의 최대값 1"의 값이 현재 값에 저장 된다.

4가지의 카운터 FUNCTION기능을 선택할수 있도록 하였다.

- ① 래치 카운터 기능: 카운터 기능 선택 시작(Y06,Y0E)신호에 의해 현재의 카운터 값을 비퍼메모리에 저장
- ② 샘플링 카운터 기능: 카운터 기능 선택 시작(Y06,Y0E) 신호에 의해 설정된 시간이 지나면 다시 카운터 기능 선택시작 신호가 입력되기 전까지 값을 유지 한다.

③주기 펄스 카운터 기능: 카운터 기능 선택 시작(Y06,Y0E) 신호에 의해 설정된 시간의 이전 카운터 값과 현재 카운터값을 버퍼에 저장한다.

④카운터 기능 사용금지:카운터 기능 선택시작(Y06,Y0E) 신호가 OFF거나 카운터 허가 신호가 OFF이면 카운터 하지 않는다.

위와 같은 기능으로 펄스의 값을 저장 및 정지 주기측정을 할 수 있다.

일치 출력 기능 및 프리셋(Preset)기능을 구현 하였다. 우선 Preset기능이 있는데 이것은 현재의 카운터 값을 Preset 값이라는 하는 어느 정수 값으로 고쳐서 쓸수 있는 것이다. 즉 내가 원하는 값부터 카운터를 시작할 수 있게 구현한 것이다. Preset 기능은 Software적으로 사용할 수도 있고 Hardware적인 외부 컨트롤 신호에 의해서 사용할 수도 있도록 하였다.

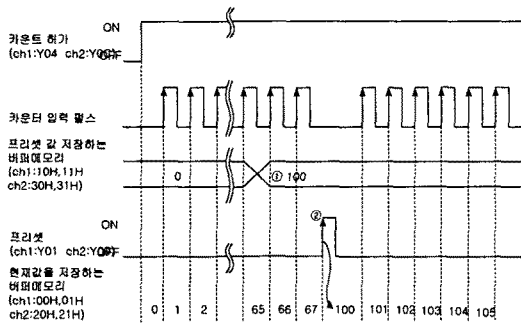


그림 6. Preset 기능에 대한 동작

위의 그림 6과 같이 버퍼메모리의 Preset 값 즉, 부호 있는 32bit binary 형태로 임의의 값이 저장 되어 있다. Preset 명령의 상승에지(OFF에서ON)에서 현재 카운터 값이 저장되어 있는 버퍼메모리에 Preset값이 저장되어 있는 버퍼메모리의 값으로 설정이 된다. 외부에서 입력을 24V, 12V, 5V로 할 수 있게 하여 이와 같은 전압이 입력 시 포토커플러를 거쳐 이를 체크하는 Signal을 두어 이 Signal이 '1'로 되었을 때 Preset 값이 카운터 값으로 들어 갈수 있게 하였다.

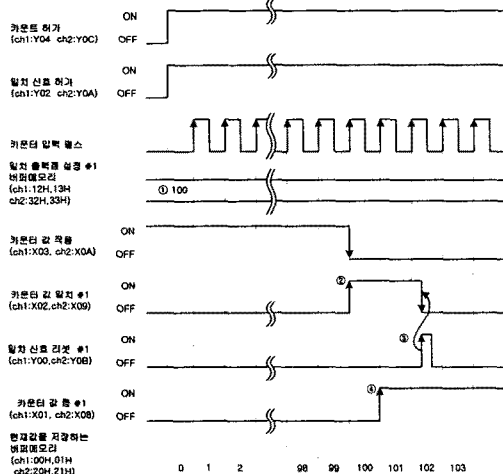


그림 7. 일치 출력 기능에 대한 동작

위의 그림 7과 같이 일치 출력 기능은 어느 카운트 값으로 미리 설정된 것을 현재의 카운터 값과 비교하여 서로 일치되었을 때 외부 신호를 출력한다. 일치 출력은 각 채널에 대해 2개의 점점을 설정할 수 있도록 하였다. 일치 신호 허가 signal(Y02(Y0A))을 구현하여 외부로 일치신호를 출력 하기 위해서는 일치신호(Y02(Y0A))를 ON 하도록 하였다. CMP\_REG1, CMP\_REG2 라는 두 개의 레지스터를 만들어 두개의 값을 비교하여 일치 하였을 시 외부로 일치 출력을 내보내도록 하였다. 그림 7에서와 같이 일치 신호 리셋 명령이 ON되면 카운터 값 일치신호가 리셋 되도록 하여 다시 설정하여 사용 할수 있게 구현하였다.

위와 같이 많은 카운터 기능을 구현하므로써 보다 편리하고 유용하게 사용할 수 있도록 하였다.

## 2.2 FPGA와 H8/3672 interface 및 모니터링

FPGA만으로 고속 카운터 기능을 구현하는데 한계가 있어 H8/3672인 Hitachi사의 16비트 내부구조를 가진 고속 H8/300H CPU를 사용하여 보다 폭넓은 환경을 구축하였다.

그리고 window 프로그램을 사용하여 카운터 기능을 제어 할 수 있게 하였으며 그 외 모니터링을 실시간으로 할 수 있게 하여 카운터 기능 및 카운터 값을 수시로 확인 할 수 있게 하였다.

H8/3672는 FPGA가 카운터 한 값을 가져와서 Dpram에 저장하고 이에 관한 데이터를 window 프로그램에 전달하여 실시간으로 모니터링을 할수 있게 하였다.

여러 가지 기능을 Software 적으로 설정 및 실행을 할 수도록 하였으며 카운터가 되는 것도 모니터링 할 수 있게 하였다.

그리고 전원 OFF시 전에 사용했던 데이터들이 사라지는 것을 방지하기 위해 EEPROM을 사용하여 전원 OFF 하기 전의 데이터를 저장하여 다시 전원 ON 하였을때 전과 같은 상태로 다시 시작 할 수 있도록 하였다.

## 3. 결 론

본 논문에서는 FPGA를 사용하여 다기능 고속카운터를 구현하여 보다 안정적이고 보다 빠른 동작을 할수 있도록 하였으며 이를 H8/3672를 이용하여 데이터를 저장 및 CIMON이란 프로그램에 모니터링을 할 수 있도록 설계를 하였다. 설계된 이 시스템은 보다 편리하게 카운터 기능을 수행 할 수 있으며 사용자가 사용시 보다 제어하기 쉽게 제어 할수 있도록 Software, Hardware적으로 둘다 될 수 있도록 설계하였다.

이를 사용하는 산업 현장에서 보다 편리하고 쉽게 기기를 제어 할수 있도록 노력하였다.

## [참 고 문 헌]

- [1] HITACHI Semiconductor, "Hitachi Single-Chip Microcomputer H8/3672 Series, H8/3672 Serises Hardware Manual", 2001년
- [2] ACTEL Semiconductor, "SX-A Family FPGAs A54SX72A Hardware Manual", 2001년
- [3] HITACHI Semiconductor, "Hitachi Single-Chip Microcomputer H8/3672 Series, H8/300H Serises Programming Manual", 2001년