

Brushless DC Motor Control Made Easy

INTRODUCTION(소개)

This application note discusses the steps of developing several controllers for brushless motors. We cover sensed, sensorless, open loop, and closed loop design. There is even a controller with independent voltage and speed controls so you can discover your motor's characteristics empirically.

이 애플리케이션 노트에서는 브러시리스 모터 용 컨트롤러를 개발하는 단계에 대해 설명합니다. sensed, sensorless, 개방루프 및 폐쇄루프 설계를 다루고 있다. 독립적 인 전압 및 속도 제어 기능이 있는 컨트롤러도 있으므로 모터의 특성을 경험적으로 발견 할 수 있습니다.

The code in this application note was developed with the Microchip PIC16F877 PICmicro® Microcontroller, in conjunction with the In-Circuit Debugger (ICD). This combination was chosen because the ICD is inexpensive, and code can be debugged in the prototype hardware without need for an extra programmer or emulator. As the design develops, we program the target device and exercise the code directly from the MPLAB® environment. The final code can then be ported to one of the smaller, less expensive, PICmicro microcontrollers. The porting takes minimal effort because the instruction set is identical for all PICmicro 14-bit core devices.

이 애플리케이션 노트의 코드는 ICD (In-Circuit Debugger)와 함께 PIC16F877 PICmicro® 마이크로 컨트롤러로 개발되었다. 이 조합은 ICD가 저렴하고 여러분의 프로그래머 또는 에뮬레이터 없이도 프로토타입 하드웨어에서 코드를 디버깅 할 수 있기 때문에 선택되었다. 설계가 진행되면서 대상 장치를 프로그래밍하고 MPLAB® 환경에서 직접 코드를 실행합니다. 최종 코드는 작고 저렴한 PICmicro 마이크로 컨트롤러 중 하나에 이식 될 수 있습니다..명령어 세트는 모든 PICmicro 14 비트 코어 장치에서 동일하기 때문에 이식 작업은 최소한의 노력을 필요로 합니다.

It should also be noted that the code was bench tested and optimized for a Pittman N2311A011 brushless DC motor. Other motors were also tested to assure that the code was generally useful.

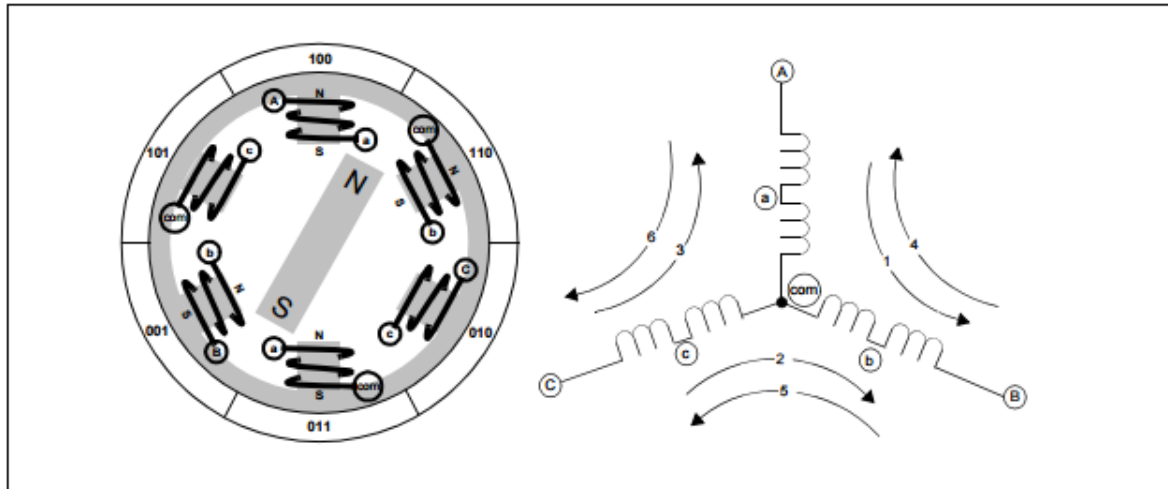
이 코드는 벤치 테스트를 거쳐 Pittman N2311A011 브러시리스 DC 모터 용으로 최적화되었습니다. 코드가 일반적으로 유용하다는 것을 보장하기 위해 다른 모터도 테스트되었습니다.

Anatomy of a BLDC

Figure 1 is a simplified illustration of BLDC motor construction. A brushless motor is constructed with a permanent magnet rotor and wire wound stator poles. Electrical energy is converted to mechanical energy by the magnetic attractive forces between the permanent magnet rotor and a rotating magnetic field induced in the wound stator poles.

그림 1은 BLDC 모터 구조의 단순화 된 그림이다. 브러시리스 모터는 영구 자석 회 전자와 권선 고정자 극으로 구성됩니다. 전기 에너지는 영구 자석 회 전자와 권선 고정자 극에 유도 된 회전 자기장 사이의 자기 흡인력에 의해 기계 에너지로 변환된다.

FIGURE 1: SIMPLIFIED BLDC MOTOR DIAGRAMS



In this example there are three electromagnetic circuits connected at a common point. Each electromagnetic circuit is split in the center, thereby permitting the permanent magnet rotor to move in the middle of the induced magnetic field. Most BLDC motors have a three-phase winding topology with star connection. A motor with this topology is driven by energizing 2 phases at a time. The static alignment shown in Figure 2, is that which would be realized by creating an electric current flow from terminal A to B, noted as path 1 on the schematic in Figure 1. The rotor can be made to rotate clockwise 60 degrees from the A to B alignment by changing the current path to flow from terminal C to B, noted as path 2 on the schematic. The suggested magnetic alignment is used only for illustration purposes because it is easy to visualize. In practice, maximum torque is obtained when the permanent magnet rotor is 90 degrees away from alignment with the stator magnetic field.

이 예에서는 공통점에 연결된 3 개의 전자 회로가있다. 각 전자 회로는 중앙에서 분할되어 있으며, 영구 자석 회 전자를 유도 자기장의 중앙으로 이동시킨다. 대부분의 BLDC 모터 스타 결선 삼상 권선 토폴로지를 가지고 있습니다. 이 토폴로지가 이쁜 L 모터는 한번데 두가지 위상에 동력을 공급합니다. 그림 2에 표시된 정적 정렬은 그림 1의 회로도에서 경로 1로 표시된 단자 A에서 B 로의 전류 흐름을 생성함으로써 구현됩니다. 그림 2에 표시된 정적 정렬은 그림 1의 회로도에서 경로 1로 표시된 단자 A에서 B 로의 전류 흐름을 생성함으로써 구현됩니다. 회로도에서 경로 2로 표시된 C 경로에서 B 경로로 전류 경로를 변경하여 회 전자를 A에서 B로 60도 시계 방향으로 회전시킬 수 있습니다. 제안 된 자기 정렬은 시각화하기 쉽기 때문에 설명 목적으로 만 사용됩니다. 실제로, 최대 토크는 영구 자석 회 전자가 고정자 자기장과 정렬로부터 90도 떨어져있을 때 얻어진다.

The key to BLDC commutation is to sense the rotor position, then energize the phases that will produce the most amount of torque. The rotor travels 60 electrical degrees per commutation step. The appropriate stator current path is activated when the rotor is 120 degrees from alignment with the corresponding stator magnetic field, and then deactivated when the rotor is 60 degrees from alignment, at which time the next circuit is activated and the process repeats. Commutation for the rotor position, shown in Figure 1, would be at the completion of current path 2 and the beginning of current path 3 for clockwise rotation. Commutating the electrical connections through the six possible combinations, numbered 1 through 6, at precisely the right moments will pull the rotor through one electrical revolution.

BLDC 정류의 핵심은 회전자 위치를 감지 한 다음 가장 많은 양의 토크를 발생시키는 위상에 에너지를 공급하는 것입니다. 회전자는 정류 단계마다 60 도의 전기 각으로 이동합니다. 적합한 고정자 전류 경로는 해당 고정자 자기장과의 정렬로부터 회전자가 120 도일 때 활성화되고, 회전자가 정렬 후 60 도일 때 비활성화되며, 다음 회로가 활성화되고 프로세스가 반복됩니다. 그림 1 회전자 위치의 정류 전류 경로 2의 완료시이며, 시계 방향의 경우의 전류 경로 3의 시작이다. 1에서 6까지의 6 개의 가능한 조합에 의한 전기 연결을 정확한 타이밍에 정확하게 하면 회전이 전기적으로 1 회전한다.

In the simplified motor of Figure 1, one electrical revolution is the same as one mechanical revolution. In actual practice, BLDC motors have more than one of the electrical circuits shown, wired in parallel to each other, and a corresponding multi-pole permanent magnetic rotor. For two circuits there are two electrical revolutions per mechanical revolution, so for a two circuit motor, each electrical commutation phase would cover 30 degrees of mechanical rotation.

그림 1의 단순화 된 모터는 1 회 전기 회전은 1 회 기계 회전과 동일합니다. 사실, BLDC 모터는 서로 병렬로 연결된 복수의 전기 회로와 해당 다극 영구 자석 회전자를 갖는다. 두 회로의 경우 기계적 회전 당 2 회 전기 회전이 있기 때문에 두 회로의 모터의 경우 각 전기 정류 단계는 30 도의 기계적 회전을 커버한다.

Sensored Commutation

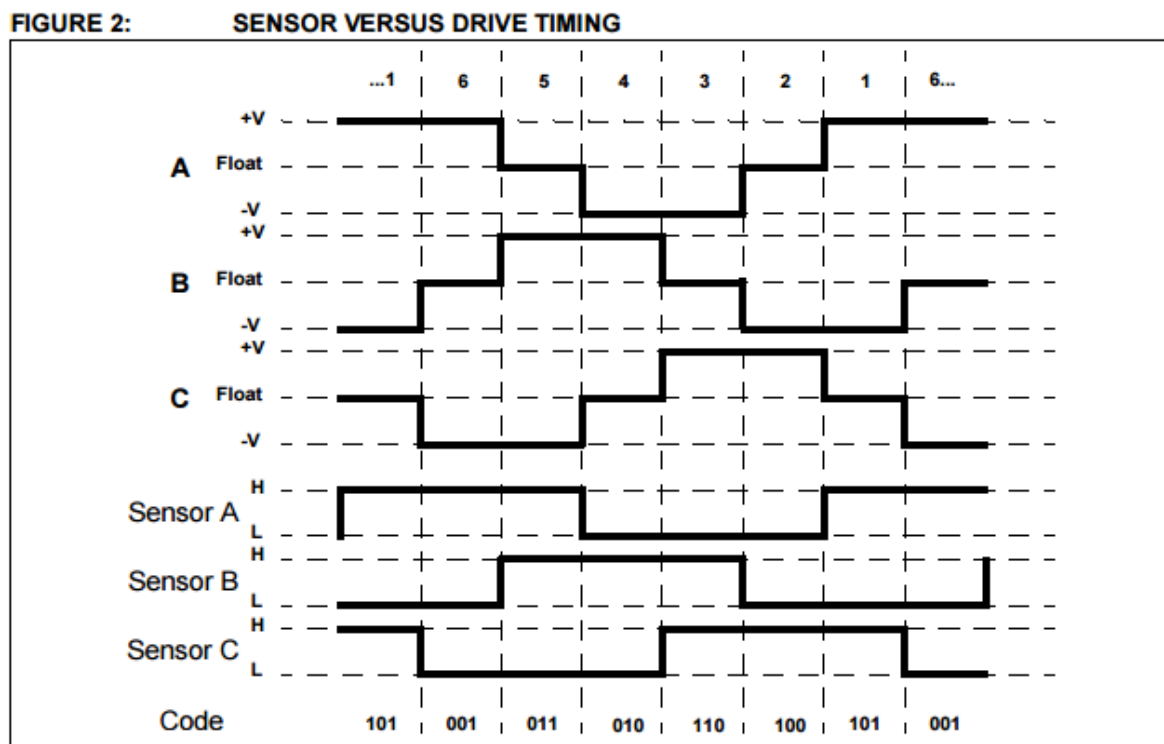
The easiest way to know the correct moment to commutate the winding currents is by means of a position sensor. Many BLDC motor manufacturers supply motors with a three-element Hall effect position sensor. Each sensor element outputs a digital high level for 180 electrical degrees of electrical rotation, and a low level for the other 180 electrical degrees.

권선 전류를 정류하기 위한 올바른 순간을 알 수 있는 가장 쉬운 방법은 위치 센서를 사용하는 것입니다. 많은 BLDC 모터 제조업체는 모터에 3 개 요소의 홀 효과 위치 센서를

제공합니다. 각 센서 소자는 180도의 전기적 회전 정도에 대해 디지털 하이 레벨을 출력하고 다른 180도 전기 레벨에 대해서는 로우 레벨을 출력합니다.

The three sensors are offset from each other by 60 electrical degrees so that each sensor output is in alignment with one of the electromagnetic circuits. A timing diagram showing the relationship between the sensor outputs and the required motor drive voltages is shown in Figure 2.

3 개의 센서는 60도 전기 각도로 서로 오프셋되어 각 센서 출력은 전자 회로 중 하나와 정렬됩니다. 센서 출력과 필요한 모터 구동 전압 사이의 관계를 보여주는 타이밍 다이어그램이 그림 2에 나와 있습니다.



The numbers at the top of Figure 2 correspond to the current phases shown in Figure 1. It is apparent from Figure 2 that the three sensor outputs overlap in such a way as to create six unique three-bit codes corresponding to each of the drive phases. The numbers shown around the peripheral of the motor diagram in Figure 1 represent the sensor position code. The north pole of the rotor points to the code that is output at that rotor position. The numbers are the sensor logic levels where the Most Significant bit is sensor C and the Least Significant bit is sensor A.

그림 2의 상단에있는 번호는 그림 1에 표시된 현재 단계에 해당합니다. 그림 2에서 알 수 있듯이, 3 개의 센서 출력은 각 구동 단계에 해당하는 6 개의 별 3 비트 코드를 생성하는 방식으로 중첩된다. 그림 1의 모터 다이어그램 주변에 표시된 숫자는 센서 위치 코드를 나타냅니다. 로터의 북극은 해당 로터 위치에서 출력되는 코드를 가리 킵니다. 숫자는 최상위 비트가 센서 C이며, 최하위 비트 센서 A이다 센서 논리 수준이다.

Each drive phase consists of one motor terminal driven high, one motor terminal driven low, and one motor terminal left floating. A simplified drive circuit is shown in Figure 3. Individual drive controls for the high and low drivers permit high drive, low drive, and floating drive at each motor terminal. One precaution that must be taken with this type of driver circuit is that both high side and low side drivers must never be activated at the same time. Pull-up and pull-down resistors must be placed at the driver inputs to ensure that the drivers are off immediately after a microcontroller RESET, when the microcontroller outputs are configured as high impedance inputs.

각 구동 단계는 하이로 구동되는 하나의 모터 단자, 로우로 구동되는 하나의 모터 단자 및 플로팅 상태로 남아있는 하나의 모터 단자로 구성됩니다. 단순화 된 드라이브 회로가 그림 3에 나와 있습니다. 높고 낮은 드라이버의 개별 드라이브 제어하여 각 모터 단자에서 높은 구동, 낮은 구동 및 부동 구동이 가능합니다. 이 유형의 드라이버 회로로 취해야 할 한 가지 주의 사항은 하이 사이드 및 로우 사이드 드라이버가 절대로 동시에 활성화되어서는 안된다는 것입니다. 마이크로 컨트롤러 출력이 하이 임피던스 입력으로 구성되어있을 때 마이크로 컨트롤러 RESET 직후에 드라이버가 꺼 지도록 풀업 및 풀다운 저항을 드라이버 입력에 배치해야 합니다.

Another precaution against both drivers being active at the same time is called dead time control. When an output transitions from the high drive state to the low drive state, the proper amount of time for the high side driver to turn off must be allowed to elapse before the low side driver is activated. Drivers take more time to turn off than to turn on, so extra time must be allowed to elapse so that both drivers are not conducting at the same time. Notice in Figure 3 that the high drive period and low drive period of each output, is separated by a floating drive phase period. This dead time is inherent to the three phase BLDC drive scenario, so special timing for dead time control is not necessary. The BLDC commutation sequence will never switch the high-side device and the low-side device in a phase, at the same time.

두 드라이버가 동시에 활성화 될 것에 대한 다른 예방책은 데드 타임 제어라고 합니다. 출력이 하이 드라이브 상태에서 로우 드라이브 상태로 전환되면, 하이 사이드 드라이버가 꺼지기 전에 적당한 시간이 경과해야만 로우 사이드 드라이버가 활성화됩니다. 드라이버는 전원을 끄면보다 시간이 걸리므로, 두 드라이버가 동시에 실행하지 않도록 여분의 시간이 경과되어야 한다. 그림 3에서 각 출력의 높은 드라이브 기간과 낮은 드라이브 기간은 플로팅 드라이브 위상 기간으로 구분됩니다 이 데드 타임은 3 상 BLDC 드라이브 시나리오에 고유 한 것이기 때문에, 데드 타임 제어를 위한 특별한 시간은 필요하지 않습니다. BLDC 정류 시퀀스는 하이 사이드 장치와 로우 사이드 장치로 같은 상에 같은 시간에 있을 수 없다.

At this point we are ready to start building the motor commutation control code. Commutation consists of linking the input sensor state with the corresponding drive state. e. This is best accomplished with a state table and a table offset pointer. The sensor inputs will form the table offset pointer, and the list of possible output drive codes will form the state table. Code development will be performed with a PIC16F877 in an ICD. I have arbitrarily assigned PORTC as the motor drive port and PORTE as the sensor input port. PORTC was chosen as the driver port because the ICD demo board also has LED indicators on that port so we can watch the slow speed commutation drive signals without any external test equipment. PORTC was chosen as the driver port because the ICD demo board also has LED indicators on that port so we can watch the slow speed commutation drive signals without any external test equipment.

이 시점에서 모터 정류 제어 코드를 만들 준비가 되었습니다. 정류는 입력 센서 상태와 해당 드라이브 상태를 연결하는 것으로 구성됩니다. 이것은 상태 테이블과 테이블 오프셋 포인터를 사용하여 실행하는 것이 좋습니다. 센서 입력은 테이블 오프셋 포인터를 형성하며 가능한 출력 드라이브 코드 목록은 상태 테이블을 형성합니다. 코드 개발은 ICD에서 PIC16F877로 수행됩니다. 필자는 모터 구동 포트로 PORTC를 센서 입력 포트로 PORTE를 임의로 할당했습니다. ICD 데모 보드에는 LED 표시등이 있어 외부 테스트 장비없이 저속 정류 드라이브 신호를 볼 수 있기 때문에 PORTC가 드라이버 포트로 선택되었습니다.

Each driver requires two pins, one for high drive and one for low drive, so six pins of PORTC will be used to control the six motor drive MOSFETS. Each sensor requires one pin, so three pins of PORTE will be used to read the current state of the motor's three-output sensor. The sensor state will be linked to the drive state by using the sensor input code as a binary offset to the drive table index. The sensor states and motor drive states from Figure 2 are tabulated in Table 1.

각 드라이버에는 2 개의 핀이 필요합니다. 하나는 하이 드라이브 용이고 다른 하나는 로우 드라이브 용입니다. 따라서 여섯 개의 PORTC 핀이 여섯 개의 모터 드라이브 MOSFET을 제어하는 데 사용됩니다. 각 센서에는 하나의 핀이 필요하므로 PORTE 핀 3 개를 사용하여 모터의 3 출력 센서의 현재 상태를 읽습니다. 센서 상태는 센서 입력 코드를 드라이브 테이블 인덱스의 바이너리 오프셋으로 사용하여 드라이브 상태에 연결됩니다. 그림 2의 센서 상태와 모터 구동 상태가 표 1에 요약되어 있습니다.

FIGURE 3: THREE PHASE BRIDGE

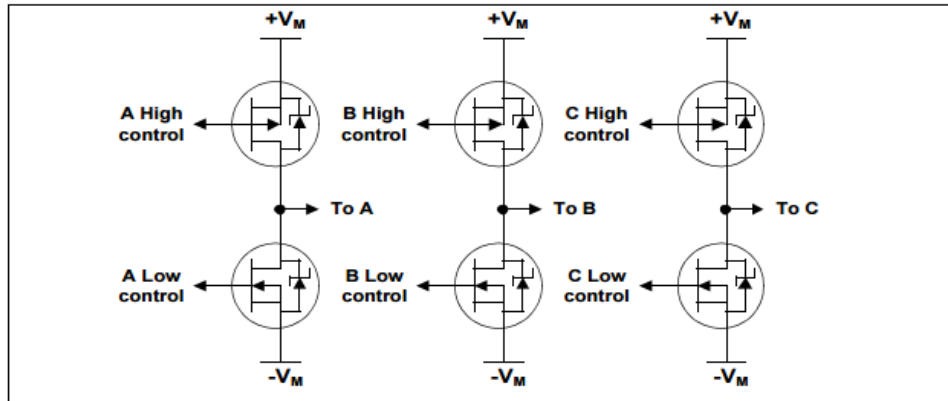


TABLE 1: CW SENSOR AND DRIVE BITS BY PHASE ORDER

Pin	RE2	RE1	RE0	RC5	RC4	RC3	RC2	RC1	RC0
Phase	Sensor C	Sensor B	Sensor A	C High Drive	C Low Drive	B High Drive	B Low Drive	A High Drive	A Low Drive
1	1	0	1	0	0	0	1	1	0
2	1	0	0	1	0	0	1	0	0
3	1	1	0	1	0	0	0	0	1
4	0	1	0	0	0	1	0	0	1
5	0	1	1	0	1	1	0	0	0
6	0	0	1	0	1	0	0	1	0

Sorting Table 1 by sensor code binary weight results in Table 2. Activating the motor drivers, according to a state table built from Table 2, will cause the motor of Figure 1 to rotate clockwise.

TABLE 2: CW SENSOR AND DRIVE BITS BY SENSOR ORDER

Pin	RE2	RE1	RE0	RC5	RC4	RC3	RC2	RC1	RC0
Phase	Sensor C	Sensor B	Sensor A	C High Drive	C Low Drive	B High Drive	B Low Drive	A High Drive	A Low Drive
6	0	0	1	0	1	0	0	1	0
4	0	1	0	0	0	1	0	0	1
5	0	1	1	0	1	1	0	0	0
2	1	0	0	1	0	0	1	0	0
1	1	0	1	0	0	0	1	1	0
3	1	1	0	1	0	0	0	0	1

Counter clockwise rotation is accomplished by driving current through the motor coils in the direction opposite of that for clockwise rotation. Table 3 was constructed by swapping all the high and low drives of Table 2. Activating the motor coils, according to a state table built from Table 3, will cause the motor to rotate counter clockwise. Phase numbers in Table 3 are preceded by a slash denoting that the EMF is opposite that of the phases in Table 2.

TABLE 3: CCW SENSOR AND DRIVE BITS

Pin	RE2	RE1	RE0	RC5	RC4	RC3	RC2	RC1	RC0
Phase	Sensor C	Sensor B	Sensor A	C High Drive	C Low Drive	B High Drive	B Low Drive	A High Drive	A Low Drive
/6	0	0	1	1	0	0	0	0	1
/4	0	1	0	0	0	0	1	1	0
/5	0	1	1	1	0	0	1	0	0
/2	1	0	0	0	1	1	0	0	0
/1	1	0	1	0	0	1	0	0	1
/3	1	1	0	0	1	0	0	1	0

The code segment for determining the appropriate drive word from the sensor inputs is shown in Figure 4.

FIGURE 4: COMMUTATION CODE SEGMENT

```

#define DrivePort      PORTC
#define SensorMask     B'00000111'
#define SensorPort     PORTE
#define DirectionBit   PORTA, 1

Commutate
    movlw  SensorMask      ;retain only the sensor bits
    andwf  SensorPort      ;get sensor data
    xorwf  LastSensor, w   ;test if motion sensed
    btfsc  STATUS, Z       ;zero if no change
    return                                ;no change - return

    xorwf  LastSensor, f   ;replace last sensor data with current
    btfss  DirectionBit    ;test direction bit
    goto   FwdCom         ;bit is zero - do forward commutation

                                ;reverse commutation
    movlw  HIGH RevTable   ;get MS byte to table
    movwf  PCLATH          ;prepare for computed GOTO
    movlw  LOW RevTable    ;get LS byte of table
    goto   Com2

FwdCom
    movlw  HIGH FwdTable   ;get MS byte of table
    movwf  PCLATH          ;prepare for computed GOTO
    movlw  LOW FwdTable    ;get LS byte of table

Com2
    addwf  LastSensor, w   ;add sensor offset
    btfsc  STATUS, C       ;page change in table?
    incf   PCLATH, f       ;yes - adjust MS byte

    call   GetDrive        ;get drive word from table
    movwf  DriveWord       ;save as current drive word
    return

GetDrive
    movwf  PCL

FwdTable
    retlw  B'00000000'     ;invalid
    retlw  B'00010010'     ;phase 6
    retlw  B'00001001'     ;phase 4
    retlw  B'00011000'     ;phase 5
    retlw  B'00100100'     ;phase 2
    retlw  B'00000110'     ;phase 1
    retlw  B'00100001'     ;phase 3
    retlw  B'00000000'     ;invalid

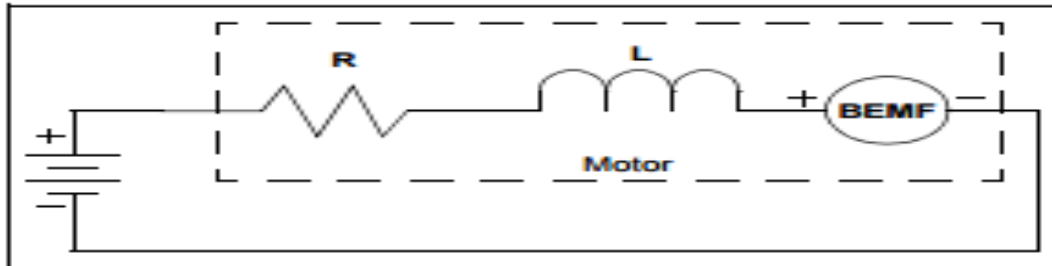
RevTable
    retlw  B'00000000'     ;invalid
    retlw  B'00100001'     ;phase /6
    retlw  B'00000110'     ;phase /4
    retlw  B'00100100'     ;phase /5
    retlw  B'00011000'     ;phase /2
    retlw  B'00001001'     ;phase /1
    retlw  B'00010010'     ;phase /3
    retlw  B'00000000'     ;invalid

```

Before we try the commutation code with our motor, let's consider what happens when a voltage is applied to a DC motor.

모터로 정류 코드를 사용하기 전에, DC모터에 전압이 가해 졌을 때 무슨 일이 발생하는지 생각해 보자. DC 모터의 크게 단순화 된 전기 모델이 그림 5에 나와 있습니다.

FIGURE 5: DC MOTOR EQUIVALENT CIRCUIT



When the rotor is stationary, the only resistance to current flow is the impedance of the electromagnetic coils. The impedance is comprised of the parasitic resistance of the copper in the windings, and the parasitic inductance of the windings themselves. The resistance and inductance are very small by design, so start-up currents would be very large, if not limited.

로터가 정지 상태 일 때, 전류 흐름에 대한 유일한 저항은 전자기 코일의 임피던스이다. 임피던스는 권선 내의 구리의 기생 저항 및 권선 자체의 기생 인덕턴스로 구성됩니다. 저항 및 인덕턴스는 설계 상 매우 작으므로 제한되지는 않더라도 시동 전류가 매우 커집니다.

When the motor is spinning, the permanent magnet rotor moving past the stator coils induces an electrical potential in the coils called Back Electromotive Force, or BEMF. BEMF is directly proportional to the motor speed and is determined from the motor voltage constant KV.

모터가 회전하는 동안 고정자 코일을 통과하여 이동하는 영구 자석 회 전자는 역기전력 또는 BEMF라는 코일의 전위를 유도한다. BEMF는 모터 속도에 직접 비례하며 모터 전압 상수 KV로 결정됩니다.

EQUATION 1:

$$\begin{aligned} \text{RPM} &= K_V \times \text{Volts} \\ \text{BEMF} &= \text{RPM} / K_V \end{aligned}$$

In an ideal motor, R and L are zero, and the motor will spin at a rate such that the BEMF exactly equals the applied voltage.

이상적인 모터에서, R과 L은 0이고, 모터는 BEMF가 적용된 전압과 정확하게 같아지는 속도로 회전 할 것입니다.

The current that a motor draws is directly proportional to the torque load on the motor shaft. Motor current is determined from the motor torque constant K_T .

모터가 인출하는 전류는 모터 축의 토크 부하에 정비례합니다. 모터 전류는 모터 토크 정수 K_T 로부터 결정됩니다.

EQUATION 2:

$$\text{Torque} = K_T \times \text{Amps}$$

An interesting fact about K_T and K_V is that their product is the same for all motors. Volts and Amps are expressed in MKS units, so if we also express K_T in MKS units, that is N-M/Rad/Sec, then the product of K_V and K_T is 1.

모터가 인출하는 전류는 모터 축의 토크 부하에 정비례합니다. 모터 전류는 모터 토크 정수 K_T 로부터 결정됩니다. 볼트 및 암페어는 MKS 단위로 표현되므로 K_T 를 MKS 단위로 표현하면 N-M / Rad / Sec이므로 K_V 와 K_T 의 곱은 1입니다.

EQUATION 3:

$$K_V \times K_T = 1$$

This is not surprising when you consider that the units of the product are $[1/(V \cdot A)] \cdot [(N \cdot M)/(Rad/Sec)]$, which is the same as mechanical power divided by electrical power.

이것은 제품의 단위가 $[1 / (V \cdot A)] \cdot [(N \cdot M) / (Rad / Sec)]$ 인 것으로 간주 할 때 놀라운 것이 아니며 기계력을 전력으로 나눈 것과 같습니다.

If voltage were to be applied to an ideal motor from an ideal voltage source, it would draw an infinite amount of current and accelerate instantly to the speed dictated by the applied voltage and K_V . Of course no motor is ideal, and the start-up current will be limited by the parasitic resistance and inductance of the motor windings, as well as the current capacity of the power source. Two detrimental effects of unlimited start-up current and voltage are excessive torque and excessive current. Excessive torque can cause gears to strip, shaft couplings to slip, and other undesirable mechanical problems. Excessive current can cause driver MOSFETS to blow out and circuitry to burn.

이상적인 전압원에서 이상적인 모터에 전압을 인가하면, 무한의 전류가 흘러인가 전압과 K_V 의해 결정 속도까지 순간적으로 가속합니다 물론 모터가 이상적이 아니며 시동 전류는 모터 권선의 기생 저항 및 인덕턴스 및 전원의 전류 용량에 의해 제한됩니다. 무제한 시동 전류 및 전압의 두 가지 유해한 영향은 과도한 토크 및 과도한 전류입니다. 과도한 토크로 인해 기어가 벗겨 지거나 샤프트 커플 링이 미끄러지거나 기타 바람직하지 않은 기계적 문제가 발생할 수 있습니다. 과도한 전류로 인해 드라이버 MOSFET이 폭발하고 회로가 소손 될 수 있습니다.

We can minimize the effects of excessive current and torque by limiting the applied voltage at start-up with pulse width modulation (PWM). Pulse width modulation is effective and fairly simple to do. Two things to consider with PWM are, the MOSFET losses due to switching, and the effect that the PWM rate has on the motor. Higher PWM frequencies mean higher switching losses, but too low of a PWM frequency will mean that the current to the motor will be a series of high current pulses instead of the desired average of the voltage waveform.

Averaging is easier to attain at lower frequencies if the parasitic motor inductance is relatively high, but high inductance is an undesirable motor characteristic. The ideal frequency is dependent on the characteristics of your motor and power switches. For this application, the PWM frequency will be approximately 10 kHz.

펄스 폭 변조 (PWM)로 시동시인가 전압을 제한함으로써 과도한 전류 및 토크의 영향을 최소화 할 수 있습니다. 펄스 폭 변조는 효과적이고 상당히 간단합니다. PWM으로 고려해야 할 두 가지 사항은 스위칭으로 인한 MOSFET 손실과 PWM 속도가 모터에 미치는 영향입니다. PWM 주파수가 높을수록 스위칭 손실은 크지 만, PWM 주파수가 너무 낮 으면 전압 파형의 바람직한 평균이 아닌 모터에 전류가 일련의 고전류 펄스가되는 것을 의미합니다. 기생 모터 인덕턴스가 상대적으로 높으면 낮은 주파수에서 평균화가 더 쉽지만 높은 인덕턴스는 바람직하지 않은 모터 특성입니다. 이상적인 주파수는 모터 및 전원 스위치의 특성에 따라 달라집니다. 이 응용 프로그램은 PWM 주파수는 10 kHz입니다.

We are using PWM to control start-up current, so why not use it as a speed control also? We will use the analog-to-digital converter (ADC), of the PIC16F877 to read a potentiometer and use the voltage reading as the relative speed control input. Only 8 bits of the ADC are used, so our speed control will have 256 levels. We want the relative speed to correspond to the relative potentiometer position. Motor speed is directly proportional to applied voltage, so varying the PWM duty cycle linearly from 0% to 100% will result in a linear speed control from 0% to 100% of maximum RPM. Pulse width is determined by continuously adding the ADC result to the free running Timer0 count to determine when the drivers should be on or off.

우리는 PWM을 사용하여 시동 전류를 제어하는데, 왜 속도 제어로 사용하지 않는가? PIC16F877의 아날로그 - 디지털 컨버터 (ADC)를 사용하여 전위차계를 읽고 상대 속도 제어 입력으로 전압 판독 값을 사용합니다. ADC의 8 비트 만 사용되므로 속도 제어는 256 레벨입니다. 우리는 상대 속도가 상대적 전위차계 위치에 대응 되기를 원합니다. 모터 속도는인가 전압에 정비례하므로 PWM 듀티 사이클을 0 %에서 100 %까지 선형 적으로 변화 시키면 최대 RPM의 0 %에서 100 %까지의 선형 속도 제어가 발생합니다.

If the addition results in an overflow, then the drivers are on, otherwise they are off. An 8-bit timer is used so that the ADC to timer additions need no scaling to cover the full range. To obtain a PWM frequency of 10 kHz Timer0 must be running at 256 times that rate, or 2.56 MHz. The minimum prescale value for Timer0 is 1:2, so we need an input frequency of 5.12 MHz. The input to Timer0 is FOSC/4. This requires an FOSC of 20.48 MHz. That is an odd frequency, and 20 MHz is close enough, so we will use 20 MHz resulting in a PWM frequency of 9.77 kHz.

추가로 오버 플로우가 발생하면 드라이버가 켜지고 그렇지 않은 경우는 해제됩니다. 8 비트 타이머를 사용하여 ADC 가산기는 전체 범위를 커버하는 스케일링을 필요로하지 않습니다. 10 kHz의 PWM 주파수를 얻으려면 Timer0가 해당 속도의 256 배 또는 2.56 MHz에서 실행되어야 합니다. Timer0의 최소 prescale 값은 1 : 2이므로 입력 주파수 5.12 MHz가 필요합니다. Timer0의 입력은 FOSC / 4입니다. 이를 위해서는 20.48 MHz의 FOSC가 필요합니다. 이것은 홀수 주파수이며, 20MHz 충분히 가깝기 때문에 20MHz를 사용하여 9.77kHz의 PWM 주파수를 사용합니다.

There are several ways to modulate the motor drivers. We could switch the high and low side drivers together, or just the high or low driver while leaving the other driver on. Some high side MOSFET drivers use a capacitor charge pump to boost the gate drive above the drain voltage. The charge pump charges when the driver is off and discharges into the MOSFET gate when the driver is on. It makes sense then to switch the high side driver to keep the charge pump refreshed. Even though this application does not use the charge pump type drivers, we will modulate the high side driver while leaving the low side driver on. There are three high side drivers, any one of which could be active depending on the position of the rotor.

The motor drive word is 6-bits wide, so if we logically AND the drive word with zeros in the high driver bit positions, and 1's in the low driver bit positions, we will turn off the active high driver regardless which one of the three it is.

모터 드라이버를 변조하는 방법에는 여러가지가 있습니다. 우리는 하이사이드 드라이버와 로우 사이드 드라이버를 함께 사용 할 수 있다. 또는 높거나 낮은 드라이버만 켜 채 또 다른 드라이버를 선택 할 수 있다. 일부 하이 사이드 MOSFET 드라이버는 게이트 구동 드레인 전압보다 높게하는 커패시터 차지 펌프를 사용합니다. 충전 펌프는 드라이버가 꺼져있을 때 충전되며 드라이버가 켜지면 MOSFET 게이트로 방전됩니다. 차지 펌프를 재생 상태로 유지하기 위해 하이 사이드 드라이버를 스위칭 하는 것이 좋다.. 이 응용 프로그램은 차지 펌프 타입의 드라이버는 사용하지 않지만, 로우 사이드 드라이버를 선택한 상태로하여 하이 사이드 드라이버를 변조 합니다. 3 개의 하이 사이드 드라이버가 있으며, 그 중 하나는 로터의 위치에 따라 활성화 될 수 있습니다. 모든 드라이브 워드는 6비트 폭이며, 따라서 높은 드라이버 비트 위치에 0을 낮은 드라이버 비트 위치에 1을 논리적으로 AND 하면, 우리는 3가지 중 어느 하나 관계없이 활성 하이 드라이버를 해제 합니다.

We have now identified 4 tasks of the control loop:

- Read the sensor inputs
- Commutate the motor drive connections
- Read the speed control ADC
- PWM the motor drivers using the ADC and Timer0 addition results

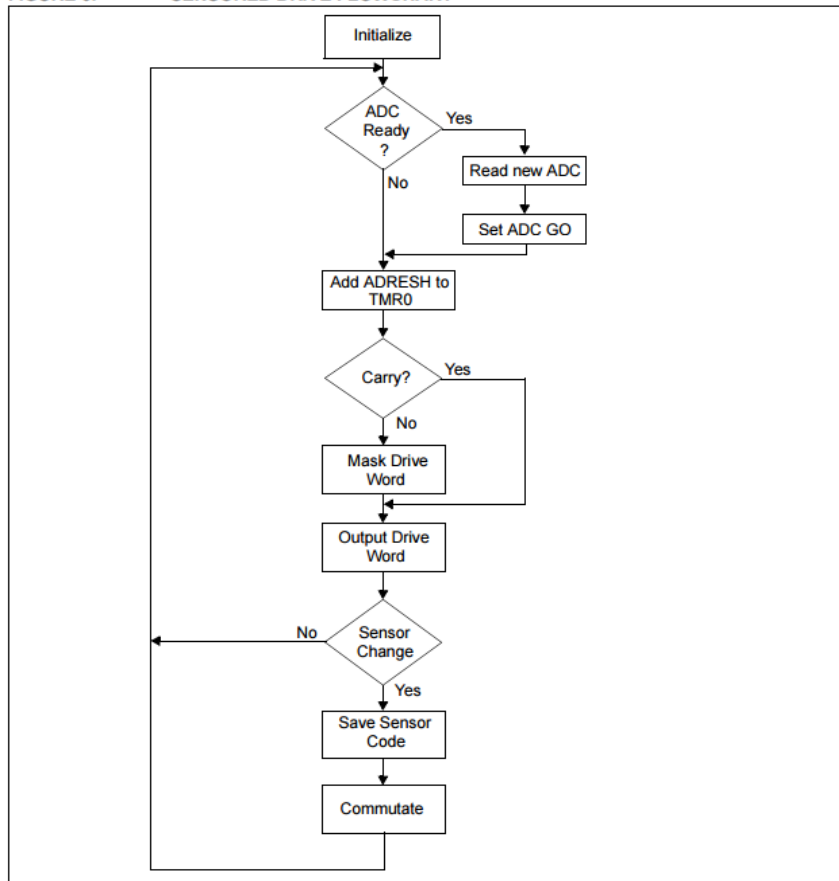
여기서 제어 루프의 4가지 작업을 확인했습니다 .

- 센서 입력을 읽어 들이기
- 모터 드라이브 연결을 정류
- 속도 제어 ADC를 읽기
- ADC와 Timer0 가산 결과를 사용하여 모터 드라이버를 PWM합니다.

At 20 MHz clock rate, control latency, caused by the loop time, is not significant so we will construct a simple polled task loop. The control loop flow chart is shown in Figure 6 and code listings are in Appendix B.

20MHz 클럭 속도는 루프 시간으로 인해 제어 지연 시간은 중요하지 않기 때문에 간단한 폴링 된 작업 루프를 구축합니다. 제어 루프 흐름도는 그림 6에 나와 있으며 코드 목록은 부록 B에 나와 있습니다.

FIGURE 6: SENSED DRIVE FLOWCHART



Sensorless Motor Control

It is possible to determine when to commutate the motor drive voltages by sensing the back EMF voltage on an undriven motor terminal during one of the drive phases. The obvious cost advantage of sensorless control is the elimination of the Hall position sensors. There are several disadvantages to sensorless control:

- The motor must be moving at a minimum rate to generate sufficient back EMF to be sensed
- Abrupt changes to the motor load can cause the BEMF drive loop to go out of lock
- The BEMF voltage can be measured only when the motor speed is within a limited range of the ideal commutation rate for the applied voltage
- Commutation at rates faster than the ideal rate will result in a discontinuous motor response

구동 단계 중 하나에서 구동되지 않은 모터 단자의 역기전력 전압을 감지하여 모터 구동 전압을 정류 할시기를 결정할 수 있습니다. 센서리스 제어의 비용 이점은 홀 위치 센서를 제거하는 것입니다. 센서리스 제어에는 몇 가지 단점이 있습니다.

- 모터는 감지 될 정도로 역기전력을 생성하기 위해 최저 속도로 이동해야한다
- 모터의 부하가 급격하게 변화하면 BEMF 드라이브루프가 잠금 상태가 될 수 있습니다.
- BEMF 전압은 모터 속도가인가 전압에 대한 이상적인 정류 율의 제한된 범위 내에있을 때만 측정 할 수 있습니다.

- 이상 속도보다 빠른 속도로 정류는 불연속적인 모터 응답을 발생합니다.

If low cost is a primary concern and low speed motor operation is not a requirement and the motor load is not expected to change rapidly then sensorless control may be the better choice for your application.

저비용이 주요 관심사이고 저속 모터 작동이 요구 사항이 아니며 모터 부하가 급격하게 변화하지 않을 것으로 예상되면 센서리스 제어가 어플리케이션에 더 적합한 선택 일 수 있습니다.

Determining the BEMF

EQUATION 4:

$$B_{BEMF} = \sin(\alpha)$$

EQUATION 5:

$$C_{BEMF} = \sin\left(\alpha - \frac{2\pi}{3}\right)$$

EQUATION 6:

$$A_{BEMF} = \sin\left(\alpha - \frac{4\pi}{3}\right)$$

FIGURE 7: BEMF EQUIVALENT CIRCUIT

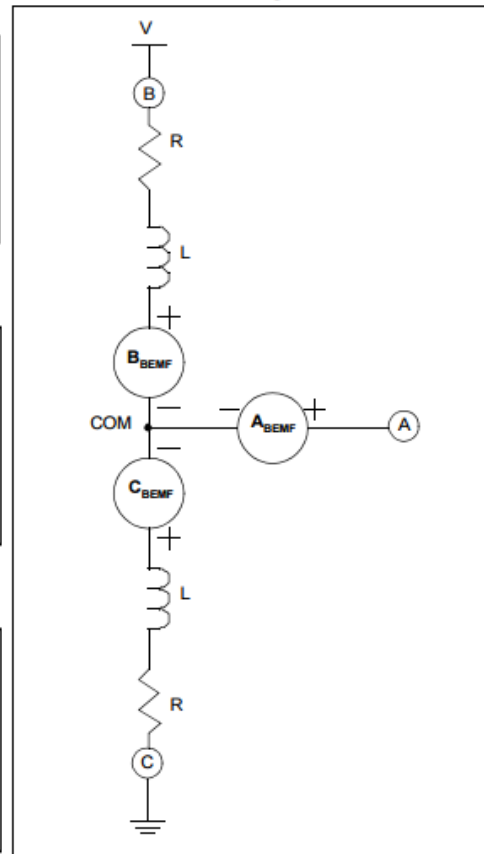


Figure 7 shows the equivalent circuit of the motor with coils B and C driven while coil A is undriven and available for BEMF measurement. At the commutation frequency the L's are negligible. The R's are assumed to be equal. The L and R components are not shown in the A branch since no significant current flows in this part of the circuit so those components can be ignored.

그림 7은 코일 A가 구동되지 않고 BEMF 측정이 가능한 동안 코일 B와 C가 구동되는 모터의 등가 회로를 보여준다. 정류 주파수에서 L은 무시할 수 있습니다. R은 동일하다고 가정된다. 회로의 이 부분에는 큰 전류가 흐르지 않으므로 L 및 R 구성 요소는 A 분기에 표시되지 않으므로 이러한 구성 요소는 무시할 수 있습니다.

The BEMF generated by the B and C coils in tandem, as shown in Figure 7, can be expressed as shown in Equation 7.

그림 7에 나타낸 바와 같이, B 코일 및 C 코일에 의해 직렬로 생성되는 BEMF는 식 7과 같이 나타낼 수 있다.

EQUATION 7:

$$BEMF_{BC} = B_{BEMF} - C_{BEMF}$$

The sign reversal of CBEMF is due to moving the reference point from the common connection to ground. Recall that there are six drive phases in one electrical revolution. Each drive phase occurs +/- 30 degrees around the peak back EMF of the two motor windings being driven during that phase. At full speed the applied DC voltage is equivalent to the RMS BEMF voltage in that 60 degree range. In terms of the peak BEMF generated by any one winding, the RMS BEMF voltage across two of the windings can be expressed as shown in Equation 8.

CBEMF의 부호 반전은 기준점을 공통 연결에서 접지로 이동하기 때문입니다. 1회 전기 회전에서 6개의 구동 단계가 있다는 것을 상기 한다, 각각의 구동 단계는 그 단계 동안 구동되는 2 개의 모터 권선의 피크 역기전 (back EMF)+/- 30도에서 발생합니다. 최대 속도에서 적용되는 DC 전압은 60도 범위의 RMS BEMF 전압과 동일합니다. 어떤 하나의 권선에 의해 생성 된 피크 BEMF에 대해 권선의 2개의 끝의 RMS BEMF전압은 식 8과 같이 나타낼 수 있다.

EQUATION 8:

$$BEMF_{RMS} = \sqrt{\frac{3}{\pi} \int_{\frac{\pi}{6}}^{\frac{\pi}{2}} \left(\sin(\alpha) - \sin\left(\alpha - \frac{2\pi}{3}\right) \right)^2 d\alpha}$$

$$BEMF_{RMS} = \sqrt{\frac{3}{\pi} \left(\frac{\pi}{2} + \frac{3\pi}{4} \right)}$$

$$BEMF_{RMS} = 1.6554$$

We will use this result to normalize the BEMF diagrams presented later, but first lets consider the expected BEMF at the undriven motor terminal. Since the applied voltage is pulse width modulated, the drive alternates between on and off throughout the phase time.

The BEMF, relative to ground, seen at the A terminal when the drive is on, can be expressed as shown in Equation 9.

이 결과를 사용하여 나중에 제시되는 BEMF 다이어그램을 정규화하지만 먼저 구동되지 않은 모터 단자에서 예상되는 BEMF를 고려합니다. 인가 된 전압은 펄스 폭으로 변조되기 때문에, 구동은 위상 시간 동안 온 (on)과 오프 (off)를 교대로 반복합니다. 드라이브가 켜져있을 때 A 단자에서 볼 때 BEMF는 접지에 대해 식 9와 같이 나타낼 수 있습니다..

EQUATION 9:

$$BEMF_A = \frac{[V - (B_{BEMF} - C_{BEMF}))R}{2R} - C_{BEMF} + A_{BEMF}$$

$$BEMF_A = \frac{V - B_{BEMF} + C_{BEMF}}{2} - C_{BEMF} + A_{BEMF}$$

Notice that the winding resistance cancels out, so resistive voltage drop, due to motor torque load, is not a factor when measuring BEMF. The BEMF, relative to ground, seen at the A terminal when the drive is off can be expressed as shown in Equation 10.

권선 저항이 상쇄되므로 모터 토크 부하로 인한 저항 전압 강하가 BEMF를 측정 할 때 중요한 요인이되지 않습니다. 드라이브가 꺼져있을 때 A 터미널에 보이는 지상에 대한 BEMF는 식 10과 같이 나타낼 수 있습니다.

EQUATION 10:

$$BEMF_A = A_{BEMF} - C_{BEMF}$$

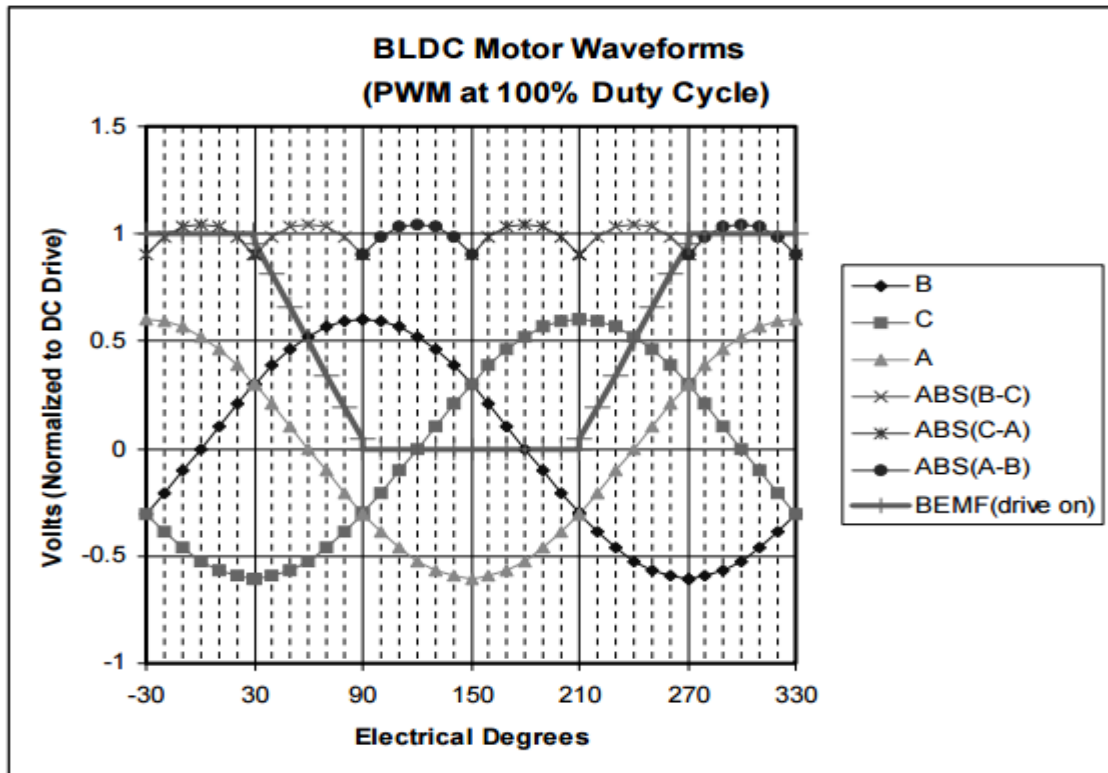
Figure 8 is a graphical representation of the BEMF formulas computed over one electrical revolution. To avoid clutter, only the terminal A waveform, as would be observed on a oscilloscope is displayed and is denoted as BEMF(drive on).

그림 8은 하나의 전기 회전에 대해 계산 된 BEMF 공식의 그래픽 표현입니다. 혼란을 피하기 위해 오실로스코프로 관측되는 같은 단자 A의 파형 만 표시되고 BEMF (드라이브 선택)가 표시됩니다.

The terminal A waveform is flattened at the top and bottom because at those points the terminal is connected to the drive voltage or ground. The 60 degree sinusoidal humps are the BEMFs of the driven coil pairs relative to ground. The entire graph has been normalized to the RMS value of the coil pair BEMFs.

이러한 점에서 터미널(단자)가 구동 전압 또는 접지에 연결되어 있기 때문에, 단자 A의 파형은 상단과 하단으로 평탄하게됩니다. 정현파 파형은 코일 공통 연결 지점을 기준으로 한 개별 코일 BEMF입니다. 60도 정현파의 분위기는 지상에 움직이는 코일 대 BEMF이다.전체 차트는 코일 대 BEMF의 RMS 값으로 정규화되어 있습니다.

FIGURE 8: BEMF AT 100% DRIVE



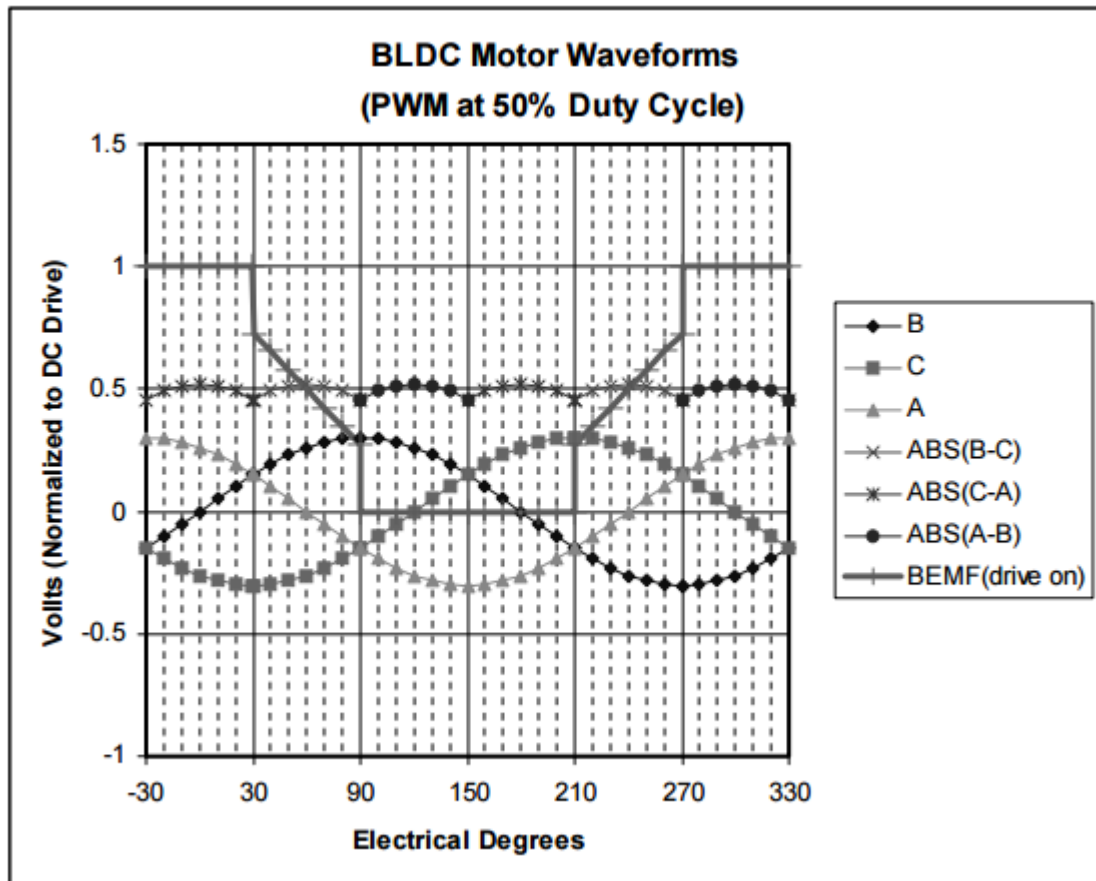
Notice that the BEMF(drive on) waveform is fairly linear and passes through a voltage that is exactly half of the applied voltage at precisely 60 degrees which coincides with the zero crossing of the coil A BEMF waveform. This implies that we can determine the rotor electrical position by detecting when the open terminal voltage equals half the applied voltage.

BEMF (드라이브 커짐) 파형은 상당히 선형이며 코일 A BEMF 파형의 제로 크로싱과 일치하는 정확히 60도에서 정확히인가 된 전압의 절반 인 전압을 통과합니다. 이것은 개방 단자 전압이인가 전압의 절반에 해당 때를 감지하여 로터의 전기적 위치를 결정할 수 있다는 것을 의미한다.

What happens when the PWM duty cycle is less than 100%? Figure 9 is a graphical representation of the BEMF formulas computed over one electrical revolution when the effective applied voltage is 50% of that shown in Figure 8. The entire graph has been normalized to the peak applied voltage.

PWM 듀티 사이클이 100%보다 작으면 어떻게 됩니까? 그림 9는 유효 적용 전압이 그림 8에 표시된 전압의 50 % 인 경우 한 번의 전기 회전에 대해 계산 된 BEMF 공식을 그래픽으로 나타낸 것입니다. 전체 그래프는 피크 적용 전압으로 정규화되었습니다.

FIGURE 9: BEMF AT 50% DRIVE

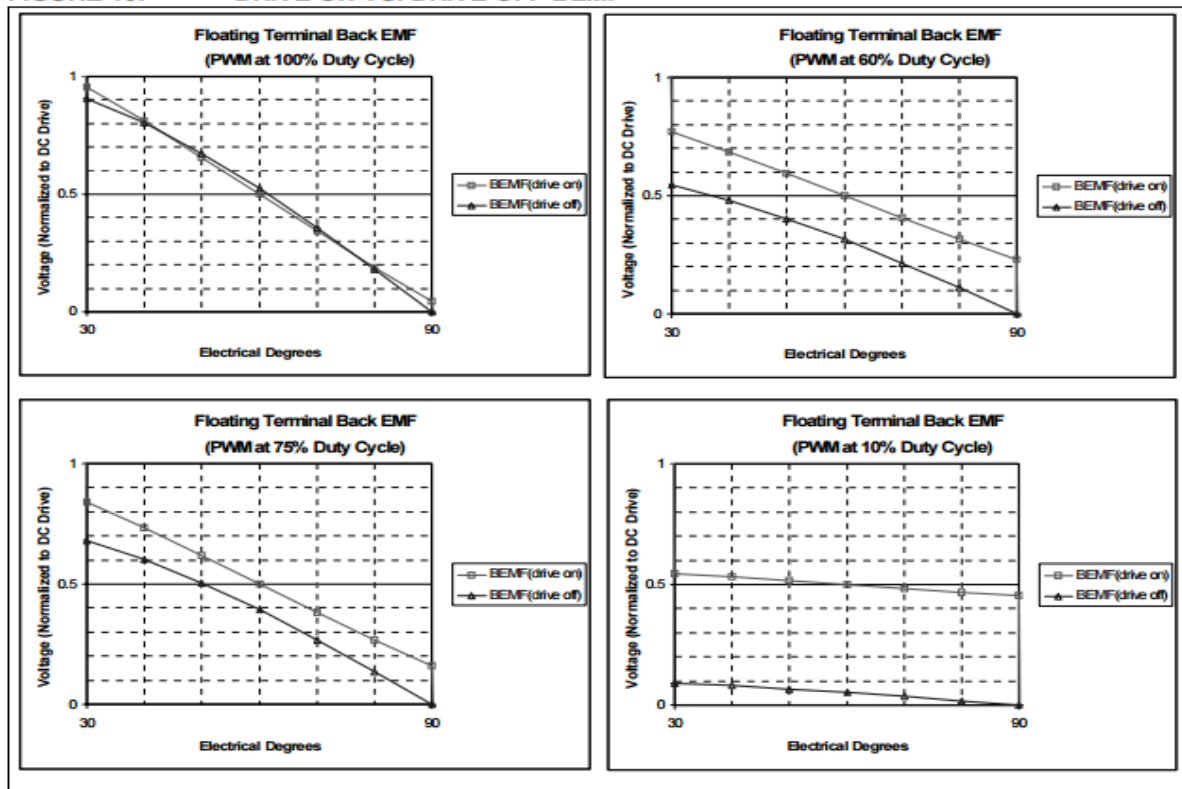


As expected the BEMF waveforms are all reduced proportionally but notice that the BEMF on the open terminal still equals half the applied voltage midway through the 60 degree drive phase. This occurs only when the drive voltage is on. Figure 10 shows a detail of the open terminal BEMF when the drive voltage is on and when the drive voltage is off. At various

duty cycles, notice that the drive on curve always equals half the applied voltage at 60 degrees.

예상대로 BEMF 파형은 모두 비례하여 감소하지만 열린 터미널(단자)의 BEMF는 60도 구동 단계의 중간에서 적용 전압의 절반에 해당합니다. 이것은 드라이브 전압이 켜져있을 때만 발생합니다. 그림 10은 드라이브 전압이 켜져 있고 드라이브 전압이 꺼져있을 때 열린 터미널 BEMF의 세부 정보를 보여줍니다. 다양한 듀티 사이클에서 곡선의 드라이브는 항상 60도에서인가 된 전압의 절반에 해당합니다.

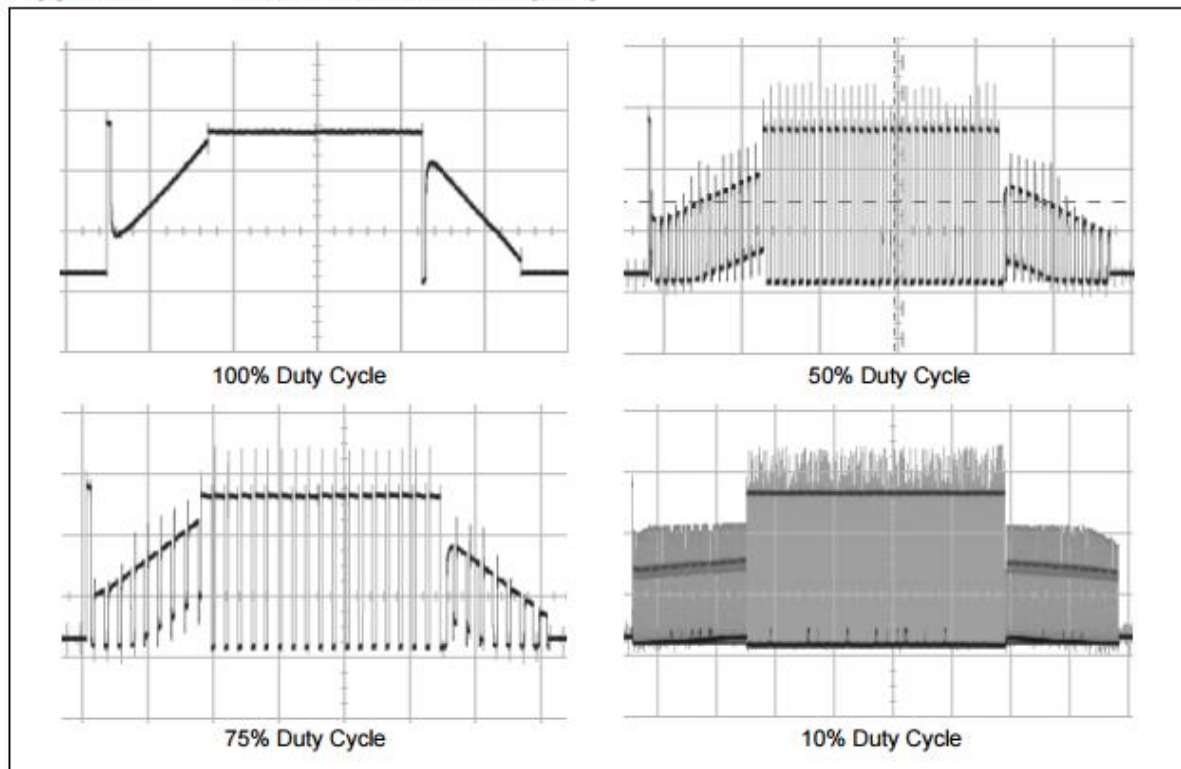
FIGURE 10: DRIVE ON VS. DRIVE OFF BEMF



How well do the predictions match an actual motor? Figure 11 shows the waveforms present on terminal A of a Pittman N2311A011 brushless motor at various PWM duty cycle configurations. The large transients, especially prevalent in the 100% duty cycle waveform, are due to flyback currents caused by the motor winding inductance.

예상이 실제 모터와 얼마나 잘 일치하고 있습니까? 그림 11은 다양한 PWM 듀티 사이클 구성에서 Pittman N2311A011 브러시리스 모터의 단자 A에 나타나는 파형을 보여줍니다. 100% 듀티 사이클 파형에서 특히 많이 나타나는 과도 현상은 모터 권선 인덕턴스로 인한 플라이백 전류로 인한 것입니다.

FIGURE 11: PITTMAN BEMF WAVEFORMS



The rotor position can be determined by measuring the voltage on the open terminal when the drive voltage is applied and then comparing the result to one half of the applied voltage. Recall that motor speed is proportional to the applied voltage. The formulas and graphs presented so far represent motor operation when commutation rate coincides with the effective applied voltage. When the commutation rate is too fast then commutation occurs early and the zero crossing point occurs later in the drive phase. When the commutation rate is too slow then commutation occurs late and the zero crossing point occurs earlier in the drive phase. We can sense and use this shift in zero crossing to adjust the commutation rate to keep the motor running at the ideal speed for the applied voltage and load torque.

회 전자의 위치는 구동 전압이인가 된 때 개방 단자 전압을 측정하여 그 결과를인가 전압의 절반과 비교하여 결정할 수있다. 모터 속도는 인가 전압에 비례 함을 상기 하십시오. 지금까지 제시된 공식및 그래프는 정류 율이 유효 전류 전압과 일치 할 때 모터 작동을 나타냅니다. 정류 율이 너무 빠르면 정류가 일찍 일어나고 제로 교차점은 구동 단계의 후반에 발생합니다. 정류 율이 너무 느리면 정류가 늦게 발생하고 제로 교차점이 구동 단계의 초기에 발생한다 . 우리는 영점 교차에서이 시프트를 감지하고 사용하여인가 된 전압 및 부하 토크에 대해 이상적인 속도로 모터가 계속 작동하도록 정류 율을 조정할 수 있습니다.

Open Loop Speed Control

An interesting property of brushless DC motors is that they will operate synchronously to a certain extent. This means that for a given load, applied voltage, and commutation rate the motor will maintain open loop lock with the commutation rate provided that these three variables do not deviate from the ideal by a significant amount. The ideal is determined by the motor voltage and torque constants. How does this work? Consider that when the commutation rate is too slow for an applied voltage, the BEMF will be too low resulting in more motor current. The motor will react by accelerating to the next phase position then slow down waiting for the next commutation.

브러시리스 DC 모터의 흥미로운 특성은 일정 수준까지 동기적으로 작동한다는 것입니다. 이것은 주어진 부하인가 전압 및 정류 속도에 모터는 이러한 세 가지 변수가 이상에서 벗어나지 않는 한, 정류 속도에서 개방 루프 잠금을 유지한다는 것을 의미합니다. 이상은 모터 전압 및 토크 상수에 의해 결정됩니다. 어떻게 작동합니까? 고려한 정류 속도가 인가 전압에 비해 너무 느리면 BEMF가 너무 낮아 모터 전류가 증가합니다. 모터는 다음 위상의 위치로 가속하여 반응하고 다음 정류기 기다리면서 속도를 감속합니다.

In the extreme case the motor will snap to each position like a stepper motor until the next commutation occurs. Since the motor is able to accelerate faster than the commutation rate, rates much slower than the ideal can be tolerated without losing lock but at the expense of excessive current. Now consider what happens when commutation is too fast.

극단적인 경우, 모터는 다음 정류가 발생할때까지 스텝퍼 모터처럼 각 위치에 스냅됩니다. 모터가 정류 속도보다 빠르게 가속 할 수 있기 때문에, 이상적인 모터 속도보다 훨씬 느린 속도가 허용되지만, 과도한 전류가 희생된다. 이제 정류가 너무 빠르면 어떻게 되는지 생각해봅시다.

When commutation occurs early the BEMF has not reached peak resulting in more motor current and a greater rate of acceleration to the next phase but it will arrive there too late. The motor tries to keep up with the commutation but at the expense of excessive current. If the commutation arrives so early that the motor can not accelerate fast enough to catch the next commutation, lock is lost and the motor spins down.

정류가 일찍 발생하면 BEMF는 정점에 도달하지 못하여 모터 전류가 증가하고 다음 단계로의 가속 속도는 빨라지지만 너무 늦게 도착합니다. 모터는 정류를 유지하려고 하지만 과대한

정류를 희생하고 있다. 정류가 매우 일찍 도착한 경우 모터다음 정류를 포착 할 수있을만큼 빠르게 가속하지 않는 잠금이 손실되고 모터가 회전합니다.

This happens abruptly not very far from the ideal rate. The abrupt loss of lock looks like a discontinuity in the motor response which makes closed loop control difficult. An alternative to closed loop control is to adjust the commutation rate until self locking open loop control is achieved. This is the method we will use in our application.

이것은 이상적인 속도에서 벗어나지 않더라도 갑자기 발생합니다. 갑작스러운 잠금 장치의 손실은 모터 응답의 불연속처럼 보이므로 패쇄 루프 제어가 어려워집니다. 패쇄 루프 제어의 대안은 자체 자금 개방 루프 제어가 달성 될 때까지 정류 율을 조정하는 것입니다 이것이 우리가 응용 프로그램에서 사용하는 방법입니다.

When the load on a motor is constant over it's operating range then the response curve of motor speed relative to applied voltage is linear. If the supply voltage is well regulated, in addition to a constant torque load, then the motor can be operated open loop over it's entire speed range. Consider that with pulse width modulation the effective voltage is linearly proportional to the PWM duty cycle. An open loop controller can be made by linking the PWM duty cycle to a table of motor speed values stored as the time of commutation for each drive phase. We need a table because revolutions per unit time is linear, but we need time per revolution which is not linear. Looking up the time values in a table is much faster than computing them repeatedly.

모터의 부하가 동장 범위에서 일정 할 때, 인가 전압에 대한 모터 속도 응답 곡선은 선형(직선)입니다. 공급 전압이 일정한 토크 부하 이외에도 잘 조절되면 코머튼 모든 속도 범위에 걸쳐 오픈 루프에서 작동할 수 있습니다. 펄스 폭 변조에서 유효 전압은 PWM 듀티 사이클에 선형 적으로 비례한다는 것을 고려하십시오. 개개의 루프 컨트롤러는 PWM 듀티 사이클을 각 구동 단계의 정류시간으로 기억 된 모터 속도 값의 테이블에 연결하여 할 수 있다. 단위 시간당 회전 수는 선형이므로 테이블이 필요하지만 선형이 아닌 1회전당 시간이 필요합니다. 테이블에서 시간 값을 알아내는 것은 그들을 반복 계산하는 것보다 훨씬 빠릅니다.

The program that we use to run the motor open loop is the same program we will use to automatically adjust the commutation rate in response to variations in the torque load. The program uses two potentiometers as speed control inputs. One potentiometer, we'll call it the PWM potentiometer, is directly linked to both the PWM duty cycle and the commutation time lookup table.

모터 개방 루프를 실행하는데 사용하는 프로그램은 토크 부하의 변동에 따라 자동으로 정류 속도를 조정하는데 사용하는 것과 동일한 프로그램이다. 이 프로그램은 속도 제어 입력으로

2개의 전위차계를 사용합니다. 전위차계 중 하나 인 PWM 포텐쇼미터는 PWM 듀티 사이클과 정류 시간 조회 테이블에 직접 연결됩니다.

The second potentiometer, we'll call this the Offset potentiometer, is used to provide an offset to the PWM duty cycle determined by the PWM potentiometer. An analog-to-digital conversion of the PWM potentiometer produces a number between 0 and 255. The PWM duty cycle is generated by adding the PWM potentiometer reading to a free running 8-bit timer. When the addition results in a carry the drive state is on, otherwise it is off. The PWM potentiometer reading is also used to access the 256 location commutation time lookup table. The Offset potentiometer also produces a number between 0 and 255. The Most Significant bit of this number is inverted making it a signed number between -128 and 127. This offset result, when added to the PWM potentiometer, becomes the PWM duty cycle threshold, and controls the drive on and off states described previously.

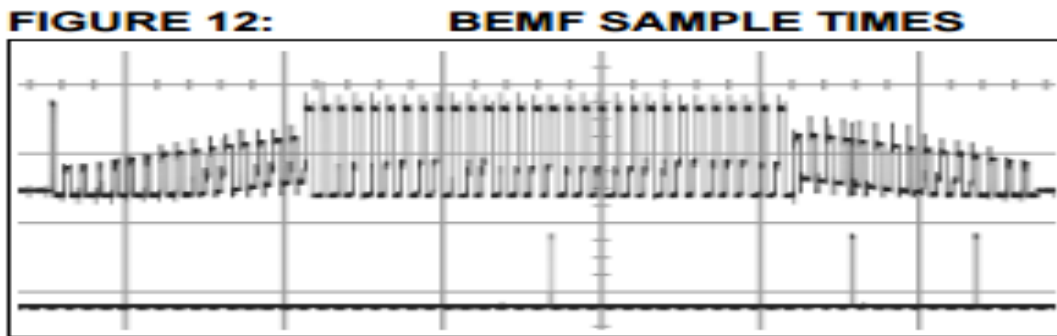
두번째 포텐서미터(오프셋 포텐서미터)는 PWM 전위차계에 의해 결정되는 PWM 듀티 사이클 오프셋을 제공하는 데 사용됩니다. PWM 전위차계의 아날로그 - 디지털 변환은 0과 255 사이의 수를 생성합니다. PWM 듀티 사이클은 프리 러닝 8 비트 타이머에 PWM 전위차계 판독 값을 추가하여 생성합니다. 가산 결과가 캐리의 경우 드라이 상태는 켜져있고, 그렇지 않으면 꺼집니다. PWM 포텐쇼미터 판독 값은 또한 256 위치 정류 시간 룩업 테이블에 액세스 하는데 사용됩니다. 오프셋 포텐쇼미터는 또한 0과 255 사이의 수치를 생성합니다. 이 숫자의 Most Significant 비트는 반전되어 -128에서 127 사이의 부호가 있는 숫자가 됩니다. 이 오프셋 결과는 PWM 포텐쇼미터에 추가 될 때, PWM 듀티 사이클 임계값이 되어, 앞에서 말한 드라이브의 켜기 및 끄기 상태를 제어합니다.

Closed Loop Speed Control (폐쇄 루프 속도 제어)

Closed loop speed control is achieved by unlinking the commutation time table index from the PWM duty cycle number. The PWM potentiometer is added to a fixed manual threshold number between 0 and 255. On entering Automatic mode the commutation index is initially set to the PWM potentiometer reading. Thereafter, as long as Automatic mode is still in effect, the commutation table index is automatically adjusted up or down according to voltages read at motor terminal A at specific times. Three voltage readings are taken.

폐쇄 루프 속도 제어는 정류 시간 테이블 인덱스와 PWM 듀티 사이클과의 연결을 해제함으로써 달성된다. PWM 전위차계는 0 ~ 255 사이의 고정 수동 임계 수에 추가됩니다. 이 가산결과가 캐리되면 모드가 자동으로 전환됩니다. 자동 모드에 들어가면 커뮤 테이블 인덱스는 최초로 PWM 전위차계 판독 값으로 설정됩니다. 그 후, 자동 모드가 아직

활성화되어있는 한, 정류 테이블 인덱스는 모터 단자 A에서 특정 시간에 판독 된 전압에 따라 자동으로 위 또는 아래로 조정된다.3가지 전압 판독값이 사용된다.



The first reading is taken during drive phase 4 when terminal A is actively driven high. This is the applied voltage. The next two readings are taken during drive phase 5 when terminal A is floating. The first reading is taken when $\frac{1}{4}$ of the commutation time has elapsed and the second reading is taken when $\frac{3}{4}$ of the commutation time has elapsed. We'll call these readings 1 and 2 respectively. The commutation table index is adjusted according to the following relationship between the applied voltage reading and readings 1 and 2:

- Index is unchanged if Reading 1 > Applied Voltage/2 and Reading 2 < Applied Voltage/2
- Index is increased if Reading 1 < Applied Voltage/ 2
- Index is decreased if Reading 1 > Applied Voltage/2 and Reading 2 > Applied Voltage/2

첫번째 판독은 단자 A가 능동적으로 하이로 구동되는 구동 단계 4에서 수행됩니다.

이것은 적용된 전압입니다. 다음 두 판독 값은 단자 A가 플로팅 상태 일 때의 구동 단계 5 중에 이루어진다. 첫 번째 읽기는 정류 시간의 1/4이 경과 한 때 이루어지고 제 2의 읽기는 정류 시간의 3/4이 경과 한 때 이루어집니다. 우리는 이러한 판독 값 1과 2를 각각 부릅니다. 정류 테이블의 인덱스는 인가 된 전압의 판독과 판독 값 1 사이의 다음 관계에 따라 조정됩니다.

- 읽기 1>인가 전압 / 2 및 읽기 2 <인가 전압 / 2의 경우, 인덱스는 변경되지 않습니다
- 읽기 1 <인가 전압 / 2의 경우 인덱스가 증가합니다
- 읽기 1>인가 전압 / 2 및 읽기 2>인가 전압 / 2의 경우 인덱스가 감소합니다

The motor rotor and everything it is connected to has a certain amount of inertia. The inertia delays the motor response to changes in voltage load and commutation time. Updates to the commutation time table index are delayed to compensate for the mechanical delay and allow the motor to catch up.

모터 로터와 이에 연결되어있는 모든 것에 일정한 관성이 있습니다. 관성은 전압 부하 및 정류 시간의 변화에 대한 모터 응답을 지연시킵니다. 정류 시간 테이블 인덱스의 업데이트는 기계적 지연을 보상하고 모터가 따라 잡기를 늦추기 위해 지연된다.

Acceleration and Deceleration Delay(가속 및 감속)

The inertia of the motor and what it is driving, tends to delay motor response to changes in the drive voltage. We need to compensate for this delay by adding a matching delay to the control loop. The control loop delay requires two time constants, a relatively slow one for acceleration, and a relatively fast one for deceleration.

모터의 관성과 그것이 작동하는 것은 구동 전압의 변화에 대한 모터의 응답을 지연하는 경향이 있습니다. 이러한 지연을 제어 루프에 매칭 지연을 추가하여 보완해야 합니다. 제어 루프 지연은 2 개의 시정이 필요하며, 가속도에 대해서는 비교적 느린 감속 대해서는 비교적 빠른 것이 있습니다.

Consider what happens in the control loop when the voltage to the motor suddenly rises, or the motor load is suddenly reduced. The control senses that the motor rotation is too slow and attempts to adjust by making the commutation time shorter. Without delay in the control loop, the next speed measurement will be taken before the motor has reacted to the adjustment, and another speed adjustment will be made. Adjustments continue to be made ahead of the motor response until eventually, the commutation time is too short for the applied voltage, and the motor goes out of lock. The acceleration timer delay prevents this runaway condition. Since the motor can tolerate commutation times that are too long, but not commutation times that are too short, the acceleration time delay can be longer than required without serious detrimental effect.

모터로의 전압이 갑자기 상승하거나 모터 부하가 갑자기 감소하면 제어 루프에서 어떤 일이 발생하는지 고려하십시오. 제어 장치는 모터 회전이 너무 느린(낮은) 것을 감지하고 정류 시간을 단축해서 조정을 시도한다. 제어 루프에서 지연없이 모터가 조정에 반응하기 전에 다음 속도 측정이 수행되고 다른 속도 조정이 수행됩니다. 최종적으로 정류 시간이 너무 짧아서 모터가 잠기지 않을 때까지 모터 응답에 앞서 조정이 계속됩니다. 가속 타이머 지연은 이 런 어웨이(폭주?) 상태를 방지합니다. 모터는 너무 긴 정류 시간을 견딜 수 있기 때문에, 너무 짧은 정류 시간은 아니지만, 가속 시간 지연은 심각한 해로운 영향없이 요구되는 것보다 길어질 수 있습니다.

Consider what happens in the control loop when the voltage to the motor suddenly falls, or the motor load is suddenly increased. If the change is sufficiently large, commutation time will immediately be running too short for the motor conditions. The motor cannot tolerate this, and loss of lock will occur. To prevent loss of lock, the loop deceleration timer delay must be short enough for the control loop to track, or precede the changing motor condition. If the time delay is too short, then the control loop will continue to lengthen the commutation time ahead of the motor response resulting in over compensation. The motor will eventually slow to a speed that will indicate to the BEMF sensor that the speed is too slow for the applied voltage. At that point, commutation deceleration will cease, and the commutation change will adjust in the opposite direction governed by the acceleration time delay. Over compensation during deceleration will not result in loss of lock, but will cause increased levels of torque ripple and motor current until the ideal commutation time is eventually reached.

모터에 대한 전압이 갑자기 떨어지거나 모터 부하가 갑자기 증가하면 제어 루프에서 어떤 일이 발생하는지 고려하십시오. 변화가 충분히 큰 경우, 정류 시간은 모터 조건에 대해 너무 짧게 작동 할 것입니다. 모터는 이를 견딜 수 없으며 잠금 장치가 손실된다. 잠금이 손실되지 않도록 루프 감속 타이머 지연은 제어 루프를 추적하기에 충분한만큼 짧아야 하며, 또는 변화하는 모터 상태에 선행해야 합니다. 시간 지연이 너무 짧으면 제어 루프는 모터 응답에 앞서 정류 시간을 길게하여 과도한 보상을 초래한다. 모터는 결국 BEMF 센서에 속도가 적용 전압에 비해 너무 느리다는 것을 나타내는 속도로 느려질 것입니다. 이 시점에서 정류장 감속은 중단되고 정류 변경은 가속 시간 지연에 의해 제어되는 반대 방향으로 조정됩니다. 감속 중의 과도한 보상은 잠금을 손상시키는 것은 아니지만 결국 이상적인 정류 시간에 도달 할 때까지 토크 리플과 모터 전류를 증가시킵니다

Determining The Commutation Time Table Values (정류 시간 테이블 값 결정)

The assembler supplied with MPLAB performs all calculations as 32-bit integers. To avoid the rounding errors that would be caused by integer math, we will use a spreadsheet, such as Excel, to compute the table entries then cut and paste the results to an include file. The spreadsheet is setup as shown in Table 4.

MPLAB과 함께 제공되는 어셈블러는 모든 계산을 32 비트 정수로 수행합니다.

정수 연산에 의해 발생하는 반올림 오류를 피하기 위해 Excel과 같은 스프레드 시트를 사용하여 테이블 항목을 계산하고 그 결과를 잘라 포함 파일에 붙여 넣습니다. 스프레드 시트는 표 4와 같이 설정됩니다.

TABLE 4: COMMUTATION TIME TABLE VALUES

Variable Name	Number or Formula	Description
Phases	12	Number of commutation phase changes in one mechanical revolution.
Fosc	20 MHz	Microcontroller clock frequency
Fosc_4	Fosc/4	Microcontroller timers source clock
Prescale	4	Timer 1 prescale
MaxRPM	8000	Maximum expected speed of the motor at full applied voltage
MinRPM	$(60 * \text{Fosc_4}) / (\text{Phases} * \text{Prescale} * 65535) + 1$	Limitation of 16-bit timer
Offset	-345	This is the zero voltage intercept on the RPM axis. A property normalized to the 8-bit A to D converter.
Slope	$(\text{MaxRPM} - \text{Offset}) / 255$	Slope of the RPM to voltage input response curve normalized to the 8-bit A to D converter.

The body of the spreadsheet starts arbitrarily at row 13. Row 12 contains the column headings.

The body of the spreadsheet is constructed as follows:

- Column A is the commutation table index number N.

The numbers in column A are integers from 0 to 255.

- Column B is the RPM that will result by using the counter values at index number N. The formula in column B is: $\text{=IF}(\text{Offset} + \text{A13} * \text{Slope} > \text{MinRPM}, \text{Offset} + \text{A13} * \text{Slope}, \text{MinRPM})$.

- Column C is the duration of each commutation phase expressed in seconds.

The formula for column C is: $\text{=60} / (\text{Phases} * \text{B13})$.

- Column D is the duration of each commutation phase expressed in timer counts.

The formula for column D is: $\text{=C13} * \text{FOSC_4} / \text{Prescale}$.

스프레드 시트의 본문은 행 13에서 임의로 시작됩니다. 12 행에는 열 머리글이 있습니다.

스프레드 시트의 본문은 다음과 같이 구성됩니다.

- 열 A는 정류 테이블 색인 번호 N입니다.

A 열의 숫자는 0에서 255 사이의 정수입니다.

- B 열은 인덱스 번호 N의 카운터 값을 사용하는 RPM입니다. B 열의 수식은 다음과 같습니다.

$\text{= IF (오프셋 + A13 * 슬로프} > \text{MinRPM, 오프셋 + A13 * 슬로프, MinRPM)}$.

- C 열은 각 정류 단계의 지속 시간을 초 단위로 나타냅니다.

C 열에 대한 수식은 $\text{= 60} / (\text{Phases} * \text{B13})$ 입니다.

- 열 D는 타이머 수로 표현 된 각 정류 단계의 지속 시간입니다.

열 D의 수식은 다음과 같습니다. $\text{= C13} * \text{FOSC_4} / \text{Prescale}$.

The range of commutation phase times at a reasonable resolution requires a 16-bit timer. The timer counts from 0 to a compare value then automatically resets to 0. The compare values are stored in the commutation time table. Since the comparison is 16 bits and tables can only handle 8 bits the commutation times will be stored in two tables accessed by the same index.

- Column E is the most significant byte of the 16-bit timer compare value.

The formula for column E is:

=CONCATENATE("retlw high D",INT(D13),",").

- Column F is the least significant byte of the 16-bit timer compare value. The formula for column F is:

=CONCATENATE("retlw low D",INT(D13),",").

적절한 해상도의 정류 위상 시간의 범위는 16비트 타이머가 필요합니다. 타이머는 0에서 비교 값까지 계산하고 자동으로 0으로 초기화됩니다. 비교 값은 정류 시간 테이블에 저장됩니다. 비교가 16 비트이고 테이블이 8 비트 만 처리 할 수 있기 때문에 정류 시간은 동일한 인덱스가 액세스하는 두 테이블에 저장됩니다..

- E 열은 16 비트 타이머 비교 값의 최상위 바이트입니다. 열 E의 수식은 다음과 같습니다.

= CONCATENATE ("retlw high D ' , INT (D13), " ").

- 열 F는 16 비트의 최하위 바이트입니다

타이머 비교 값. 열 F의 수식은 다음과 같습니다.

= CONCATENATE ("retlw low D ' , INT (D13), " ").

When all spreadsheet formulas have been entered in row 13, the formulas can be dragged down to row 268 to expand the table to the required 256 entries. Columns E and F will have the table entries in assembler ready format. An example of the table spreadsheet is shown in Figure 13.

모든 스프레드 시트 수식이 13 행에 입력되면 수식을 268 행으로 끌어 테이블을 필요한 256 항목으로 확장 할 수 있습니다. 열 E와 F는 어셈블러 준비 형식의 테이블 항목을 가집니다. 표 스프레드 시트의 예가 그림 13에 나와 있습니다.

FIGURE 13: PWM LOOKUP TABLE GENERATOR

The screenshot shows a spreadsheet titled "BLDC Table Generator.xls". It has columns A through F. Rows 1-8 contain input parameters: Phases/Rev (12), Fosc (2.00E+07), Fosc/4 (5.00E+06), Prescale (4), MaxRPM (8000), MinRPM (96), Offset (-345.00), and Slope (32.73). Rows 12-28 show a generated table with columns N, RPM, Sec per Transition, Timer Counts, MS Byte Code, and LS Byte Code. The table lists values for N from 0 to 15, with RPM increasing from 96 to 146. The MS Byte Code and LS Byte Code are shown in hexadecimal format.

	A	B	C	D	E	F
1	Phases/Rev	12				
2	Fosc	2.00E+07				
3	Fosc/4	5.00E+06				
4	Prescale	4				
5	MaxRPM	8000				
6	MinRPM	96				
7	Offset	-345.00				
8	Slope	32.73				
9						
10						
11						
12	N	RPM	Sec per Transition	Timer Counts	MS Byte Code	LS Byte Code
13	0	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
14	1	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
15	2	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
16	3	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
17	4	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
18	5	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
19	6	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
20	7	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
21	8	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
22	9	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
23	10	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
24	11	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
25	12	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
26	13	96	5.19E-02	64855	retlw high D'64854'	retlw low D'64854'
27	14	113	4.42E-02	55233	retlw high D'55233'	retlw low D'55233'
28	15	146	3.43E-02	42843	retlw high D'42842'	retlw low D'42842'

Using Open Loop Control to Determine Motor Characteristics

(개방 루프 제어를 사용하여 모터 특성 결정)

You can measure the motor characteristics by operating the motor in Open Loop mode, and measuring the motor current at several applied voltages. You can then chart the response curve in a spreadsheet, such as Excel, to determine the slope and offset numbers. Finally, plug the maximum RPM and offset numbers back into the table generator spreadsheet to regenerate the RPM tables.

오픈 루프 모드에서 모터를 조작하고 일부인가 전압에서 모터 전류를 측정함으로써 모터의 특성을 측정 할 수 있습니다 그런 다음 Excel과 같은 스프레드 시트에서 응답 곡선을 차트로 작성하여 기울기 및 오프셋 수를 결정할 수 있습니다. 마지막으로 최대 RPM 및 오프셋 수를 테이블 생성기 스프레드 시트에 다시 연결하여 RPM 테이블을 재생성합니다.

To operate the motor in Open Loop mode:

- Set the manual threshold number (ManThresh) to 0xFF.
This will prevent the Auto mode from taking over.
- When operating the motor in Open Loop mode,
start by adjusting the offset control until the motor starts to move.
You may also need to adjust the PWM control slightly above minimum.
- After the motor starts, you can increase the PWM control to increase the motor speed.
The RPM and voltage will track, but you will need to adjust the offset frequently to optimize the voltage for the selected RPM.
- Optimize the voltage by adjusting the offset for minimum current.

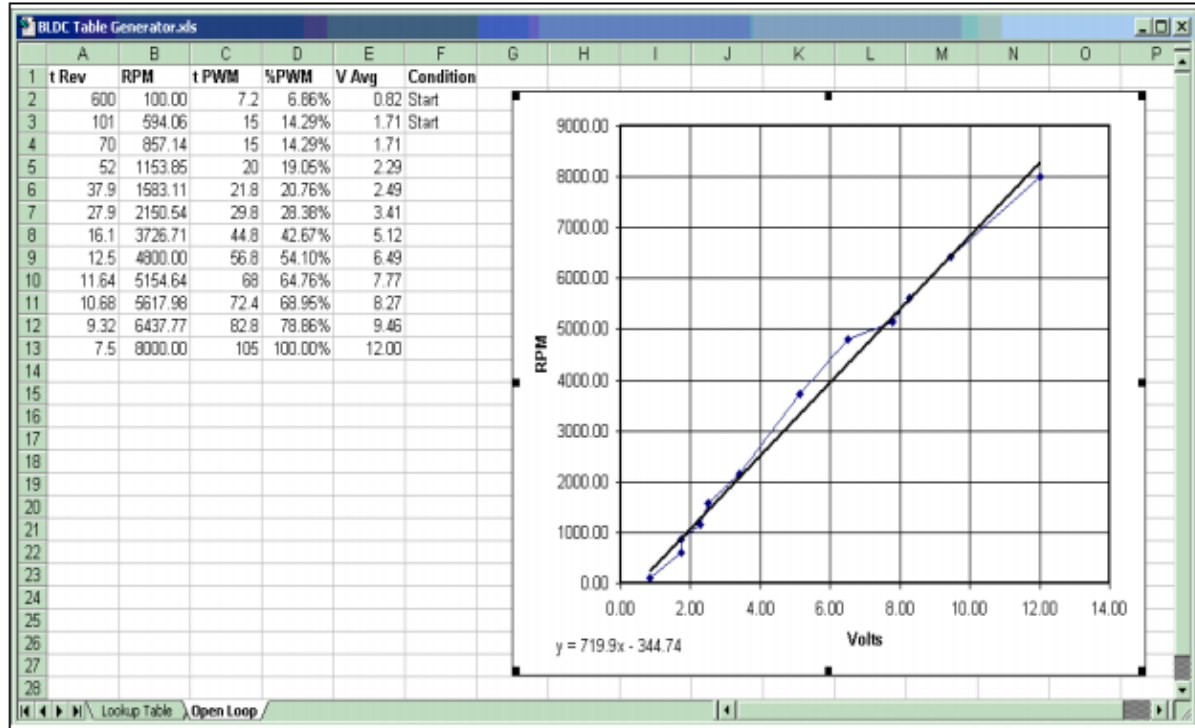
- 수동 임계 값 (ManThresh)을 0xFF로 설정합니다.
이렇게하면 자동 모드가 인계받는 것을 막을 수 있습니다.
- 개방 루프 모드에서 모터를 작동 할 때,
모터가 움직이기 시작할 때까지 오프셋 제어를 조정하여 시작하십시오. PWM 제어를
최소보다 약간 높게 조정해야 할 수도 있습니다.
- 모터가 시작된 후 PWM 제어를 증가시켜 모터 속도를 증가시킬 수 있습니다.
RPM 및 전압은 추적되지만, 선택한 RPM에 대한 전압을 최적화하려면 오프셋을 자주
조정해야 합니다.
- 최소 전류에 대한 오프셋을 조정하여 전압을 최적화하십시오.

To obtain the response offset with Excel®, enter the voltage (left column), and RPM (right column) pairs in adjacent columns of the spreadsheet. Use the chart wizard to make an X-Y scatter chart. When the chart is finished, right click on the response curve and select the pop-up menu “add trendline. . .” option.

Choose the linear regression type and, in the Options tab, check the “display equation on chart” option. An example of the spreadsheet is shown in Figure 14.

Excel®을 사용하여 응답 오프셋을 얻으려면 스프레드 시트의 인접 열에 전압 (왼쪽 열) 및 RPM (오른쪽 열) 쌍을 입력하십시오. chart wizard를 사용하여 X-Y 분산 형 차트를 만듭니다. 차트가 끝나면 응답 곡선을 마우스 오른쪽 버튼으로 클릭하고 팝업 메뉴 "추세선 추가" 옵션을 선택하십시오. "선형 회귀 유형을 선택하고 옵션 탭에서 "차트의 공식 표시" 옵션을 선택하십시오. 스프레드 시트의 예가 그림 14에 나와 있습니다.

FIGURE 14: MOTOR RESPONSE SCOPE DETERMINATION



Constructing The Sensorless Control Code (센서리스 제어 코드 구성)

At this point we have all the pieces required to control a sensorless motor. We can measure BEMF and the applied voltage then compare them to each other to determine rotor position. We can vary the effective applied voltage with PWM and control the speed of the motor by timing the commutation phases. Some measurement events must be precisely timed. Other measurement events need not to interfere with each other. The ADC must be switched from one source to another and allow for sufficient acquisition time. Some events must happen rapidly with minimum latency. These include PWM and commutation.

이 시점에서 센서리스 모터를 제어하는데 필요한 모든 부품을 갖추고 있습니다. BEMF와 인가 전압을 측정 한 후이를 서로 비교하여 로터의 위치를 결정할 수 있습니다.우리는 PWM을 사용하여 유효 적용 전압을 변화시키고 정류 위상을 타이밍 화하여 모터의 속도를 제어 할 수 있습니다. 일부 측정 이벤트는 정확하게 시간을 측정해야합니다. 다른 측정 이벤트는 서로 간섭하지 않아도됩니다. ADC는 한 소스에서 다른 소스로 전환되어야하며 충분한 검색 시간을 허용해야합니다. 이벤트 기간은 최소 지연 시간으로 신속하게 발생하는 것이 있습니다. 여기에는 PWM 및 정류가 포함되어 있습니다.

We can accomplish everything with a short main loop that calls a state table. The main loop will handle PWM and commutation and the state table will schedule reading the two potentiometers, the peak applied voltage and the BEMF voltages at two times when the attached motor terminal is floating. Figure A-1 through Figure A-10, in Appendix A, is the resulting flow chart of sensorless motor control. Code listings are in Appendix C and Appendix D.

상태 테이블을 호출 짧은 메인 루프에서 모두를 달성 할 수 있습니다. 메인 루프는 PWM 및 정류를 처리하고 상태 테이블은 설치된 모터 단자가 부동하고있을 때, 2 개의 전위차계, 최대인가 전압 및 BEMF 전압을 두 번 읽을 수 있도록 예약합니다. 부록 A의 그림 A-1 ~ 그림 A-10은 센서리스 모터 제어 결과의 흐름도입니다. 코드 목록은 부록 C와 부록 D에 있습니다.

APPENDIX A: SENSORLESS CONTROL FLOWCHART.

FIGURE A-1: MAIN LOOP

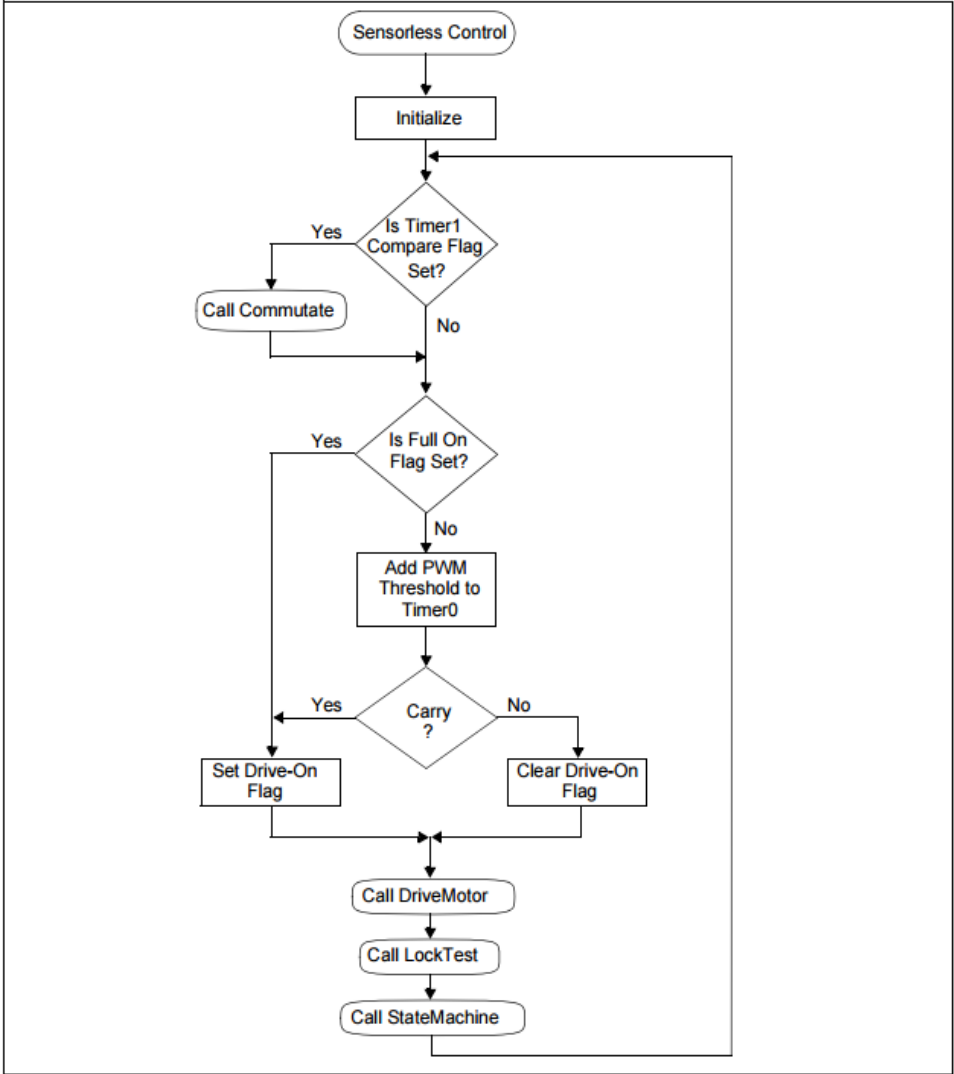


FIGURE A-2: MOTOR COMMUTATION

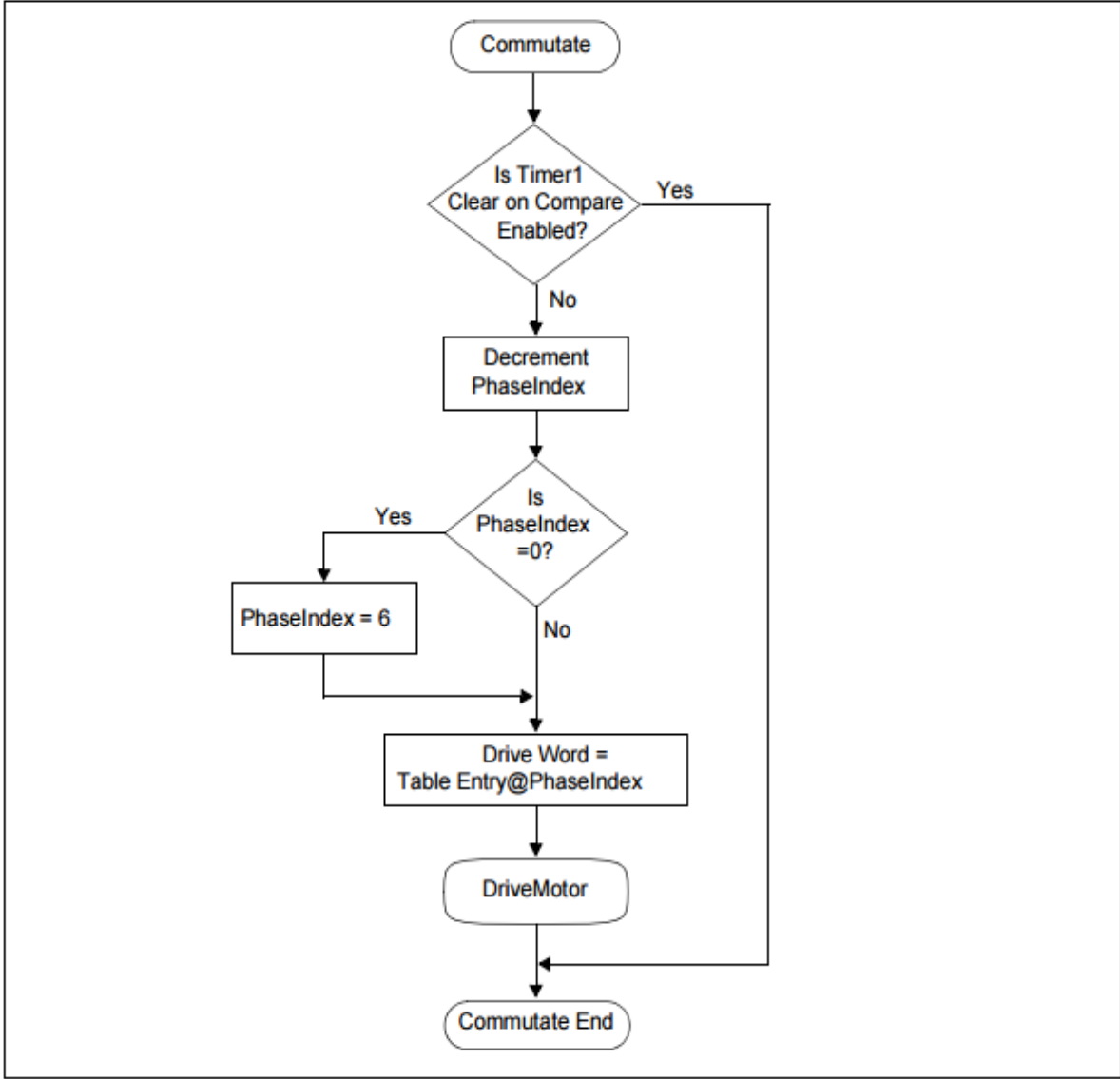


FIGURE A-3: MOTOR DRIVER CONTROL

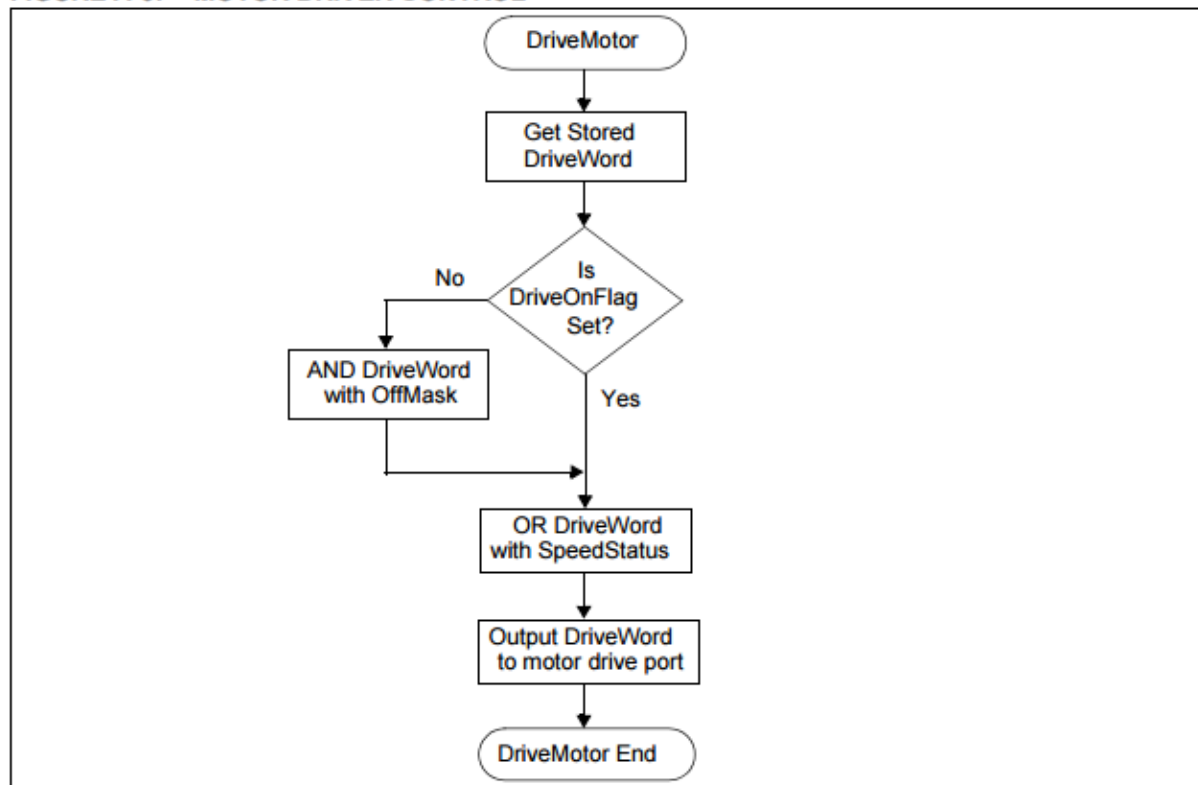


FIGURE A-4: PHASE DRIVE PERIOD

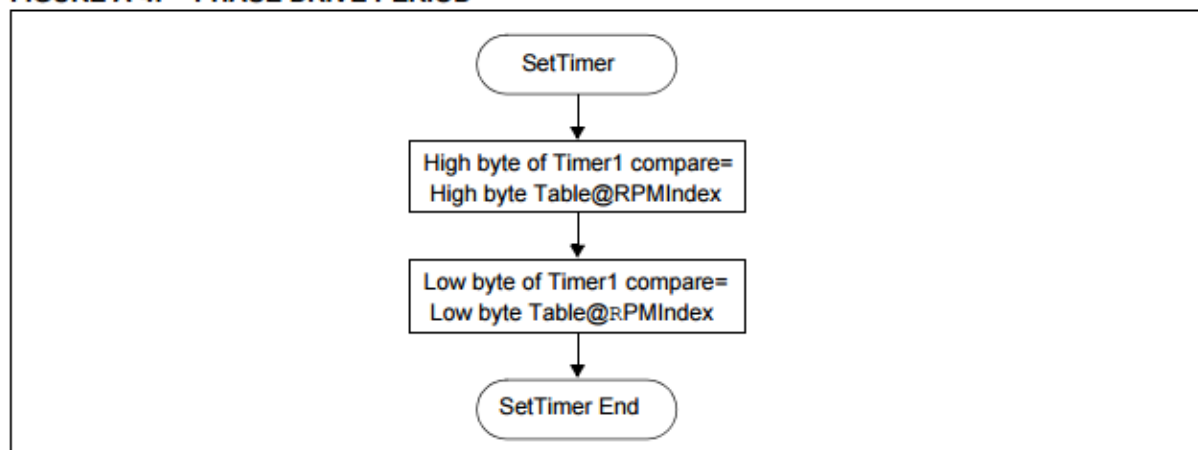


FIGURE A-5: MOTOR SPEED LOCKED WITH COMMUTATION RATE

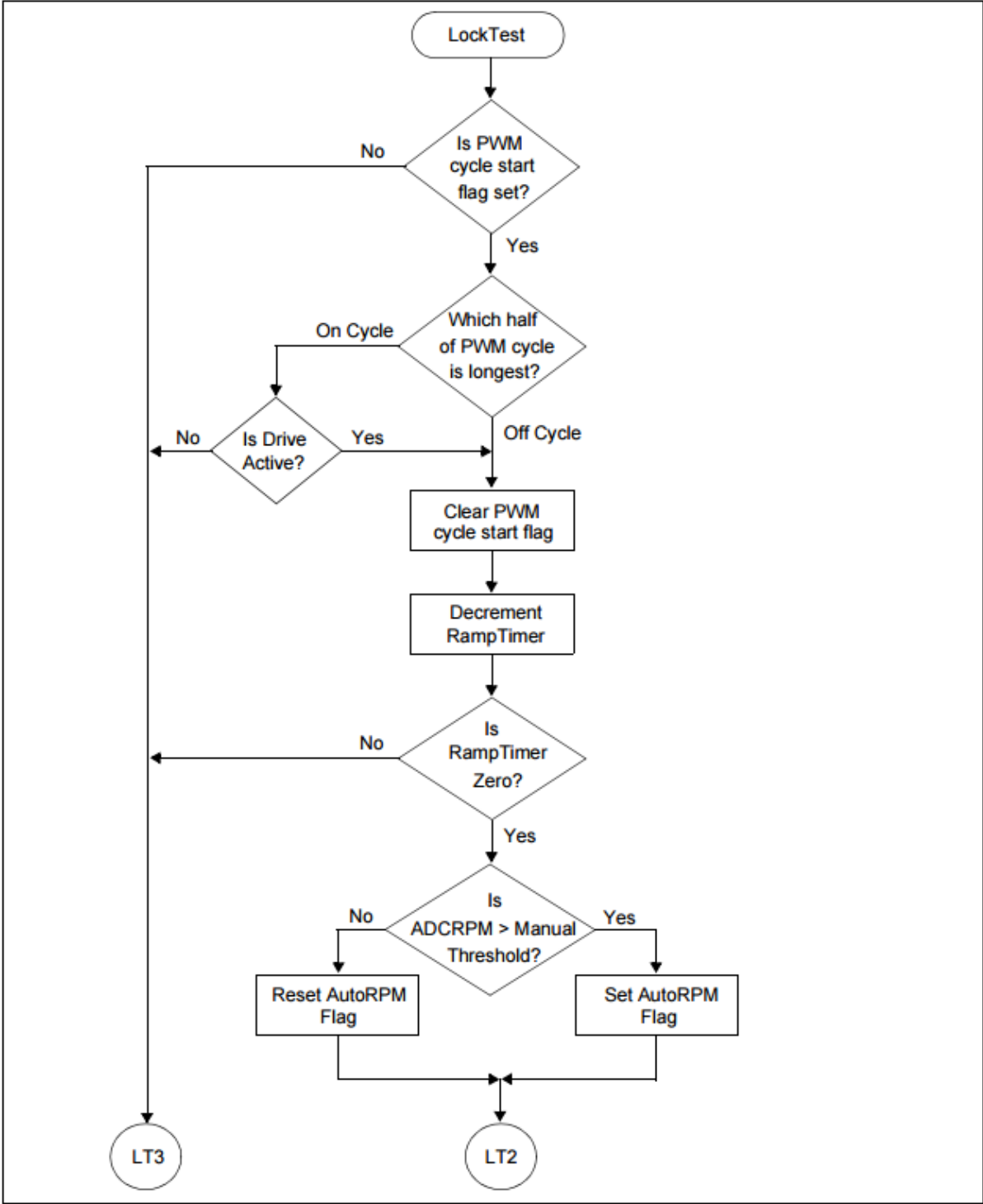


FIGURE A-6: MOTOR SPEED LOCKED WITH COMMUTATION RATE (CONT.)

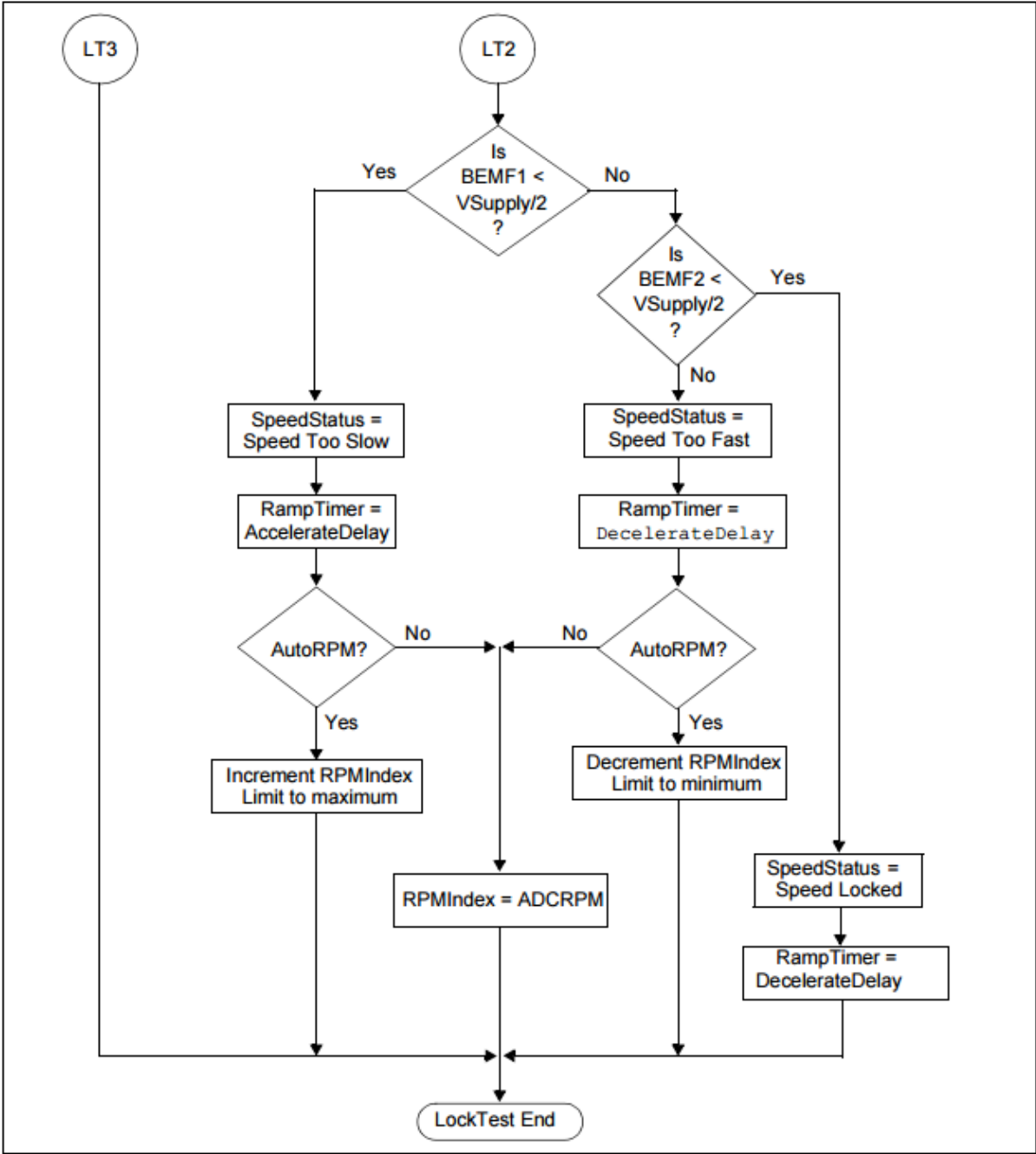


FIGURE A-7: MOTOR CONTROL STATE MACHINE

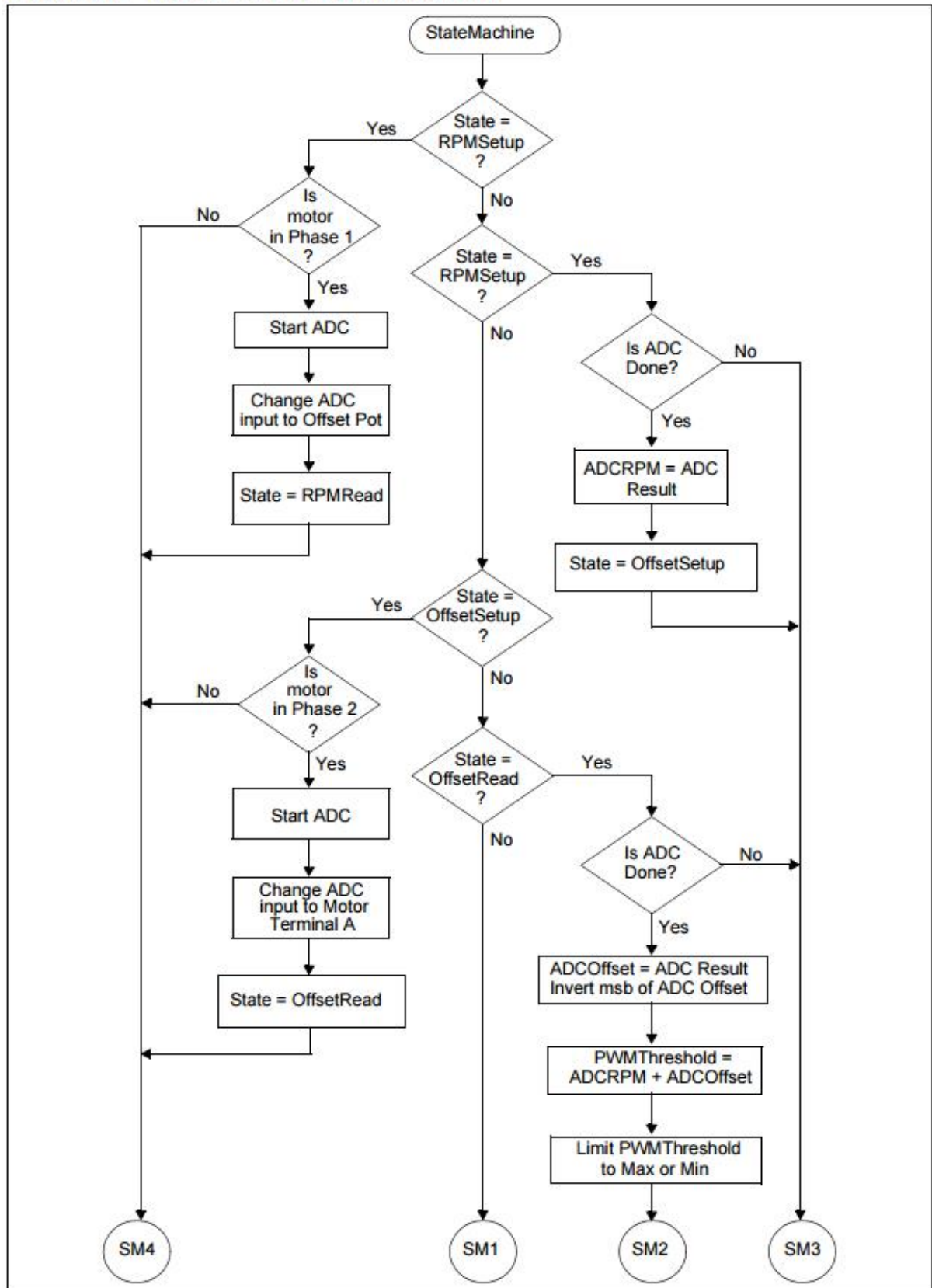


FIGURE A-8: MOTOR CONTROL STATE MACHINE (CONT.)

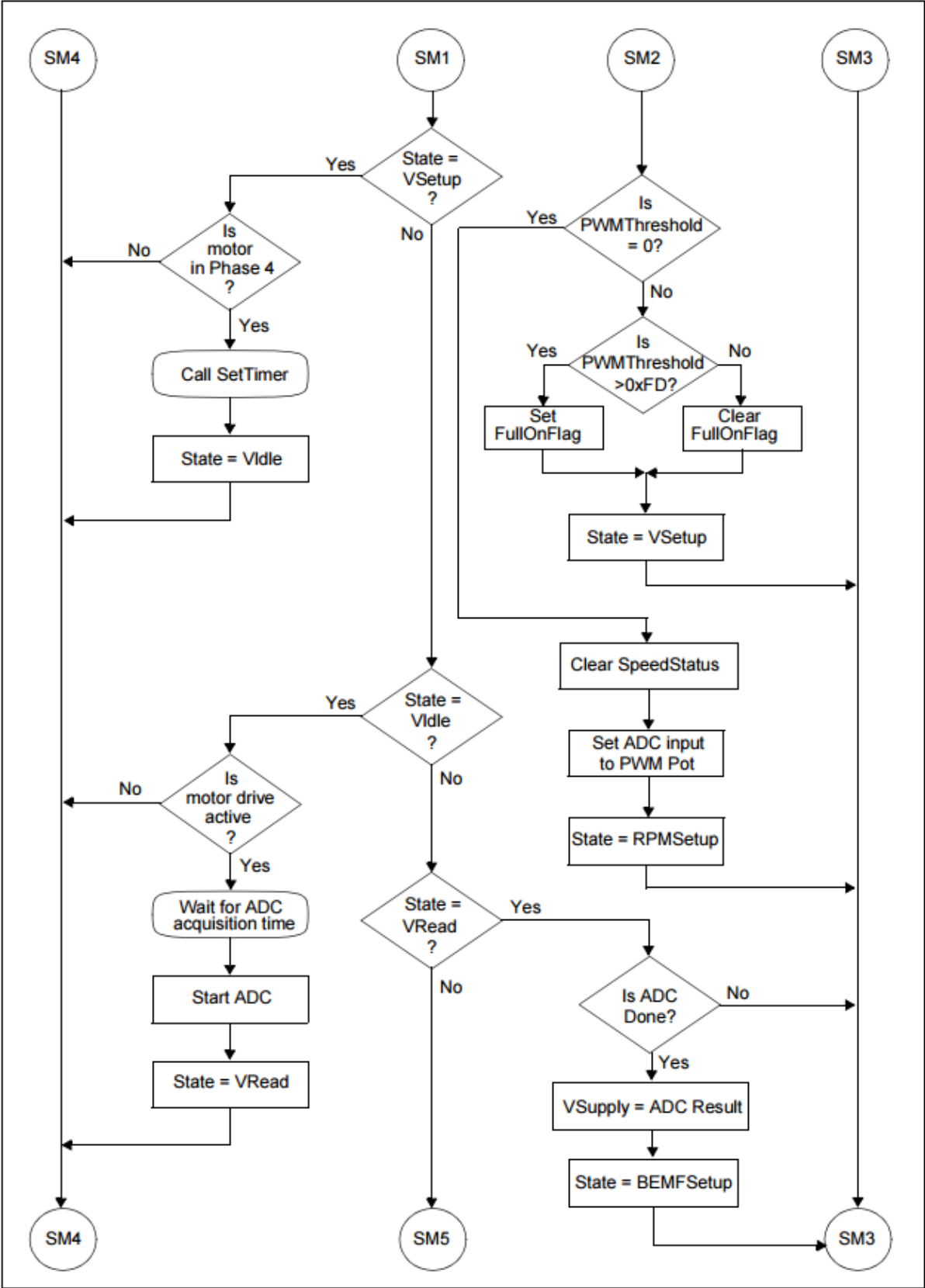


FIGURE A-9: MOTOR CONTROL STATE MACHINE (CONT.)

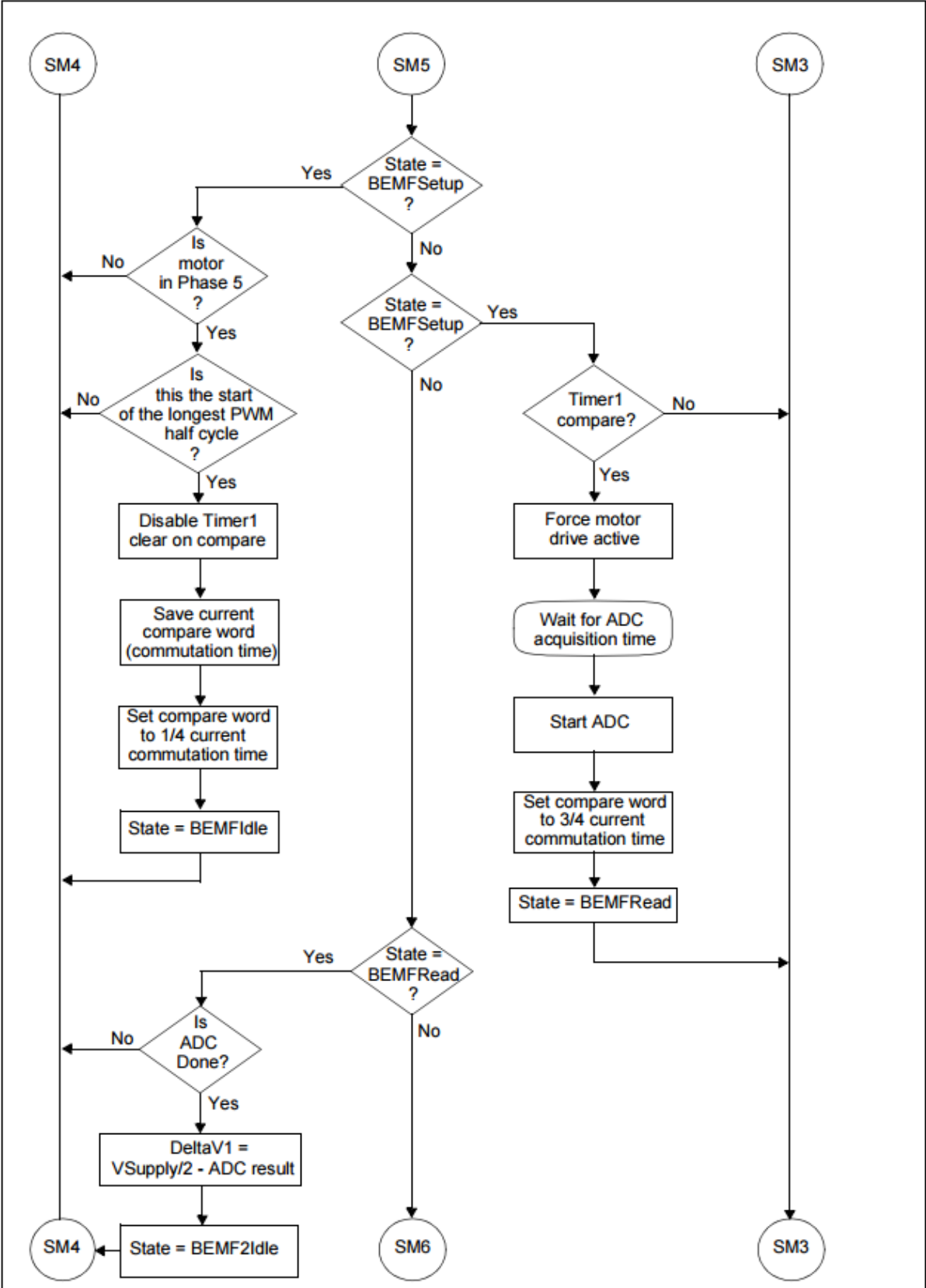


FIGURE A-10: MOTOR CONTROL STATE MACHINE (CONT.)

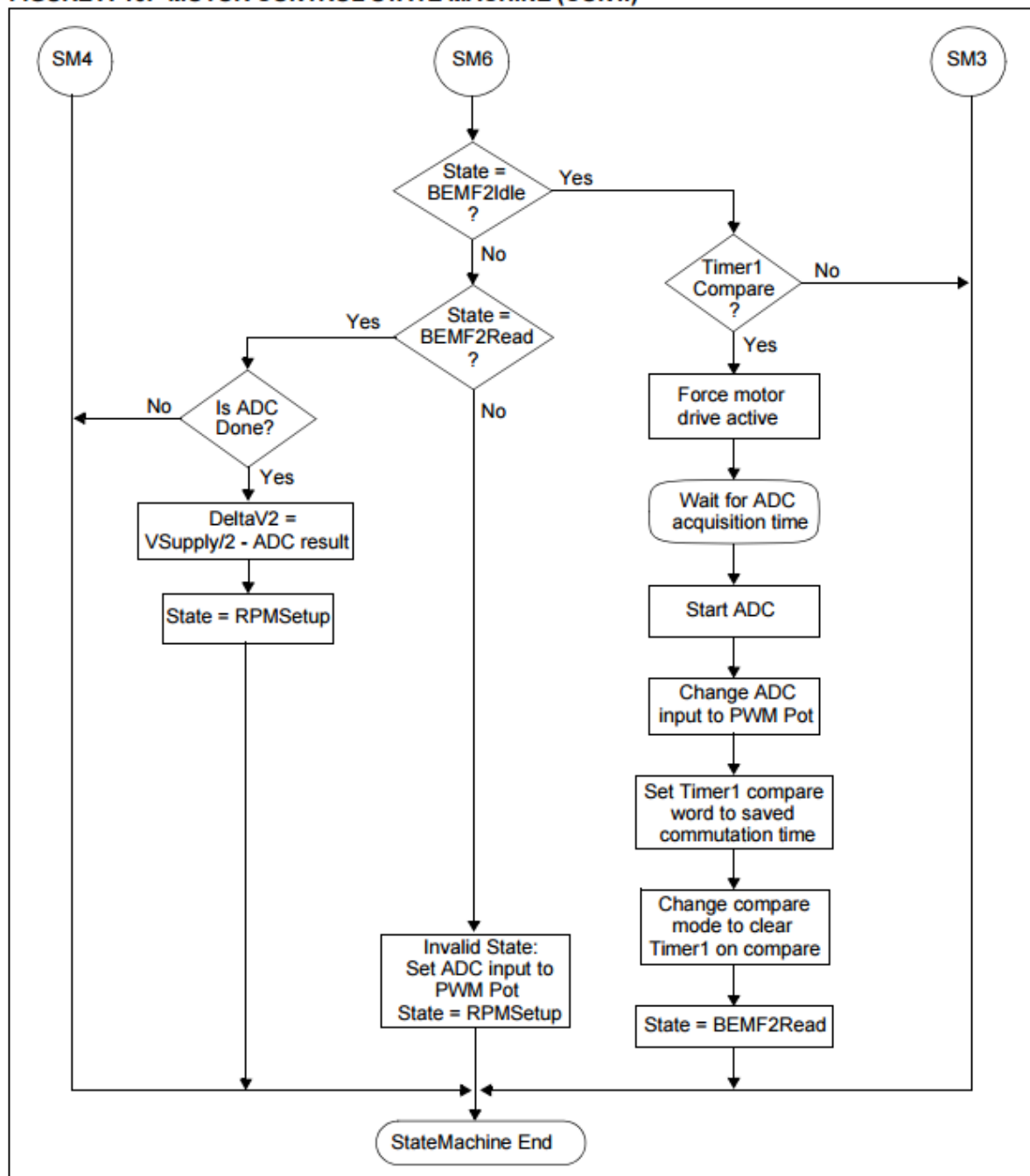
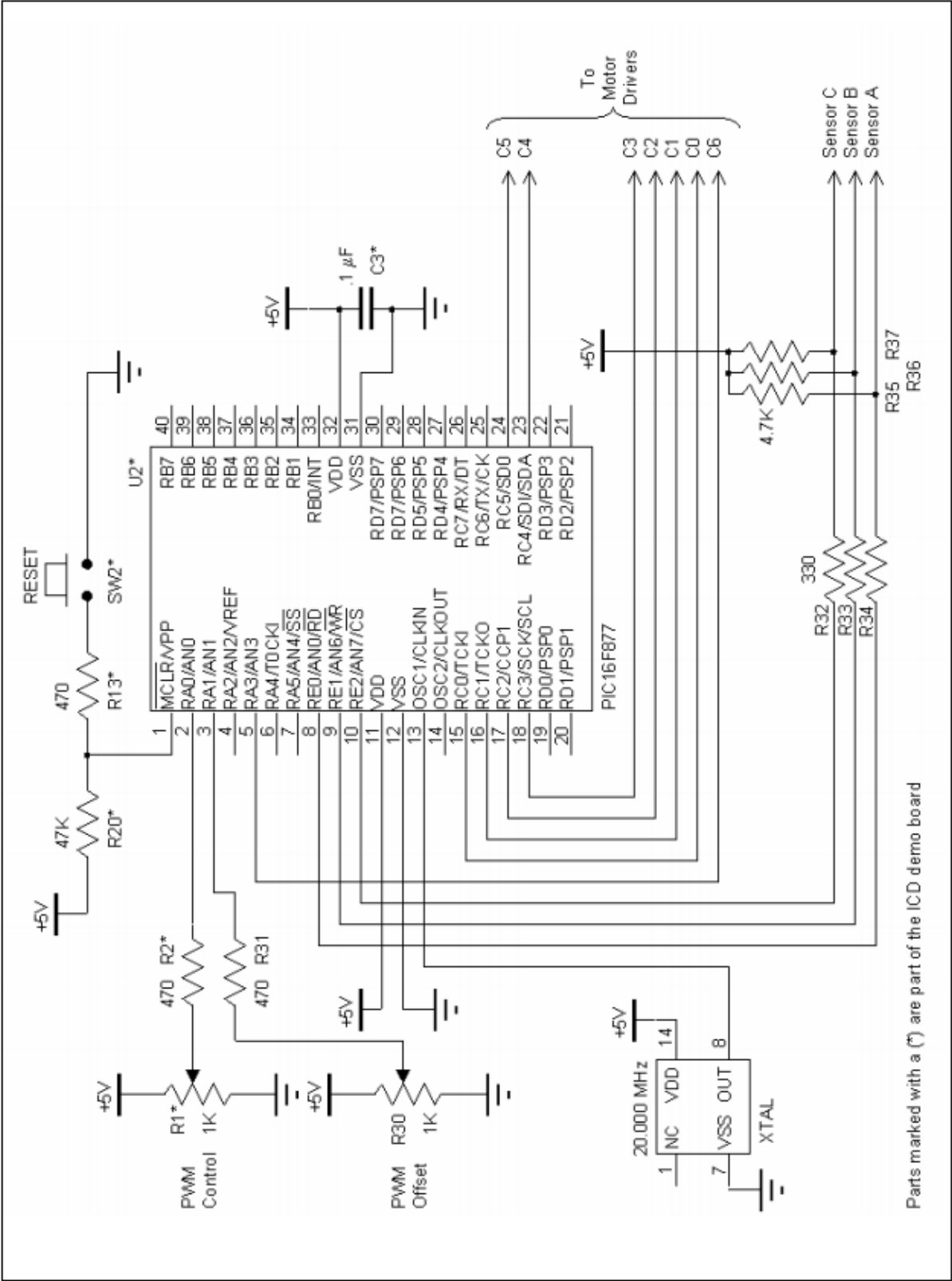


FIGURE B-1: SCHEMATIC A - MOTOR DRIVERS

[illegible]

(3) Required only for sensorless control.

FIGURE B-2: SCHEMATIC B - CONTROLLER



Software License Agreement

The software supplied herewith by Microchip Technology Incorporated (the "Company") for its PICmicro® Microcontroller is intended and supplied to you, the Company's customer, for use solely and exclusively on Microchip PICmicro Microcontroller products.

The software is owned by the Company and/or its supplier, and is protected under applicable copyright laws. All rights are reserved. Any use in violation of the foregoing restrictions may subject the user to criminal sanctions under applicable laws, as well as to civil liability for the breach of the terms and conditions of this license.

THIS SOFTWARE IS PROVIDED IN AN "AS IS" CONDITION. NO WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE APPLY TO THIS SOFTWARE. THE COMPANY SHALL NOT, IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, FOR ANY REASON WHATSOEVER.

<http://ww1.microchip.com/downloads/en/appnotes/00857a.pdf>

논문 주소임으로 이 뒤에부터는 코드이다.

APPENDIX D: SENSORLESS CODE