Stepping Motor Control 예비보고서

20101449 한가은

1. 실험 개요

1.1. 실험 목적

Stepping motor를 원하는 position 및 velocity가 되도록 제어한다. 사다리꼴 속도 프로파일을 통한 가감속 제어 기법을 통해 탈조가 발생하지 않는 최대 속도로 모터를 제어해본다.

1.2. 실험 대상 및 배경

1.2.1. Stepping motor란

스텝 각을 기본 회전 단위로 하는 모터로, 고정자로 코일을, 회전자로 영구자석을 갖는다. 지속적으로 한 방향으로 회전하기 위하여 180 degree 마다 코일에 흐르는 방향을 반전시키는데, 이는 고정자 권선 결선 구조에 따라 H-bridge 등의 전기 회로에 의존한다.

고정자 결선 구조에 따라 Unipolar 구조와 Bipolar 구조로 나뉘며, 구동 방식에 따라 Full step, Half step, Micro step 구동으로 나뉜다.

- Full step 구동: 분해능 = 스텝각. 고정자의 align 방향에 따라 1상 여자 방식 or 2상 여자 방식
- Half step 구동: 분해능 = ‘스텝각/2’. 1상 여자 방식 + 2상 여자 방식
- Micro step 구동: 분해능 = ‘스텝각/N’ where N is the number of micro step

Stepping motor 사용 시 코일 전류 상의 변화 속도를 따라가지 못해 고정자가 제자리에서 진동하는, 탈조 현상을 유의해야 한다.

1.2.2. Stepping motor에 사용되는 제어 기법

Stepping motor는 통상적으로 open loop으로 제어된다.

- overshoot을 줄이는 제어기 없음 → **micro step 구동을 통해 진동 감소**
- 탈조 현상 발생 여부 감지 불가 → **가감속 제어 기법 도입**

<Stepping motor의 위치 제어>

- 회전 각도 $\propto$ 스텝 수
- 스텝 각 $\Delta\theta$, 상 전환 횟수 k, micro step 수 N에 대해,
  - Full step: $s = \Delta\theta \times k$
  - Half step: $s = \frac{\Delta\theta}{2} \times k$
  - Micro step: $s = \frac{\Delta\theta}{N} \times k$

<Stepping motor의 속도 제어>

- 회전 속도 $\propto \Delta t$
- 스텝 각 $\Delta\theta$, 상 전환 시간 Δt, micro step 수 N에 대해,
  - Full step: $v = \frac{\Delta\theta}{\Delta t}$
  - Half step: $v = \frac{\Delta\theta}{2\times\Delta t}$
  - Micro step: $\frac{\Delta\theta}{N\times\Delta t}$

2. 실험 내용

2.1. 실험 절차

실험은 세 단계로 진행되며, 각 단계는 다음과 같다.

2.1.1. GPIO를 이용한 stepping 모터 구동

- Skeleton code를 통한 1상여자 구동방식을 사용
- delay time 변경을 통해 모터의 회전 속도 변경
- 과도하게 짧은 delay time을 설정하여 탈조 현상 확인
- Half step 구동 구현
- 모터 1회전에 필요한 스텝 계산
  - Full step: $\theta = 360/1.8 = 200steps$
  - Half step: $\theta = 360/(1.8/2) = 400steps$

2.1.2. Stepping 모터 구동 API 작성

스테핑 모터 구동을 위한 두 함수를 작성한다.

- `void OneSteopMove(unsigned int dir, unsigned int tDelayCnt)`
  - 모터를 정방향 or 역방향으로 1 스텝 회전시킨다.
  - *dir:* 0은 정방향, 1은 역방향
  - *tDelayCnt:* 상 변환 후, tDelayCnt에 지정한 횟수 만큼 타이머 인터럽트가 발생할때까지 대기
  - **현재 인가된 상의 상태를 알아야** 다음 상태를 생성할 수 있음 → global variable.

- `void OneSteopMove(unsigned int dir, unsigned int tDelayCnt)`
  - 모터를 원하는 각도까지, 원하는 속도로 회전시킨다.
  - *angle*: angle에 지정한 스텝 만큼 회전. 양수면 정방향, 음수면 역방향.
  - *spd*: 회전 스피드, deg/s 단위.
  - *angle*과 *spd*로부터 회전 방향, 스텝 수, 스텝간 시간 지연을 타이머 인터럽트 주기 수로 환산하고, *OneStepMove*를 호출하여 구현한다.

StepMoveCV 함수 호출 시, *spd* 초기값을 너무 크게 설정하면 탈조가 발생하니 주의한다.

2.1.3. 속도 프로파일의 적용

감가속 제어를 위한 사다리꼴 속도 프로파일을 생성하기 위해 두 함수를 작성한다.

- `unsigned int MakeVelProfile(float maxVel, float accel)`
  - 가속 구간에 대해서만 '*maxVel, accel*' 두 개의 parameter을 사용하여 룩업테이블을 작성.
  - 룩업테이블은 200개를 넘지 않도록
- `void StepMoveVP(float angle, float maxVel, float accel)`
  - *MakeVelProfile* 함수를 호출 → 룩업테이블을 생성, 이를 활용해 스텝 모터 회전.
  - 사다리꼴 속도 프로파일을 결정하기 위한 3개의 parameter: '*angle, maxVel, accel*'. '*maxVel, accel*'은 절대값으로만 사용.
  - 가속 구간: 룩업테이블 index 증가
  - 등속 구간: min_delay 사용(일정한 index)
  - 감속 구간: index 감소

StepMoveVP 함수를 호출할 때, *maxVel*과 *accel* 값이 너무 크지 않게, *angle*값은 너무 작지 않게 설정할 것에 유의한다. 또한, 기구의 공진 주파수와 상전환 주파수가 일치해 진동으로 인해 탈조가 발생한 상황인지 확인한다.