

MCP2515 datasheet 해석

특징

- 1 Mb / s에서 CAN V2.0B를 구현합니다.
- 0 - 8 바이트 길이의 데이터 필드
- 표준 및 확장 데이터 및 원격 프레임
- 수신 버퍼, 마스크 및 필터 :
- 우선 순위가 지정된 메시지 저장소가있는 두 개의 수신 버퍼
- 6 개의 29 비트 필터
- 두 개의 29 비트 마스크

- 처음 두 데이터 바이트에 대한 데이터 바이트 필터링
(표준 데이터 프레임에 적용)

우선 순위 지정 및 중단 기능이있는 3개의 전송 버퍼

- 고속 SPI 인터페이스 (10 MHz) :
- SPI 모드 0,0 및 1,1
- 원샷 모드는 메시지 전송이 한번만 시도 되도록합니다.
- 프로그래밍 가능한 프리스케일러가있는 클럭 출력 핀 :
- 다른 장치의 클럭 소스로 사용할 수 있습니다.

- SOF (Start-of-Frame) 신호를 사용할 수 있습니다.

SOF 신호 모니터링 :

- 타임 슬롯 기반 프로토콜 및 / 또는 버스 진단에 사용하여 조기 버스 퇴화를 감지 할 수 있습니다.

- 선택 가능한 인 에이블이 있는 인터럽트 출력 핀
- 버퍼 전체 출력 핀은 다음과 같이 구성 할 수 있습니다.
- 각 수신 버퍼에 대한 인터럽트 출력

- 범용 출력

• Request-to-Send (RTS) 입력 핀을 개별적으로

다음과 같이 구성 가능 :

- 각각에 대한 전송을 요청하는 제어 핀

송신 버퍼

- 범용 입력

• 저전력 CMOS 기술 :

- 2.7V ~ 5.5V에서 동작

- 5 mA 활성 전류 (일반)

- 대기 전류 $1\mu A$ (일반) (절전 모드)

• 지원 온도 범위 :

- 산업용 (I) : $-40^{\circ} C \sim +85^{\circ} C$

- 확장 (E) : $-40^{\circ} C \sim +125^{\circ} C$

기술

마이크로 칩 테크놀로지의 MCP2515는 독립형

컨트롤러 영역 네트워크 (CAN) 컨트롤러.

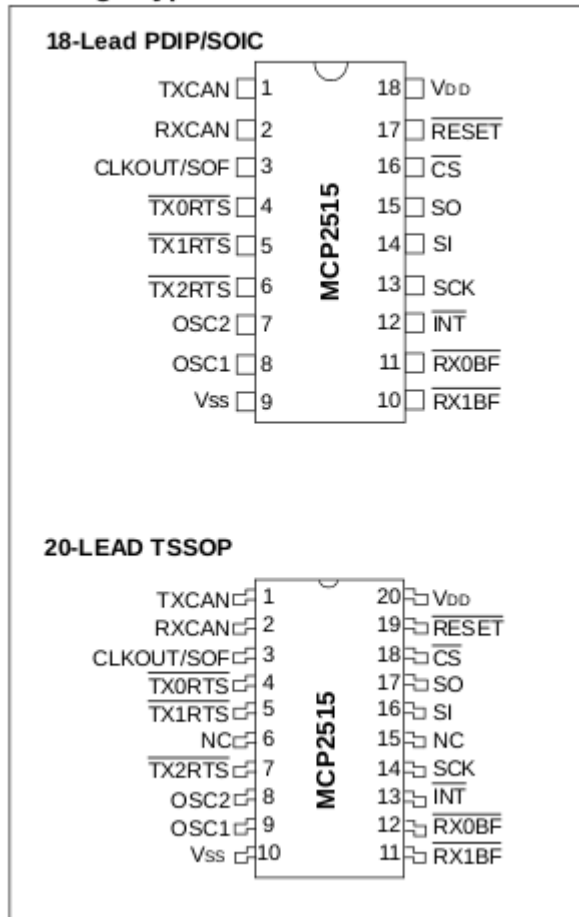
CAN 사양, 버전 2.0B.

표준 및 확장 데이터 및 원격 프레임을 송수신 할 수 있습니다.

MCP2515에는 불필요한 메시지를 걸러내는 데 사용되는 2 개의 수용 마스크와 6 개의 수용 필터가있어 호스트 MCU 오버 헤드를 줄입니다.

MCP2515는 산업 표준 SPI (Serial Peripheral Interface)를 통해 마이크로 컨트롤러 (MCU)와 인터페이스합니다.

Package Types



1.0 장치 개요

MCP2515는 독립형 CAN 컨트롤러입니다.

CAN 버스와 인터페이스가 필요한 어플리케이션을 단순화하기 위해 개발되었습니다.

그림 1-1에는 MCP2515의 간단한 블록 다이어그램이 나와 있습니다.

이 장치는 세 가지 주요 블록으로 구성됩니다.

1. CAN 모듈은 CAN프로토콜 엔진, 마스크, 필터, 전송 및 수신 버퍼를 포함하고있다.
2. 장치와 장치의 작동을 구성하는 데 사용 되는 제어 논리 및 레지스터가 있다.
- 3.The SPI 프로토콜 블록이다.

이 장치를 사용하는 시스템 구현 예가 그림 1-2에 나와 있습니다

1.1 캔 모듈

CAN 모듈은 CAN 버스에서 메시지를 수신하고 전송하기 위한 모든 기능을 처리합니다.

메시지는 먼저 적절한 메시지 버퍼 및 제어 레지스터를 로드하여 전송됩니다.

전송은 SPI 인터페이스를 통한 제어 레지스터 비트를 사용하거나 전송 인 에이블 핀을 사용하여 시작된다.

상태 및 오류는 해당 레지스터를 읽음으로써 확인할 수 있습니다.

CAN 버스에서 감지된 모든 메시지는 오류를 검사한 다음 사용자 정의 필터와 대조하여 두 개의 수신 버퍼 중 하나로 이동해야 하는지 확인합니다.

1.2 제어 로직

제어 논리 블록은 정보를 전달하고 제어하기 위해 다른 블록과 인터페이싱하여 MCP2515의 설정 및 작동을 제어합니다.

인터럽트 핀은 시스템 유연성을 높이기 위해 제공됩니다.

유효한 메시지가 수신 버퍼 중 하나로 로드되었음을 나타내는 데 사용할 수 있는 각 수신 레지스터에 대해 하나의 다목적 인터럽트 핀 (특정 인터럽트 핀은 물론)이 있습니다.

특정 인터럽트 핀의 사용은 선택 사항입니다.

일반적인 목적의 인터럽트 핀과 status 레지스터 (SPI 인터페이스를 통해 액세스)는 유효한 메시지가 수신된 시기를 결정하는 데에도 사용할 수 있습니다.

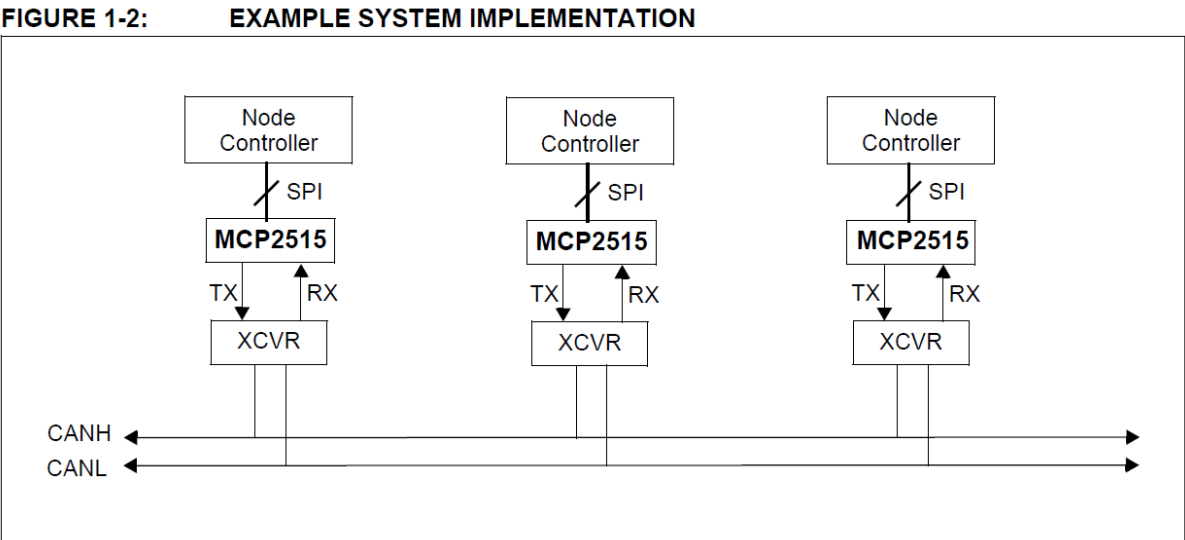
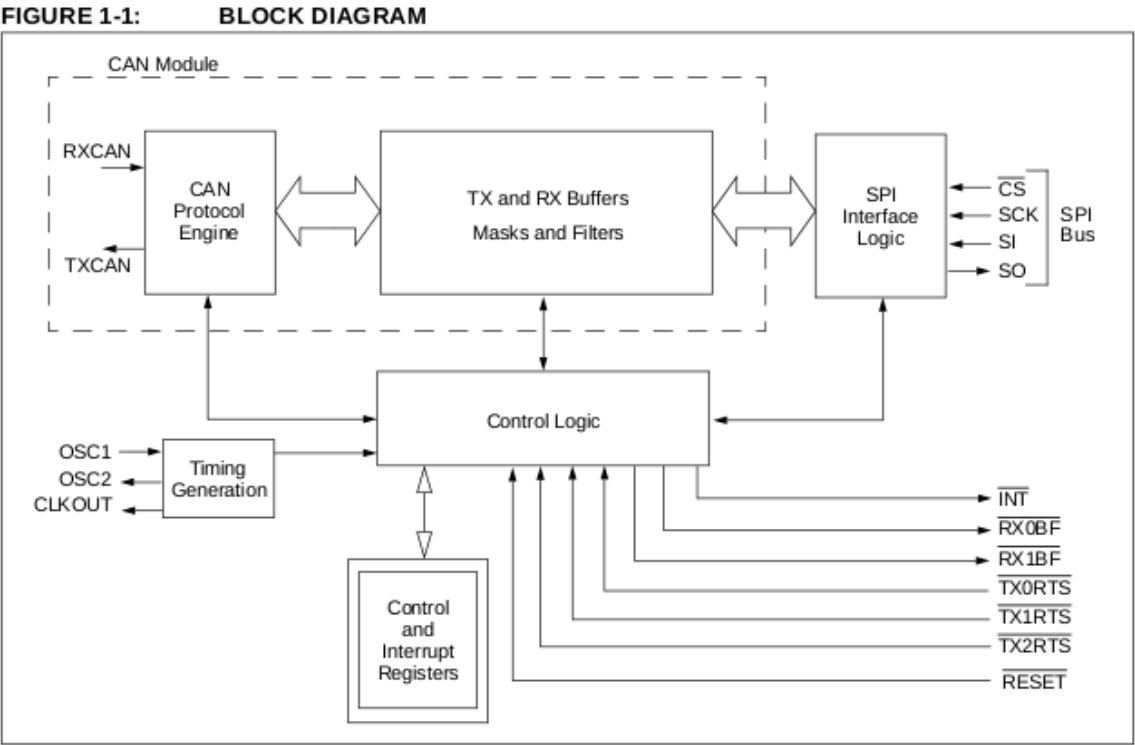
또한 3 개의 전송 레지스터 중 하나에 로드된 메시지의 즉시 전송을 시작하는 데 사용할 수 있는 3 개의 핀이 있습니다.

메신저 전송은 SPI 인터페이스를 통해 액세스되는 제어 레지스터를 사용하여 수행할 수 있기 때문에 이 핀의 사용은 선택 사항이다.

1.3 SPI 프로토콜 블록

MCU는 SPI 인터페이스를 통해 장치와 인터페이스합니다.

모든 레지스터에 쓰기 및 읽기는 특수 SPI 명령 외에도 표준 SPI 읽기 및 쓰기 명령을 사용하여 완성됩니다.



Name	PDIP/SOIC Pin#	TSSOP Pin	입력/출력 /파워 type	기술	대체 핀 기능
TXCAN	1	1	O	출력핀을	-

				CAN버스로 송신	
RXCAN	2	2	I	CAN버스로 부터 입력된 수신	-
CLKOUT	3	3	O	프로그래밍 프리스케일 러가 내장된 클록 출력핀	시작 프레임 신호
TX0RTS	4	4	I	송신 버퍼 TXB0 송신 요청. VDD에 100 kΩ 내부 풀업	범용 디지털 입력. VDD에 100 kΩ 내부 풀업
TX1RTS	5	5	I	송신 버퍼 TXB1 송신 요청. VDD에 100 kΩ 내부 풀업	범용 디지털 입력. VDD에 100 kΩ 내부 풀업
TX2RTS	6	7	I	송신 버퍼 TXB2 송신 요청. VDD에 100 kΩ 내부 풀업	범용 디지털 입력. VDD에 100 kΩ 내부 풀업
OSC2	7	8	O	오실레이터 출력	-
OSC1	8	9	I	오실레이터 입력	확장클록입 력
VSS	9	10	P	로직에대한	-

				GND참조 및 I/O핀	
RX1BF	10	11	O	수신 버퍼 RXB1 인터럽트 핀 또는 범용 디지털 출력	범용 디지털 출력
RX0BF	11	12	O	수신 버퍼 RXB0 인터럽트 핀 또는 범용 디지털 출력	범용 디지털 출력
INT	12	13	O	인터럽트 출력핀	-
SCK	13	14	I	SPI 인터페이스 를위한 클록 입력 핀	-
SI	14	16	I	SPI 인터페이스 용 데이터 입력 핀	-
SO	15	17	O	SPI 인터페이스 용 데이터 출력 핀	-
CS	16	18	I	SPI 인터페이스 용 칩 선택 입력 핀	-
RESET	17	19	I	액티브 로우 디바이스	-

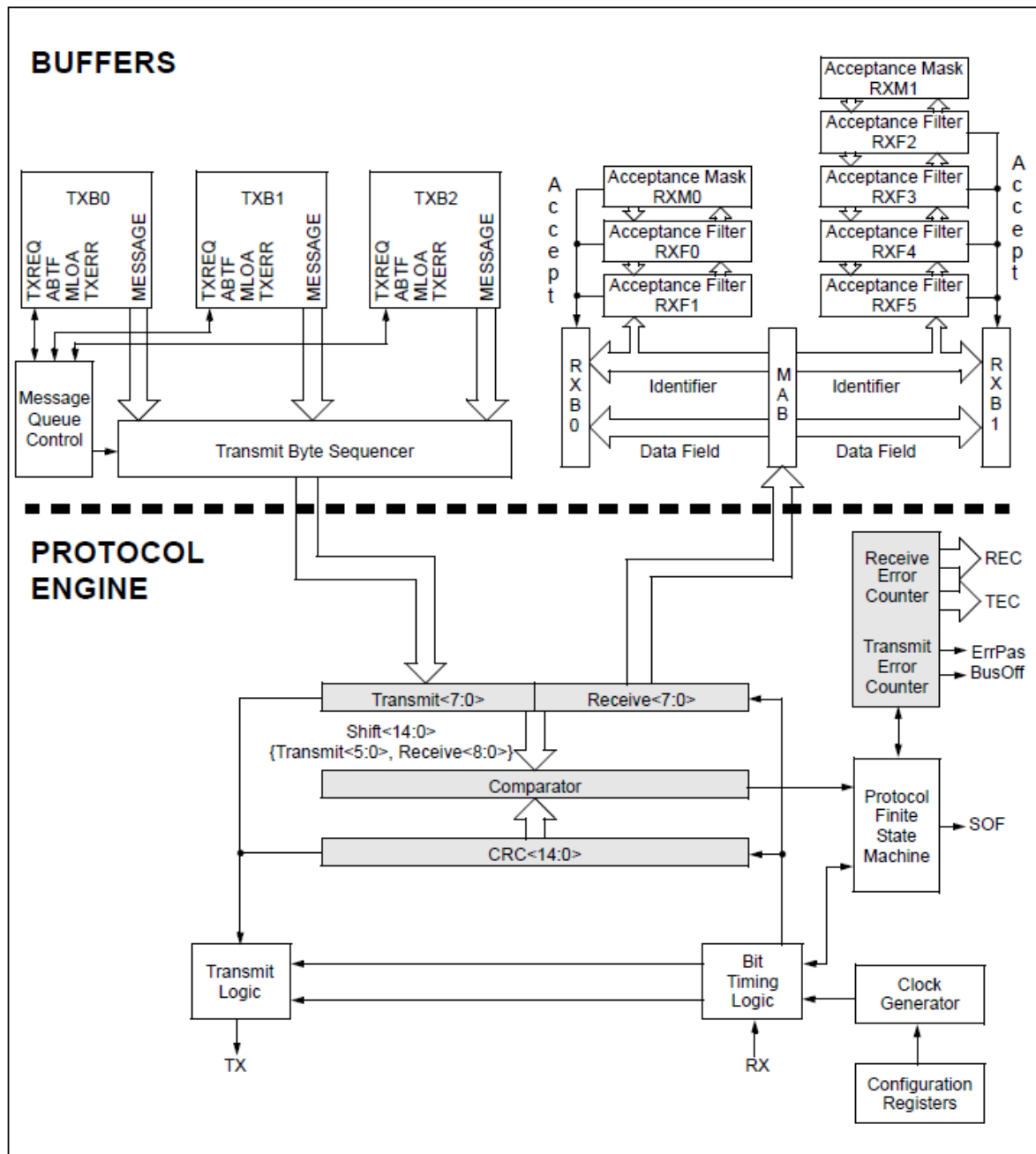
				리셋 입력	
VDD	18	20	P	로직 및 I/O 핀에 대한 양의 공급	-
NC	-	6,15	-	내부 연결 없음	

1.4 송신 / 수신 버퍼 / 마스크 / 필터

MCP2515는 3 개의 송신 버퍼와 2 개의 수신 버퍼, 2 개의 수신 마스크 (각 수신 버퍼 당 하나) 및 총 6 개의 수신 필터를 가지고 있습니다.

그림 1-3은 이러한 버퍼의 블록 다이어그램과 프로토콜 엔진에 대한 연결을 보여줍니다.

FIGURE 1-3: CAN BUFFERS AND PROTOCOL ENGINE BLOCK DIAGRAM



1.5 CAN 프로토콜 엔진

CAN 프로토콜 엔진은 그림 1-4에 나와 있으며 아래에 설명 된 여러 기능 블록을 결합합니다.

1.5.1 프로토콜 유한 상태 기계

엔진의 핵심은 Finite State Machine (FSM)입니다.

FSM은 TX / RX 시프트 레지스터, CRC 레지스터 및 버스 라인 사이. 순차적 데이터 스트림을 제어하는 시퀀서입니다.

또한 FSM은 오류 관리 논리 (EML) 및 TX / RX 시프트 레지스터와 버퍼 간의 병렬 데이터 스트림을 제어합니다.

FSM은 수신, 중재, 전송 및 에러 시그널링은 CAN 프로토콜에 따라 수행됩니다.

버스 라인의 메시지 자동 재전송은 FSM에서도 처리됩니다.

1.5.2 환급 리스크 점검

CRC (Cyclic Redundancy Check) 레지스터는 제어 필드 (데이터 바이트가 0 인 메시지의 경우) 또는 데이터 필드 다음에 전송되고 수신 메시지의 CRC 필드를 확인하는 데 사용되는 순환 중복 검사 (CRC) 코드를 생성합니다.

1.5.3 오류 관리 논리

오류 관리 논리 (EML)는 CAN 장치의 오류 감금을 담당합니다.

두 개의 카운터 인 REC (Receive Error Counter)와 TEC (Transmit Error Counter)는 비트 스트림 프로세서의 명령에 의해 증가 및 감소됩니다.

에러 카운터의 값에 따라 CAN 컨트롤러는 에러 - 활성, 에러 - 패시브 또는 버스 - 오프 상태로 설정됩니다.

1.5.4 비트 타이밍 로직

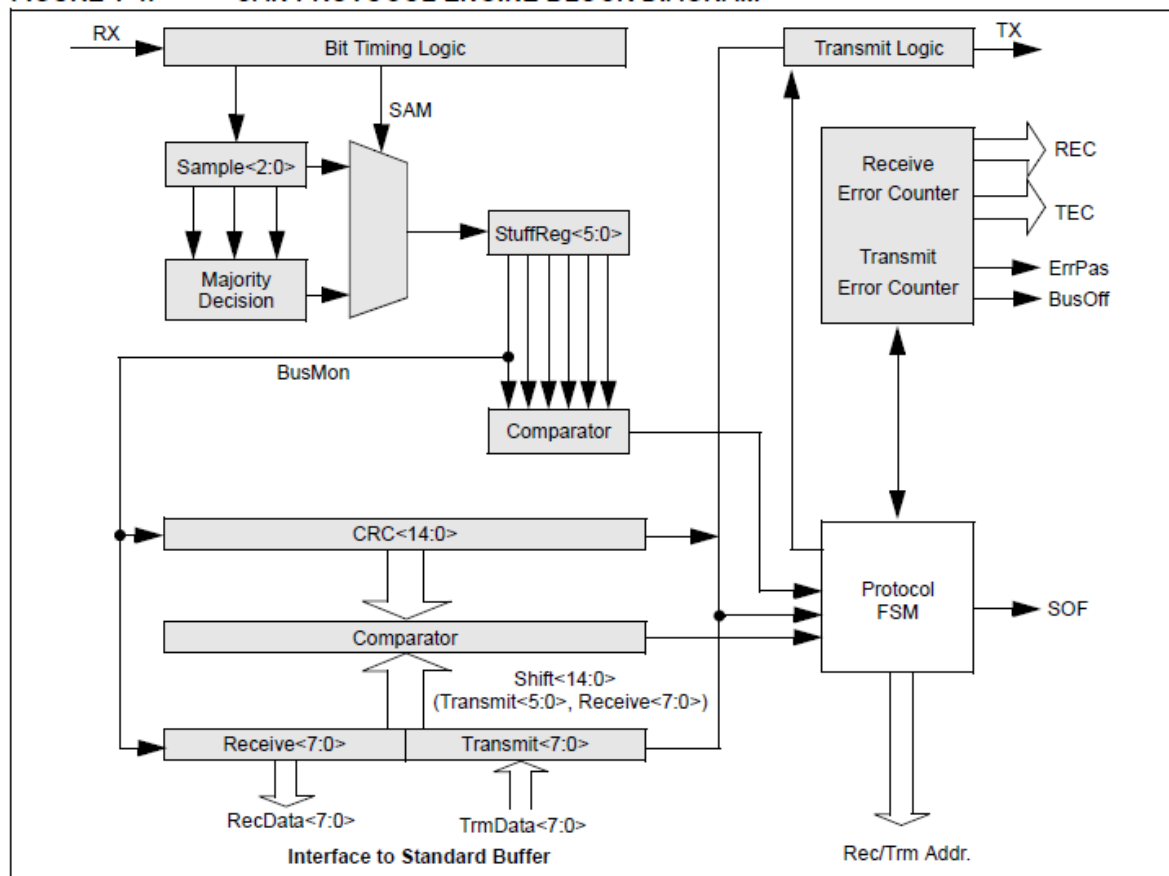
BTL (Bit Timing Logic)은 버스 라인 입력을 모니터링하고 CAN 프로토콜에 따라 버스 관련 비트 타이밍을 처리합니다.

BTL은 CAN 컨트롤러 자체가 도미넌트 비트 (재 동기화)를 전송하지 않으면 시작 프레임 (하드 동기화)에서의 우세한 버스 전환과 더 열성에서 우세한 버스 라인 전환에 동기화됩니다.

BTL은 또한 전파 지연 시간, 위상 시프트를 보상하고 비트 시간 내에 샘플 포인트의 위치를 정의 할 수있는 프로그래밍 가능한 시간 세그먼트를 제공합니다.

BTL의 프로그래밍은 보드 올 및 외부 물리적 지연 시간에 따라 다릅니다.

FIGURE 1-4: CAN PROTOCOL ENGINE BLOCK DIAGRAM



2.0 CAN 메시지 프레임

MCP2515는 (표준 및 확장) CAN 2.0B 사양에 정의 된대로 표준 데이터 프레임, 확장 데이터 프레임 및 원격 프레임을 지원합니다.

2.1 표준 데이터 프레임

CAN 표준 데이터 프레임은 그림 2-1에 나와 있습니다.

다른 모든 프레임과 마찬가지로 프레임은 SOF (Start-Of-Frame) 비트로 시작됩니다.

이 비트는 지배적 인 상태이며 모든 노드의 하드 동기화를 허용합니다.

SOF 다음에는 11 비트 식별자와 RTR (Remote Transmission Request) 비트 인 12 비트로 구성된 조정 필드가옵니다.

RTR 비트는 원격 프레임 (RTR 비트 열성)에서 데이터 프레임 (RTR 비트 도미넌트)을 구별하는 데 사용됩니다.

중재 필드 다음에 제어 필드가 있으며, 6 비트로 구성됩니다.

이 필드의 첫 번째 비트는 Identifier Extension (IDE) 비트이며 표준 프레임을 지정하기 위해 반드시 사용해야 합니다.

다음 비트 인 Reserved Bit Zero (RB0)는 예약되어 있으며 CAN 프로토콜에 의해 도미넌트 비트로 정의됩니다.

제어 필드의 나머지 4 비트는 메시지에 포함 된 데이터 바이트 수 (0 - 8 바이트)를 지정하는 데이터 길이 코드 (DLC)입니다.

제어 필드가 전송 된 데이터 바이트를 포함하고 DLC (0 - 8 바이트)에 의해 정의 된 길이입니다.

Cyclic Redundancy Check (CRC) 필드는 데이터 필드 다음에 오며 전송 오류를 탐지 하는 데 사용됩니다.

CRC 필드는 열세한 CRC 구분 문자 비트가 뒤 따르는 15 비트 CRC 시퀀스로 구성됩니다.

마지막 필드는 2 비트 ACK (Acknowledge) 필드입니다.

ACK 슬롯 비트 동안, 송신 노드는 열성 비트를 전송한다.

오류가없는 프레임을 수신 한 모든 노드는 노드가 특정 메시지를 수락하도록 구성되었는지 여부에 관계없이 도미넌트 비트를 되돌려 보내 프레임의 올바른 수신을 확인합니다.

열성 수신 확인 구분 기호는 승인 필드를 완료하며 도미넌트 비트로 덮어 쓰지 않을 수 있습니다.

2.2 확장 데이터 프레임

그림 2-2에 나와있는 확장 CAN 데이터 프레임에서 SOF 비트 다음에 32 비트로 구성된 조정 필드가옵니다.

첫 번째 11 비트는 29 비트 식별자의 최상위 비트 (MSB) (Base-1D)입니다.

이 11 비트 다음에 열성 형으로 정의 된 대체 원격 요청 (SRR) 비트가옵니다.

SRR 비트 뒤에는 확장 CAN 프레임을 나타내는 열성 인 IDE 비트가옵니다.

식별자의 처음 11 비트가 전송 된 후에 중재가 해결되지 않고 중재에 관련된 노드 중 하나가 표준 CAN 프레임 (11 비트 식별자)을 보내는 경우, 표준 CAN 프레임은 지배적 인 IDE 비트의 표명 으로 인해 중재를 이룰 것입니다

또한 확장 된 CAN 프레임의 SRR 비트는 표준 CAN 원격 프레임을 전송하는 노드에 의해 지배적 인 RTR 비트의 주장을 허용하려면 열성(서서히약해지는것)이어야합니다.

SRR 및 IDE 비트 다음에 식별자 (확장 ID)의 나머지 18 비트와 원격 전송 요청 비트가옵니다.

표준 및 확장 프레임을 공유 네트워크를 통해 전송할 수 있도록 29 비트 확장 메시지 식별자는 11 비트 (최상위) 및 18 비트 (최하위) 섹션으로 나뉩니다.

이 분할은 IDE 비트가 표준 및 확장 프레임의 동일한 비트 위치에 남아있게합니다.

중재 필드 다음에 6 비트 제어 필드가 있습니다.

이 필드의 처음 두 비트는 예약되어 있으므로 반드시 지배적이어야합니다.

제어 필드의 나머지 4 비트는 메시지에 포함 된 데이터 바이트 수를 지정하는 DLC입니다.

프레임의 나머지 부분 (데이터 필드, CRC 필드, 승인 필드, 프레임 끝 및 중단)은 표준 데이터 프레임과 동일하게 구성됩니다 (2.1 "표준 데이터 프레임"참조).

2.3 원격 프레임

통상적으로, 데이터 전송은 데이터 소스 노드 (예를 들어, 데이터 프레임을 전송하는 센서)에 의해 자율적으로 수행된다.

그러나 대상 노드가 소스에서 데이터를 요청할 수는 있습니다.

이를 달성하기 위해, 목적지 노드는 필요한 데이터 프레임의 식별자와 일치하는 식별자를 갖는 원격 프레임을 전송한다.

적절한 데이터 소스 노드는 원격 프레임 요청에 대한 응답으로 데이터 프레임을 전송합니다.

원격 프레임 (그림 2-3 참조)과 데이터 프레임에는 두 가지 차이점이 있습니다.

첫째, RTR 비트는 열성 상태에 있고, 둘째, 데이터 필드가 없습니다.

동일한 식별자를 가진 데이터 프레임과 원격 프레임이 동시에 전송되는 경우, 데이터 프레임은 식별자 다음의 지배적 인 RTR 비트로 인하여 동기를 맞출 수 있습니다.

이러한 방식으로, 원격 프레임을 전송 한 노드는 원하는 데이터를 즉시 수신합니다.

2.4 오류 프레임

오류 프레임은 버스 오류를 감지 한 모든 노드에 의해 생성됩니다.

그림 2-4에 표시된 오류 프레임은 다음과 같습니다.

두 필드 : 오류 플래그 필드 다음에 오류 구분 기호 필드가옵니다.

오류 플래그 필드에는 두 가지 유형이 있습니다.

전송 된 오류 플래그 필드 유형은 오류 플래그 필드를 감지하여 생성하는 노드의 오류 상태에 따라 다릅니다.

2.4.1 액티브 오류

오류 활성 노드가 버스 오류를 감지하면, 노드는 활성 오류 플래그를 생성하여 현재 메시지의 전송을 인터럽트합니다.

활성 에러 플래그는 6 개의 연속적인 도미넌트 비트들로 구성된다.

이 비트 시퀀스는 비트 스테핑 규칙을 능동적으로 위반합니다.

다른 모든 스테이션은 결과로 발생하는 비트 채움 오류를 인식하고 오류 에코 플래그라고하는 오류 프레임 자체를 생성합니다.

따라서 오류 플래그 필드는 6 개에서 12 개의 연속적인 지배 비트 (하나 이상의 노드에 의해 생성 됨)로 구성됩니다.

에러 구분 기호 필드 (8 개의 리세 시브 비트)는 에러 프레임을 완성합니다.

오류 프레임이 완료되면 버스 활동은 정상으로 돌아가고 중단 된 노드는 중단 된 메시지를 재전송하려고 시도합니다.

NOTE : 오류 반향 플래그는 일반적으로 국부 화 된 교란은 하나 이상의 (모든 노드는 아니지만) 노드가 오류 플래그를 보냅니다.

나머지 노드는 원래 오류 플래그에 응답하여 오류 플래그를 생성합니다 (반향).

2.4.2 수동 오류

오류 패시브 노드가 버스 오류를 감지하면 노드는 오류 패시브 플래그 다음에 오류 구분 기호 필드를 전송합니다.

에러 - 패시브 플래그는 6 개의 연속적인 리세 시브 비트로 구성됩니다.

errorpassive 노드의 오류 프레임은 14 개의 열성 비트로 구성됩니다.

이로부터, 에러 활성 노드 또는 송신 노드에 의해 버스 에러가 검출되지 않으면, 에러 - 패시브 플래그가 버스와 간섭하지 않기 때문에 메시지가 계속 송신 될 것이다.

전송 노드가 오류 패시브 플래그를 생성하면, 결과적으로 비트 스템핑 위반으로 인해 다른 노드가 오류 프레임을 생성하게됩니다.

에러 프레임 전송 후, 에러 - 패시브 노드는 버스 통신에 재결합하기 전에 버스상의 6 개의 연속적인 리세 시브 비트를 기다려야한다.

오류 구분 기호는 8 개의 리세 시브 비트로 구성되며 오류가 발생한 후 버스 노드가 버스 통신을 완전히 다시 시작하게합니다.

2.5 과부하 프레임

그림 2-5에 표시된 오버로드 프레임은 활성 오류 프레임과 동일한 형식을 갖습니다.

그러나 오버로드 프레임은 프레임 간 공간에서만 생성 될 수 있습니다.

이러한 방식으로 과부하 프레임은 오류 프레임과 구별 될 수 있습니다 (오류 프레임은 메시지 전송 중 전송됩니다).

오버로드 프레임은 두 개의 필드로 구성된다:오버로드 플래그 다음에 오버로드 구분 기호가옵니다.

과부하 플래그는 다른 노드들에 의해 생성 된 오버 플래그 뒤에 여섯 지배적 비트 (활성 에러 플래그와 같이, 열두 지배적 비트의 최대주기)로 구성된다.

과부하 구분 기호는 8 개의 리세 시브 비트로 구성됩니다.

과부하 프레임은 다음 두 조건의 결과로 노드에 의해 생성 될 수 있습니다.

1. 노드는 프레임 간 공간 동안 불법적 인 조건 인 도미넌트 비트를 검출합니다.

예외 : 지배적 비트는 IFS의 세 번째 비트 중에 감지됩니다. 이 경우 수신자는이를 SOF로 해석합니다.

2. 내부 조건으로 인해, 노드는 아직 다음 메시지의 수신을 시작할 수 없습니다.

노드는 최대 두 개의 순차적 오버로드 프레임을 생성하여 다음 메시지의 시작을 지연시킬 수 있습니다.

2.6 프레임 간 공간

프레임 간 공간은 선행 프레임 (모든 유형의)을 후속 데이터 또는 원격 프레임과 구분합니다.

프레임 간 공간은 중단 (Intermission)이라고하는 적어도 세 개의 열성 비트로 구성됩니다.

이렇게하면 노드가 다음 메시지 프레임이 시작되기 전에 내부 처리를위한 시간을 가질 수 있습니다.

차단 후 버스 라인은 다음 전송이 시작될 때까지 열성 상태 (버스 유휴 상태)로 유지됩니다.

FIGURE 2-1: STANDARD DATA FRAME

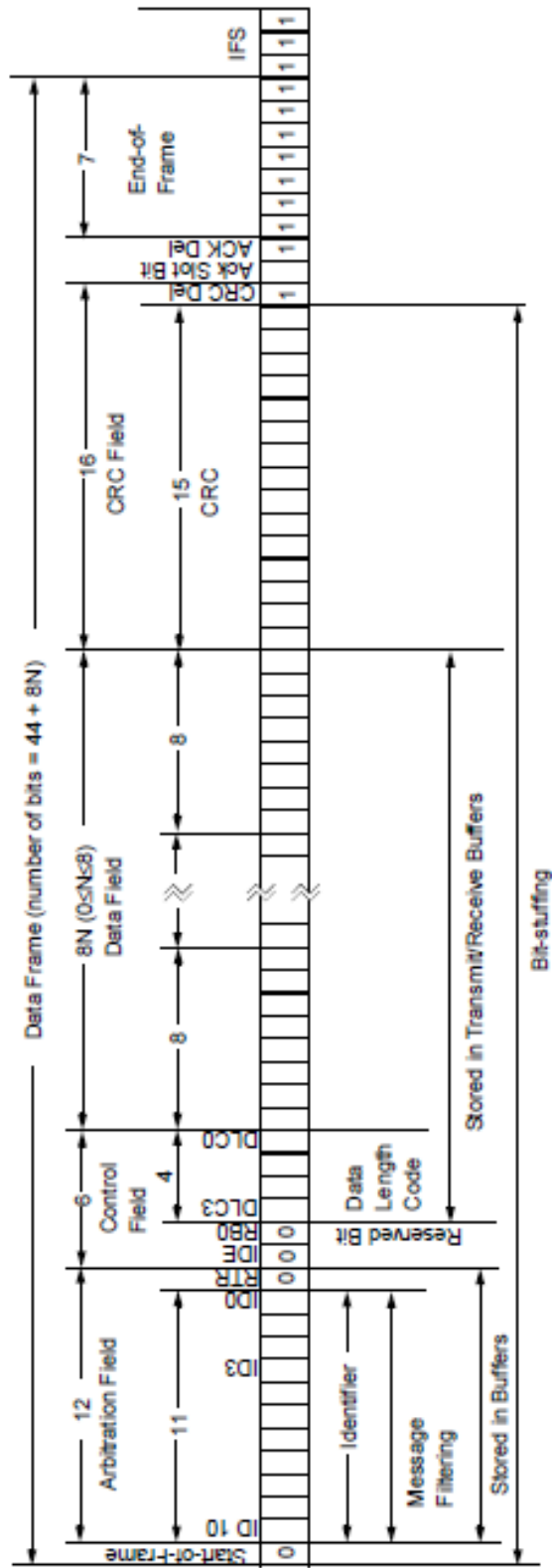


FIGURE 2-2: EXTENDED DATA FRAME

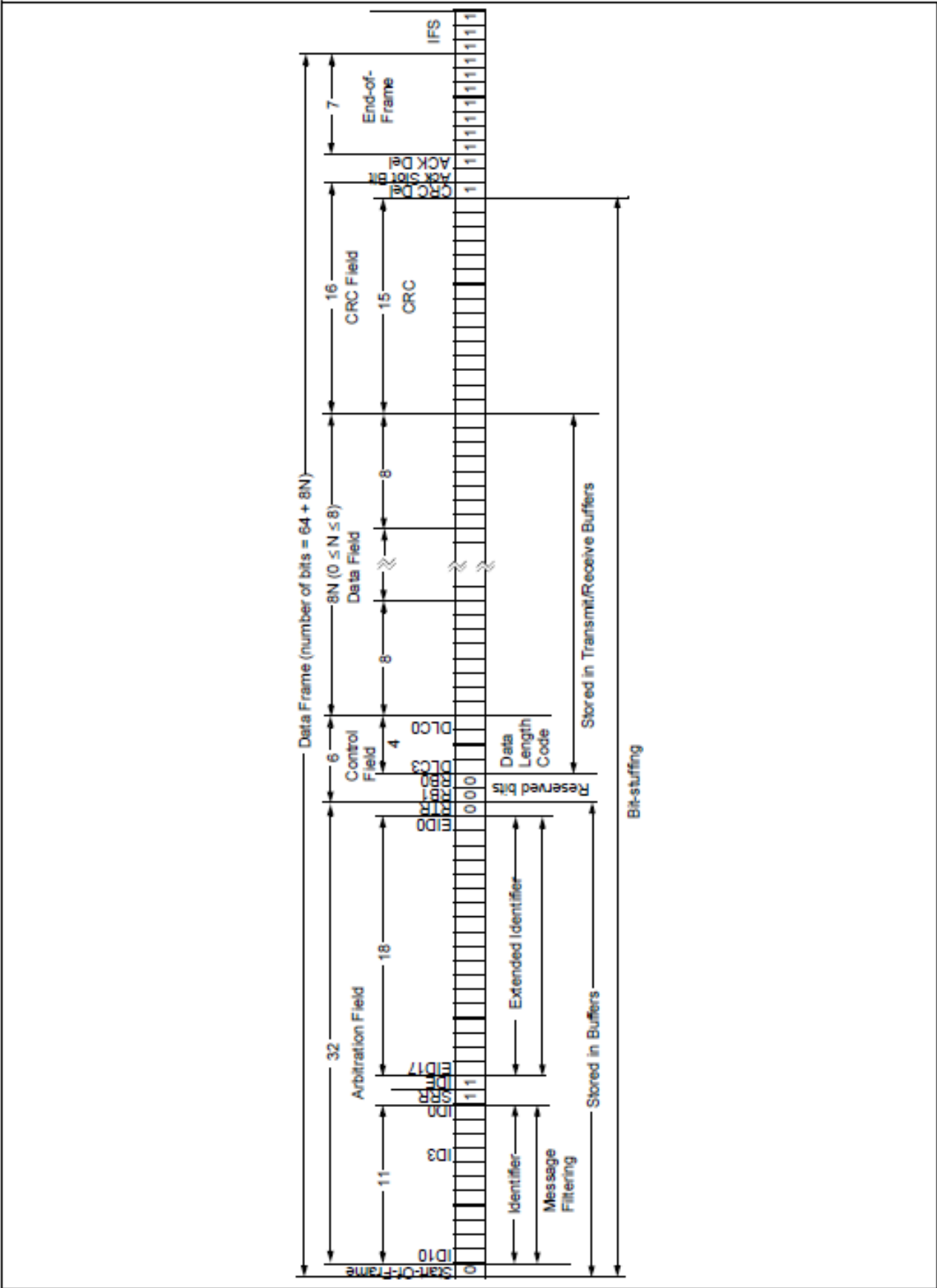


FIGURE 2-3: REMOTE FRAME

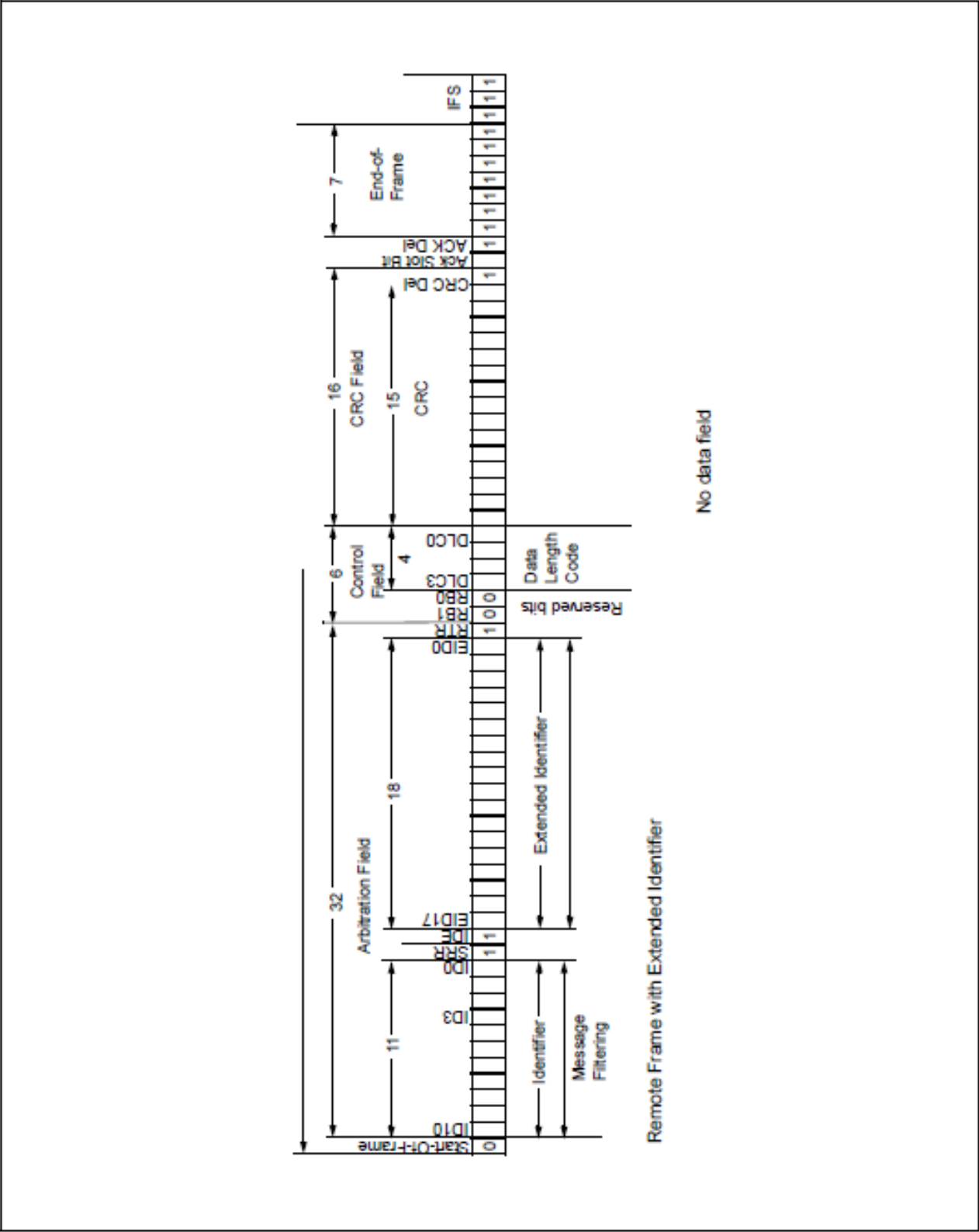
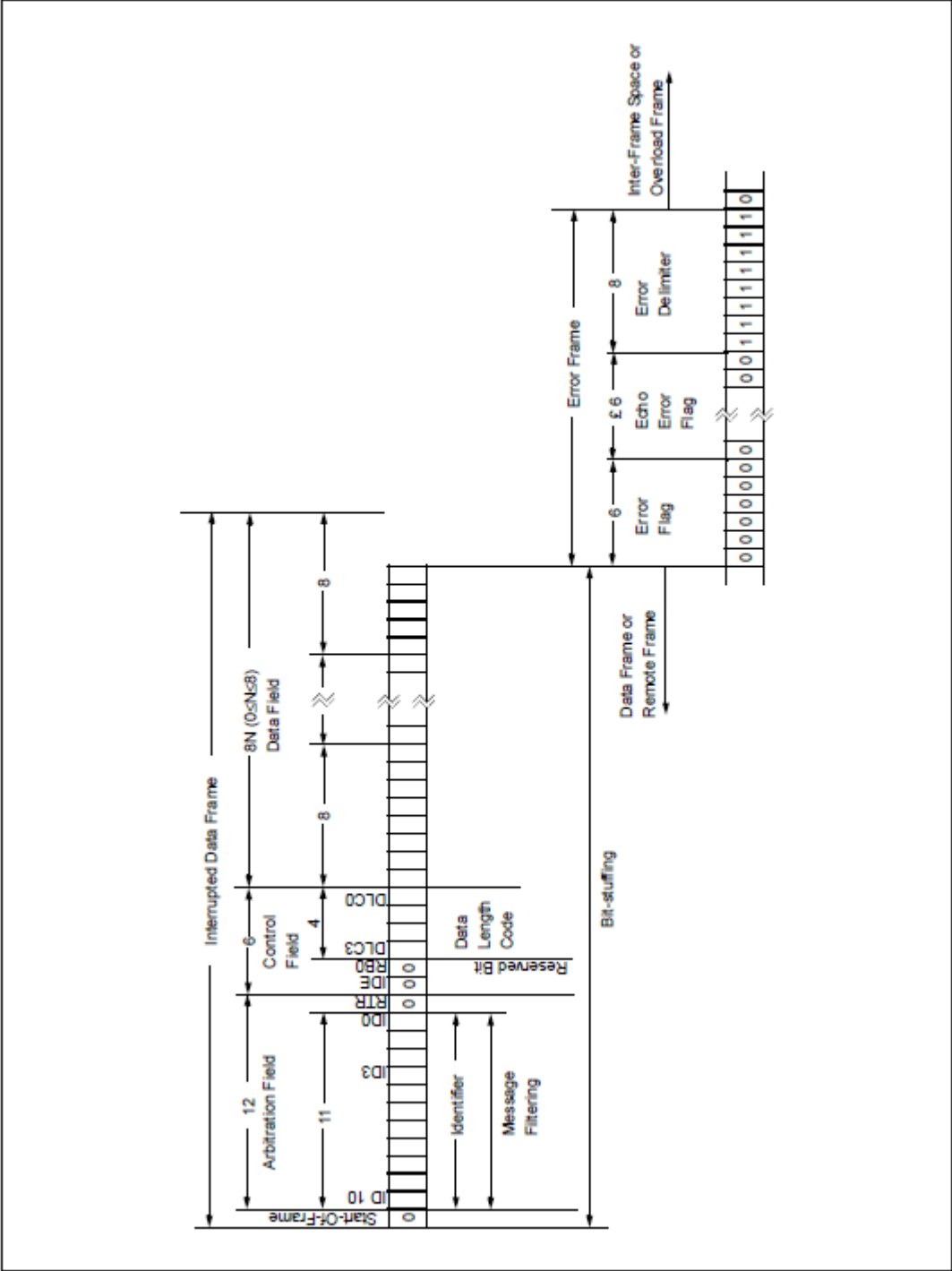


FIGURE 2-4: ACTIVE ERROR FRAME



The diagram illustrates the structure of a CAN frame. The main frame consists of the following fields:

- Start-Of-Frame**: Indicated by a '0' bit.
- Arbitration Field**: 12 bits long.
- Control Field**: 6 bits long, containing sub-fields: ID (10), RTR (1), IDE (1), RBD (1), and DLC3 (1).
- CRC Field**: 16 bits long, containing a CRC (15) and a CRC Delimiter (1).
- End-of-Frame**: 7 bits long, containing an ACK Slot Delimiter (1) and an End-of-Frame (6).

Below the main frame, an **Overload Frame** structure is shown:

- Overload Flag**: 6 bits, all set to 0.
- Overload Delimiter**: 8 bits, all set to 1.
- Inter-Frame Space or Error Frame**: The period following the overload frame.

3.0 메시지 전송

3.1 송신 버퍼

MCP2515는 3 개의 전송 버퍼를 구현합니다.

이들 각각의 버퍼는 14 바이트의 SRAM을 차지하며 장치 메모리 맵에 맵핑된다.

첫 번째 바이트 인 TXBnCTRL은 메시지 버퍼와 관련된 제어 레지스터입니다.

이 레지스터의 정보는 메시지가 전송되는 조건을 결정하고 메시지 전송 상태를 나타냅니다 (레지스터 3-2 참조).

5 바이트는 표준 및 확장 식별자뿐만 아니라 다른 메시지 중재 정보 (레지스터 3-4에서 레지스터 3-7 참조)를 저장하는 데 사용됩니다.

마지막 8 바이트는 메시지의 8 개의 가능한 데이터 바이트에 대한 것입니다 (레지스터 3-8 참조).

최소한 TXBnSIDH, TXBnSIDL 및 TXBnDLC 레지스터가 로드되어야합니다.

데이터 바이트가 메시지에 있으면 TXBnDm 레지스터도 로드해야합니다.

메시지가 확장 식별자를 사용하는 것이라면, TXBnEIDm 레지스터도 또한 로드되어야하고 TXBnSIDL도 있어야합니다. EXIDE 비트는 설정되었습니다.

메시지를 보내기 전에, MCU는 CANINTE를 초기화해야합니다.

TXInE 비트는 메시지가 전송 될 때 인터럽트 생성을 활성화 또는 비활성화합니다.

3.2 전송 우선 순위

전송 우선 순위는 보류중인 전송 가능 메시지의 MCP2515 내 우선 순위입니다.

이것은 어떤 것과도 관련이 없으며 반드시 관련이있는 것은 아닙니다.

메시지 중재 계획에 내포 된 우선 순위

CAN 프로토콜에 내장되어 있습니다.

SOF를 보내기 전에 모든 버퍼의 우선 순위
전송을 위해 큐에 대기 중인지를 비교합니다.

우선 순위가 가장 높은 전송 버퍼가 먼저 전송됩니다.

예를 들어, 송신 버퍼 0이 더 높은 우선 순위 설정
전송 버퍼 1보다 버퍼 0이 우선 전송됩니다.

두 개의 버퍼가 동일한 우선 순위 설정을 갖는 경우, 버퍼
가장 높은 버퍼 번호가 먼저 전송됩니다.

예를 들어, 송신 버퍼 1이 송신 버퍼 0과 동일한 우선 순위 설정을 갖는 경우, 버퍼 1이 먼저
송신됩니다.

전송 우선 순위에는 4 가지 레벨이 있습니다.

특정 메시지 버퍼에 대해 TXBnCTRL.TXP <1 : 0>
가 11로 설정된 경우 해당 버퍼는 가능한 가장 높은 우선 순위를 갖습니다.

특정 메시지 버퍼의 TXBnCTRL.TXP <1 : 0>이 00이면 해당 버퍼는 가능한 가장 낮은 우선
순위를 갖습니다.

3.3 전송 개시

메시지 전송을 시작하기 위해 TXBnCTRL.

전송할 각 버퍼마다 TXREQ 비트를 설정해야 합니다.

이것은 다음과 같이 수행 할 수 있습니다.

- SPI 쓰기 명령을 통해 레지스터에 쓰기
- SPI RTS 명령 보내기
- 송신 할 특정 송신 버퍼에 대해 TXNRTS 핀을 로우로 설정한다.

SPI 인터페이스를 통해 전송이 시작되면 TXREQ 비트는 TXP 우선 순위 비트와 동시에
설정할 수 있습니다.

TXBnCTRL.TXREQ가 설정되면 TXBnCTRL.ABTF, TXBnCTRL.MLOA와 TXBnCTRL.TXERR 비트는 자동으로 소거됩니다.

NOTE : TXBnCTRL.TXREQ 비트를 설정해도 메시지 전송은 시작되지 않습니다.

단순히 메시지 버퍼에 전송할 준비가되었다고 플래그를 지정합니다.

장치가 사용 가능한 버스를 감지하면 전송이 시작됩니다.

전송이 성공적으로 완료되면

TXBnCTRL.TXREQ 비트가 클리어되고 CANINTF.TXnIF 비트가 설정되고 CANINTE.TXnIE 비트가 설정되면 인터럽트가 생성됩니다.

메시지 전송이 실패하면 TXBnCTRL.TXREQ가 설정된 상태로 유지됩니다.

이는 메시지가 아직 보류 중이며 다음 조건 플래그 중 하나가 설정됨을 나타냅니다.

- 메시지가 전송되기 시작했지만 오류 상태가 발생하면 TXBnCTRL.TXERR 및 CANINTF.MERRF 비트가 설정되고 CANINTE.MERRE 비트가 설정되면 INT 핀에 인터럽트가 생성됩니다

- 메시지가 분실되면 TXBnCTRL.MLOA 비트의 조정이 설정됩니다

NOTE : 원샷 모드가 활성화 된 경우 (CANCTRL.OSM) 위의 조건은 계속 존재합니다.

그러나 TXREQ 비트는 지워지고 메시지는 전송을 다시 시도하지 않습니다.

3.4 원샷 모드

원샷 모드는 메시지가 한 번만 전송되도록합니다.

일반적으로 CAN 메시지가 중재를 잃거나 오류 프레임에 의해 손상되면 메시지가 재전송됩니다.

원샷 모드를 사용하면 메시지는 중재 손실이나 오류 프레임과 상관없이 한 번만 전송을 시도합니다.

TTCAN과 같은 결정 론적 시스템에서 타임 슬롯을 유지하려면 원샷 모드가 필요합니다.

3.5 TXnRTS PINS

TXnRTS 핀은 다음과같이 구성 할 수있는 입력 핀이 있다.

- Request-to-send 입력 : 전송 버퍼 중 하나에서 메시지 전송을 시작하는 대체 수단을 제공합니다.

- 표준 디지털 입력

이 핀의 구성 및 제어는 TXRTSCTRL 레지스터 (레지스터 3-3 참조)를 사용하여 수행된다.

TXRTSCTRL 레지스터는 MCP2515가 Configuration 모드에있을 때만 수정할 수 있습니다 (10.0 절 "동작 모드"참조).

요청 송신 핀으로 작동하도록 구성된 경우, 핀은 송신 버퍼의 TXBnCTRL.TXREQ 비트에 매핑된다.

TXREQ 비트는 TXnRTS 핀의 하강 에지에 의해 닫히게된다.

TXnRTS 핀은 RXnBF 핀이 로우가 될 때 자동으로 메시지 전송을 시작하기 위해 RXnBF 핀에 직접 연결되도록 설계되어있다.

TXnRTS 핀은 100k Ω (공칭)의 내부 풀업 저항을 갖는다.

3.6 전송 중단

MCU는 관련된 TXBnCTRL.TXREQ 비트를 지움으로써 특정 메시지 버퍼에서 메시지를 중단하도록 요청할 수있다.

또한 대기중인 모든 메시지는 CANCTRL.ABAT 비트를 설정하여 중단되도록 요청할 수 있습니다.

이 비트는 반드시 리셋되어야한다 (일반적으로 TXREQ 비트가 삭제 된 것으로 확인 된 메시지)를 전송합니다.

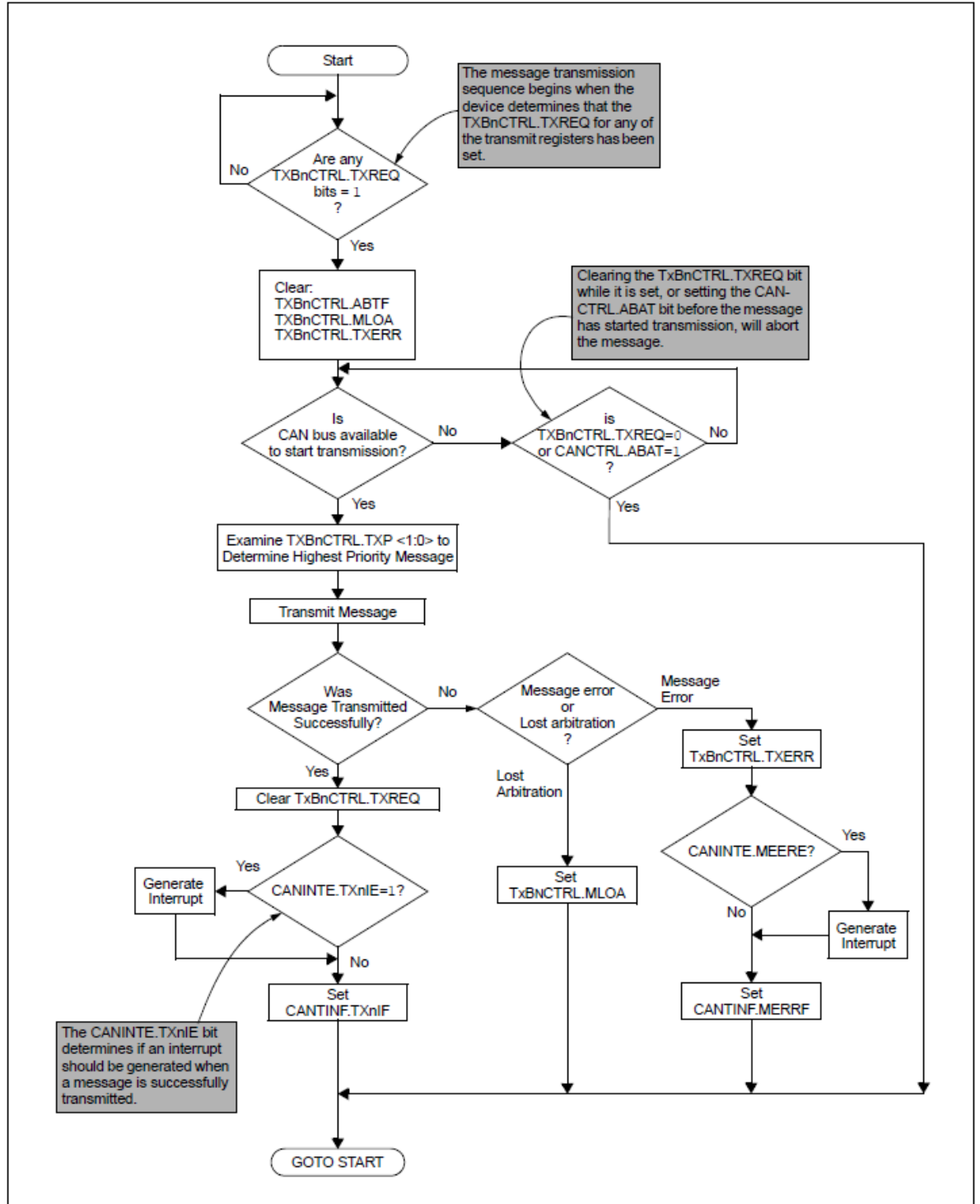
CANCTRL.ABTF 플래그는 CANCTRL.ABAT 비트를 통해 중단이 요청 된 경우에만 설정됩니다.

TXREQ 비트를 재설정하여 메시지를 중단하면 ABTF 비트를 설정하지 마십시오.

NOTE : 중단 요청시 전송중인 메시지는 계속 전송됩니다.

메시지가 전송을 성공적으로 완료하지 못하면 (즉, 중재가 손실되거나 오류 프레임에 의해 중단 된 경우) 오류가 중단됩니다.

FIGURE 3-1: TRANSMIT MESSAGE FLOWCHART



REGISTER 3-1: TXBnCTRL – TRANSMIT BUFFER n CONTROL REGISTER
(ADDRESS: 30h, 40h, 50h)

U-0	R-0	R-0	R-0	R/W-0	U-0	R/W-0	R/W-0
—	ABTF	MLOA	TXERR	TXREQ	—	TXP1	TXP0
bit 7		bit 0					

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

bit 7 : '0'을 읽음 으로서 구현을 하지 않음.

bit 6 ABTF : 메시지 중단 플래그 비트

1 = 메시지가 중단되었습니다.

0 = 메시지 전송 완료

bit 5 MLOA : 메시지 손실 중재 비트

1 = 전송 중 메시지 손실 중재

0 = 메시지가 전송되는 동안 중재를 잃지 않았습니다.

bit 4 TXERR : 전송 오류 감지 비트

1 = 메시지가 전송되는 동안 버스 오류가 발생했습니다.

0 = 메시지를 전송하는 동안 버스 오류가 발생하지 않았습니다.

bit 3 TXREQ : 메시지 송신 요구 비트

1 = 버퍼가 현재 전송 대기 중입니다.

(MCU는이 비트를 설정하여 메시지 전송을 요청합니다. 비트는 자동으로 삭제됩니다.

메시지가 전송됩니다)

0 = 버퍼는 현재 전송이 보류 중이 아닙니다.

(MCU는이 비트를 지워서 메시지 중단을 요청할 수 있음)

bit 2: '0'을 읽음 으로서 구현을 하지 않음.

bit 1-0 TXP : 전송 버퍼 우선 순위 <1 : 0> 비트

- 11 = 가장 높은 메시지 우선 순위
- 10 = 높은 중간 메시지 우선 순위
- 01 = 중간 중간 메시지 우선 순위 낮음
- 00 = 가장 낮은 메시지 우선 순위

**REGISTER 3-2: TXRTSCTRL – TXnRTS PIN CONTROL AND STATUS REGISTER
(ADDRESS: 0Dh)**

U-0	U-0	R-x	R-x	R-x	R/W-0	R/W-0	R/W-0
—	—	B2RTS	B1RTS	B0RTS	B2RTSM	B1RTSM	B0RTSM
bit 7							
							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7: '0'을 읽음 으로서 구현을 하지 않음.

bit 6: '0'을 읽음 으로서 구현을 하지 않음.

bit 5 B2RTS : TX2RTS 핀 상태 비트

- 디지털 입력 모드 일 때 TX2RTS 핀의 상태를 읽습니다.
- 핀이 'Request-to-Send'모드 일 때 '0'으로 읽음

bit 4 B1RTS : TX1RTX 핀 상태 비트

- 디지털 입력 모드 일 때 TX1RTS 핀의 상태를 읽습니다.
- 핀이 'Request-to-Send'모드 일 때 '0'으로 읽음

bit 3 B0RTS : TX0RTS 핀 상태 비트

- 디지털 입력 모드 일 때 TX0RTS 핀의 상태를 읽습니다.
- 핀이 'Request-to-Send'모드 일 때 '0'으로 읽음

bit 2 B2RTSM : TX2RTS 핀 모드 비트

- 1 = 핀은 TXB2 버퍼의 메시지 전송을 요청하는 데 사용됩니다 (하강 에지에서)
- 0 = 디지털 입력

bit 1 B1RTSM : TX1RTS 핀 모드 비트

- 1 = 핀은 TXB1 버퍼의 메시지 전송을 요청하는 데 사용됩니다 (하강 에지에서)
- 0 = 디지털 입력

bit 0 B0RTSM : TX0RTS 핀 모드 비트

1 = 핀은 TXB0 버퍼의 메시지 전송을 요청하는 데 사용됩니다 (하강 에지에서)

0 = 디지털 입력

**REGISTER 3-3: TXBnSIDH – TRANSMIT BUFFER n STANDARD IDENTIFIER HIGH
(ADDRESS: 31h, 41h, 51h)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 SID : 표준 식별자 비트 Standard Identifier bits <10 : 3>

**REGISTER 3-4: TXBnSIDL – TRANSMIT BUFFER n STANDARD IDENTIFIER LOW
(ADDRESS: 32h, 42h, 52h)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-5 SID : 표준 식별자 비트 Standard Identifier bits <2 : 0>

bit 4 비 구현 : '0'

bit 3 EXIDE : 확장 식별자 가능 비트

1 = 메시지가 확장 식별자를 전송합니다.

0 = 메시지는 표준 식별자를 전송합니다.

bit 2 비 구현 : '0'

bit 1-0 EID : 확장 식별자 비트 Extended Identifier bits <17:16>

**REGISTER 3-5: TXBnEID8 – TRANSMIT BUFFER n EXTENDED IDENTIFIER HIGH
(ADDRESS: 33h, 43h, 53h)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 EID : 확장 식별자 비트 Extended Identifier bits <15 : 8>

**REGISTER 3-6: TXBnEID0 – TRANSMIT BUFFER n EXTENDED IDENTIFIER LOW
(ADDRESS: 34h, 44h, 54h)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 EID : 확장 식별자 비트 Extended Identifier bits <7 : 0>

**REGISTER 3-7: TXBnDLC - TRANSMIT BUFFER n DATA LENGTH CODE
(ADDRESS: 35h, 45h, 55h)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	RTR	—	—	DLC3	DLC2	DLC1	DLC0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7 비 구현 : '0'

bit 6 RTR : 원격 전송 요청 비트

1 = 전송 된 메시지는 원격 전송 요청이됩니다.

0 = 전송 된 메시지는 데이터 프레임이됩니다.

bit 5-4 비 실행 : '0'

bit 3-0 DLC : 데이터 길이 코드 <3 : 0> 비트 전송할 데이터 바이트 수를 설정합니다 (0 ~ 8 바이트).

참고 : DLC를 8보다 큰 값으로 설정할 수 있지만 8 바이트 만 전송할 수 있습니다.

REGISTER 3-8: TXBnDm – TRANSMIT BUFFER n DATA BYTE m
(ADDRESS: 36h - 3Dh, 46h - 4Dh, 56h - 5Dh)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
TXBnDm7	TXBnDm6	TXBnDm5	TXBnDm4	TXBnDm3	TXBnDm2	TXBnDm1	TXBnDm0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 TXBnDM7 : TXBnDM0 : 송신 버퍼 n 데이터 필드 바이트 m

4.0 메시지 수신

4.1 메시지 버퍼링 받기

MCP2515는 2 개의 수신 버퍼와 각각의 수신 필터를 포함합니다.

세 번째 수신 버퍼 역할을하는 별도의 메시지 어셈블리 버퍼 (MAB)도 있습니다 (그림 4-2 참조).

4.1.1 메시지 배정 버퍼 (MESSAGE ASSEMBLY BUFFER)

3 개의 수신 버퍼 중 MAB는 항상 버스에서 다음 메시지를 수신하기 위해 최선을 다하고 있습니다.

MAB는 수신 된 모든 메시지를 어셈블합니다.

이러한 메시지는 acceptancefilter 기준이 충족되는 경우에만 RXBn 버퍼로 전송됩니다 (레지스터 4-4에서 레지스터 4-9 참조).

4.1.2 RXB0 및 RXB1

RXB0 및 RXB1이라고하는 나머지 두 개의 수신 버퍼는 MAB를 통해 프로토콜 엔진으로부터 완전한 메시지를 수신 할 수 있습니다.

MCU는 하나의 버퍼를 액세스 할 수있는 반면, 다른 버퍼는 메시지 수신 또는 이전에 수신 된 메시지를 유지하는 데 사용할 수 있습니다.

NOTE : 메시지가 수락되면 MAB의 전체 내용이 수신 버퍼로 이동됩니다.

즉, 식별자 유형 (표준 또는 확장) 및 수신 된 데이터 바이트 수에 관계없이 전체 수신 버퍼가 MAB 내용으로 덮어 쓰여집니다.

따라서 버퍼에있는 모든 레지스터의 내용은 메시지가 수신 될 때 수정 된 것으로 가정해야 합니다.

4.1.3 수신 플래그 / 인터럽트

메시지가 수신 버퍼 중 하나로 이동하면 적절한 CANINTF.RXnIF 비트가 설정됩니다.

새 메시지를 버퍼로 수신하려면 이 비트를 MCU가 지워야 합니다.

이 비트는 MCP2515가 새로운 메시지를 수신 버퍼에로드하기 전에 MCU가 메시지를 완료했는지 확인하기 위해 긍정적 인 잠금을 제공합니다.

CANINTE.RXnIE 비트가 설정되면 INT 핀에 인터럽트가 발생하여 유효한 메시지가 수신되었음을 나타냅니다.

또한 RXnBF 핀은 수신 버퍼 풀 핀으로 구성된 경우 low로 구동된다.자세한 내용은 4.4 절 "RX0BF 및 RX1BF 핀"을 참조하십시오.

4.2 수신 우선 순위

우선 순위가 더 높은 버퍼 인 RXB0에는 하나의 마스크와 두개의 메세지 수용 필터가 연결되어 있습니다.

수신 된 메시지는 먼저 마스크에 적용되고 RXB0에 대한 필터가 먼저 적용됩니다.

RXB1은 하나의 마스크와 4 개의 수락 필터가 연관된 하위 우선 순위 버퍼입니다.

메시지가 RB0 마스크와 필터에 먼저 적용되는 것 외에도 수용 필터의 수가 적 으면 RXB0에서의 일치가 더 제한적이게되고 해당 버퍼에 대해 더 높은 우선 순위가 부여됩니다.

메시지가 수신되면 RXBnCTRL 레지스터의 비트 <3 : 0>은 수신을 활성화 한 수용 필터 번호와 수신 된 메시지가 원격 전송 요청인지 여부를 나타냅니다.

4.2.1 롤오버

또한 RXB0CTRL 레지스터는 RXB0에 유효한 메시지가 있고 다른 유효한 메시지가 수신되면 오버플로 오류가 발생하지 않고 RXB1의 승인 기준에 관계없이 새 메시지가 RXB1로 이동됩니다.

4.2.2 RXM 비트

RXBnCTRL.RXM 비트는 특수 수신 모드를 설정합니다.

일반적으로이 비트들은 00으로 클리어되어 적절한 승인 필터에 의해 결정된 모든 유효한 메시지의 수신을 가능하게합니다.

이 경우 표준 또는 확장 메시지를 수신할지 여부는 수용 필터 레지스터의 RFXnSIDL.EXIDE 비트에 의해 결정됩니다.

RXBnCTRL.RXM 비트가 01 또는 10으로 설정된 경우 수신자는 각각 표준 또는 확장 식별자가있는 메시지 만 수락합니다.

수용 필터의 RFXnSIDL.EXIDE 비트가 RXBnCTRL.RXM 모드와 일치하지 않도록 설정된 경우 해당 수용 필터는 쓸모 없게 렌더링됩니다.

이 두 가지 RXBnCTRL.RXM 비트 모드는 표준 또는 확장 메시지 만 버스에 있다는 것을 알고있는 인 시스템에서 사용할 수 있습니다.

RXBnCTRL.RXM 비트가 11로 설정되면 버퍼는 승인 필터의 값에 관계없이 모든 메시지를 수신합니다.

또한 메시지가 EOF 이전에 오류가있는 경우 오류 프레임 전에 MAB에 어셈블 된 메시지 부분이 버퍼로로드됩니다.

이 모드는 CAN 시스템 디버깅에 약간의 가치가 있으며 실제 시스템 환경에서는 사용되지 않습니다.

4.3 프레임 시작 신호

활성화 된 경우 Start-Of-Frame 신호는 RXCAN 핀에서 감지 된 각 CAN 메시지의 시작 부분에서 SOF 핀에 생성됩니다.

RXCAN 핀은 휴면 버스가 recessive-dominant 에지를 모니터링합니다.

지배적 인 조건이 샘플 포인트까지 남아 있으면 DSTEMP는이를 SOF로 해석하고 SOF 펄스가 생성됩니다.

지배적 인 조건이 샘플 포인트까지 남아 있지 않으면 DSTEMP는이를 버스에서 글리치로 해석하고 SOF 신호가 생성되지 않습니다.

그림 4-1은 SOF 신호 및 글리치 필터링을 보여줍니다.

원샷 모드와 마찬가지로, SOF 신호 전달을위한 용도 중 하나는 TTCAN 유형 시스템을위한 것입니다.

또한 RXCAN 핀과 SOF 핀을 모두 모니터링함으로써 MCU는 CAN 통신에 영향을 미치기 전에 작은 글리치를 감지함으로써 초기 물리적 버스 문제를 감지 할 수있다.

4.4 RX0BF 및 RX1BF 핀

다양한 조건에 대해 MCU에 인터럽트 신호를 제공하는 INT 핀 이외에 수신 버퍼 풀 핀 (RX0BF 및 RX1BF)을 사용하여 RXB0 또는 RXB1에 유효한 메시지가 각각로드되었음을 나타낼 수있다.

핀은 세 가지 다른 구성을 가지고 있습니다 (레지스터 4-1) :

1. 비활성화 됨.
2. 버퍼 완전 인터럽트.

3. 디지털 출력.

4.4.1 DISABLED

RXBnBF 핀은 BFPCTRL.BnBFE를 클리어하여 하이 임피던스 상태로 비활성화 할 수있다.

4.4.2 BUFFER FULL로 설정 됨

RXBnBF 핀은 버퍼 풀 인터럽트 핀 또는 표준 디지털 출력으로 동작하도록 구성 할 수있다.

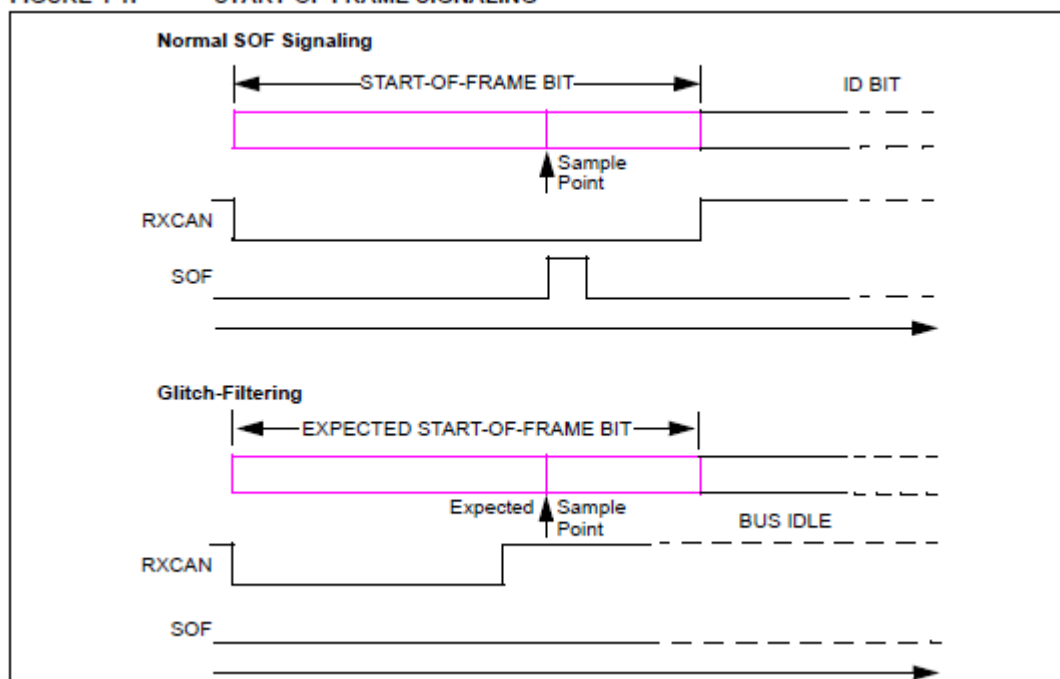
이 핀의 구성 및 상태는 BFPCTRL 레지스터 (레지스터 4-3)를 통해 사용할 수 있습니다.

인터럽트 모드 (BFPCTRL.BxBFE 및 BFPCTRL.BxBFM 비트 설정)로 작동하도록 설정되면 이 핀은 액티브 로우이며 각 수신 버퍼에 대한 CANINTF.RXnIF 비트에 매핑됩니다.

이 비트가 수신 버퍼 중 하나에 대해 하이가되면 (유효한 메시지가 버퍼에로드되었음을 나타냄), 해당 RXBnBF 핀은 로우가된다.

CANINTF.RXnIF 비트가 MCU에 의해 클리어되면, 다음 메시지가 수신 버퍼에로드 될 때까지 해당 인터럽트 핀이 로직 하이 상태가됩니다.

FIGURE 4-1: START-OF-FRAME SIGNALING



4.4.3 디지털 출력으로 구성

디지털 출력으로 사용될 때, BFPCTRL.BxBFM 비트는 지워져야 하며 BFPCTRL.BnBFE는 관련 버퍼에 설정되어야 합니다.

이 모드에서, 핀의 상태는 BFPCTRL.BnBFS 비트에 의해 제어된다.

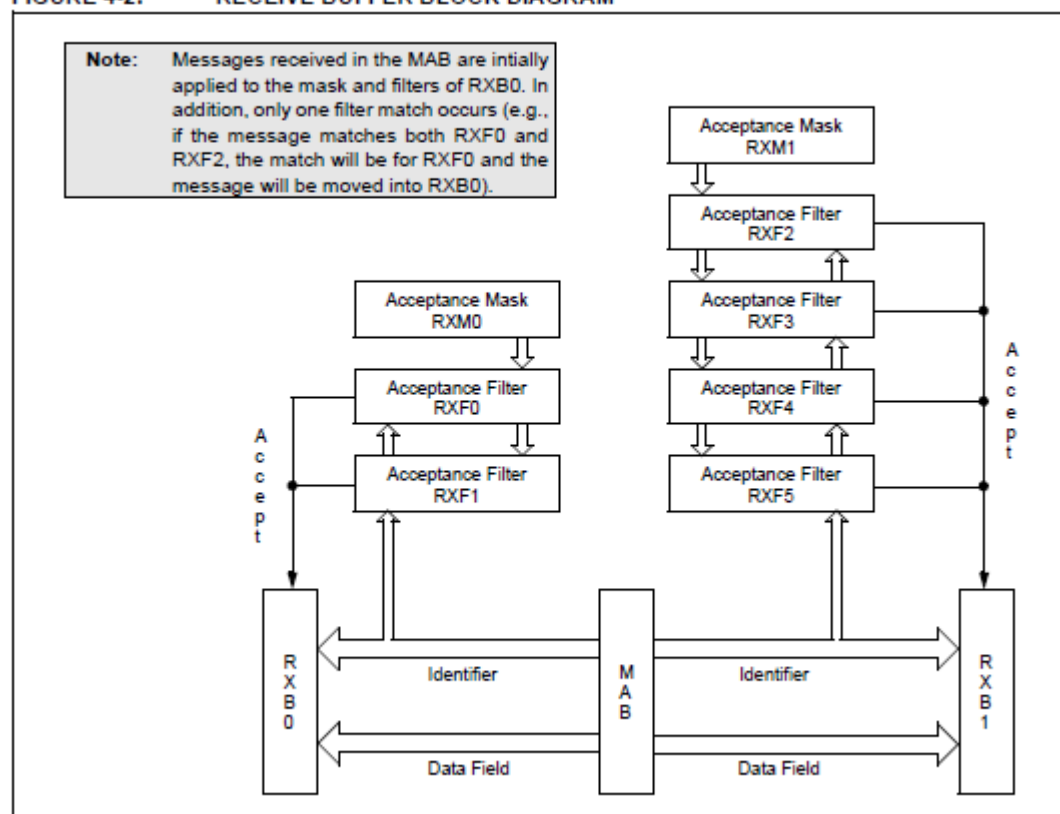
BnBFS 비트에 '1'을 쓰면 관련 버퍼 풀 핀에서 하이 레벨이 구동되고 '0'이면 핀이 로우로 구동된다.

이 모드에서 핀을 사용할 때, 핀의 상태는 비트 수정 SPI 명령을 사용해서 만 버퍼 전체 핀 중 하나에서 글리치가 발생하는 것을 방지해야 한다.

TABLE 4-1: CONFIGURING RXNBF PINS

BnBFE	BnBFM	BnBFS	Pin Status
0	X	X	Disabled, high-impedance
1	1	X	Receive buffer interrupt
1	0	0	Digital output = 0
1	0	1	Digital output = 1

FIGURE 4-2: RECEIVE BUFFER BLOCK DIAGRAM



```

graph TD
    Start([Start]) --> Detect{Detect Start of Message?}
    Detect -- No --> Detect
    Detect -- Yes --> Load[Begin Loading Message into Message Assembly Buffer (MAB)]
    Load --> Valid{Valid Message Received?}
    Valid -- No --> Error[Generate Error Frame]
    Error --> Detect
    Valid -- Yes --> FilterRXB0{Meets a filter criteria for RXB0?}
    FilterRXB0 -- No --> FilterRXB1{Meets a filter criteria for RXB1?}
    FilterRXB1 -- No --> Start
    FilterRXB1 -- Yes --> CANINTFRX1IF{Is CANINTF.RX1IF = 0?}
    CANINTFRX1IF -- No --> CANINTEERRIE{Is CANINTE.ERRIE = 1?}
    CANINTEERRIE -- No --> Start
    CANINTEERRIE -- Yes --> GenerateINT[Generate Interrupt on INT]
    GenerateINT --> CANSTAT[Set CANSTAT <3:0> according to which receive buffer the message was loaded into]
    CANSTAT -- RXB1 --> BPFCTRLB1BFM{Are BFPCTRL.B1BFM = 1 and BF1CTRL.B1BFE = 1?}
    BPFCTRLB1BFM -- Yes --> SetRXBF1[Set RXBF1 Pin = 0]
    SetRXBF1 --> CANSTAT
    BPFCTRLB1BFM -- No --> CANSTAT
    CANSTAT -- RXB0 --> BFPCTRLB0BFM{Are BFPCTRL.B0BFM = 1 and BF1CTRL.B0BFE = 1?}
    BFPCTRLB0BFM -- Yes --> SetRXBF0[Set RXBF0 Pin = 0]
    SetRXBF0 --> CANSTAT
    BFPCTRLB0BFM -- No --> CANSTAT
    CANSTAT --> CANINTERX0IE{Is CANINTE.RX0IE = 1?}
    CANINTERX0IE -- No --> CANINTFRX0IF{Is CANINTF.RX0IF = 0?}
    CANINTFRX0IF -- No --> RXB0Full{Is RXB0CTRL.BUKT = 1?}
    RXB0Full -- No --> OverflowRX0[Generate Overflow Error: Set EFLG.RX0OVR]
    OverflowRX0 --> CANINTEERRIE
    RXB0Full -- Yes --> CANINTFRX1IF
    CANINTFRX0IF -- Yes --> MoveRXB0[Move message into RXB0]
    MoveRXB0 --> SetCANINTFRX0IF[Set CANINTF.RX0IF = 1]
    SetCANINTFRX0IF --> SetRXB0CTRLFILHIT[Set RXB0CTRL.FILHIT <0> according to which filter criteria]
    SetRXB0CTRLFILHIT --> CANINTERX0IE
    CANINTERX0IE -- Yes --> GenerateINT
    CANINTERX0IE -- No --> CANINTFRX0IF
    
```

The flowchart illustrates the CAN Message Reception Process. It begins with a 'Start' terminal, leading to a decision 'Detect Start of Message?'. If 'No', it loops back. If 'Yes', it proceeds to 'Begin Loading Message into Message Assembly Buffer (MAB)'. This leads to a decision 'Valid Message Received?'. If 'No', it goes to 'Generate Error Frame' and loops back. If 'Yes', it checks 'Meets a filter criteria for RXB0?'. If 'No', it checks 'Meets a filter criteria for RXB1?'. If 'No' here, it goes to 'Go to Start'. If 'Yes', it checks 'Is CANINTF.RX1IF = 0?'. If 'No', it checks 'Is CANINTE.ERRIE = 1?'. If 'No', it goes to 'Go to Start'. If 'Yes', it generates an interrupt on INT, sets CANSTAT, and checks for pin configuration (RXBF1 or RXBF0). If 'No' for either, it loops back to the RXB1 filter check. If 'Yes' for either, it loops back to the RXB0 filter check. If 'Yes' for 'Is CANINTF.RX1IF = 0?', it moves the message to RXB1, sets CANINTF.RX1IF = 1, sets RXB1CTRL.FILHIT, and checks 'Is CANINTE.RX1IE = 1?'. If 'No', it loops back to the RXB1 filter check. If 'Yes', it generates an interrupt on INT, sets CANSTAT, and checks for pin configuration (RXBF1 or RXBF0). If 'No' for either, it loops back to the RXB1 filter check. If 'Yes' for either, it loops back to the RXB0 filter check. If 'Yes' for 'Is CANINTF.RX0IF = 0?', it moves the message to RXB0, sets CANINTF.RX0IF = 1, sets RXB0CTRL.FILHIT, and checks 'Is CANINTE.RX0IE = 1?'. If 'No', it loops back to the RXB0 filter check. If 'Yes', it generates an interrupt on INT, sets CANSTAT, and checks for pin configuration (RXBF1 or RXBF0). If 'No' for either, it loops back to the RXB0 filter check. If 'Yes' for either, it loops back to the RXB0 filter check.

U-0	R/W-0	R/W-0	U-0	R-0	R/W-0	R-0	R-0
—	RXM1	RXM0	—	RXRTR	BUKT	BUKT1	FILHIT0
bit 7				bit 0			

R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared
		x = Bit is unknown

비트 7 비 구현 : '0'

비트 6-5 RXM : 수신 버퍼 작동 모드 비트

11 = 마스크 / 필터를 끕니다. 어떤 메시지를 받는다.

10 = 필터 조건을 충족시키는 확장 식별자가있는 유효한 메시지 만 수신

01 = 필터 기준을 충족하는 표준 식별자로 유효한 메시지 만 수신

00 = 필터 조건을 충족하는 표준 또는 확장 식별자를 사용하여 모든 유효한 메시지를 수신합니다.

비트 4 비 구현 : '0'

비트 3 RXRTR : 수신 된 원격 전송 요청 비트

1 = 원격 전송 요청 수신 됨

0 = 수신 된 원격 전송 요청 없음

비트 2 BUKT : 롤오버 활성화 비트

1 = RXB0이 가득 차면 RXB0 메시지가 롤오버되고 RXB1에 기록됩니다.

0 = 롤오버 사용 중지됨

비트 1 BUKT1 : BUKT 비트의 읽기 전용 복사본 (MCP2515에 의해 내부적으로 사용됨)

bit 0 FILHIT : 필터 히트 비트 - 메시지를 수신 할 수있는 수용 필터를 나타냅니다.

1 = 수락 필터 1 (RXF1)

0 = 수락 필터 0 (RXF0)

참고 : RXB0에서 RXB1 로의 롤오버가 발생하면 FILHIT 비트는 뒤집힌 메시지를 수락 한 필터를 반영합니다.

REGISTER 4-3: BFPCTRL – RXnBF PIN CONTROL AND STATUS
(ADDRESS: 0Ch)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	B1BFS	B0BFS	B1BFE	B0BFE	B1BFM	B0BFM
bit 7		bit 0					

**REGISTER 4-4: RXBnSIDH – RECEIVE BUFFER n STANDARD IDENTIFIER HIGH
(ADDRESS: 61h, 71h)**

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-0 SID : 표준 식별자 비트 Standard Identifier bits <10 : 3>

이 비트는 수신 된 메시지에 대한 표준 식별자의 8 개의 최상위 비트를 포함한다.

**REGISTER 4-5: RXBnSIDL – RECEIVE BUFFER n STANDARD IDENTIFIER LOW
(ADDRESS: 62h, 72h)**

R-x	R-x	R-x	R-x	R-x	U-0	R-x	R-x
SID2	SID1	SID0	SRR	IDE	—	EID17	EID16
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-5 SID : 표준 식별자 비트 <2 : 0> 이 비트는 수신 된 메시지에 대한 표준 식별자의 세 개의 최하위 비트를 포함합니다

비트 4 SRR : 표준 프레임 원격 전송 요청 비트 (IDE 비트 = '0'인 경우에만 유효)

1 = 표준 프레임 원격 전송 요청 수신 됨

0 = 수신 된 표준 데이터 프레임

비트 3 IDE : 확장 식별자 플래그 비트

이 비트는 수신 된 메시지가 표준 또는 확장 프레임인지 여부를 나타냅니다.

1 = 수신 된 메시지가 확장 프레임이었습니다.

0 = 받은 메시지가 표준 프레임이었습니다.

비트 2 비 구현 : '0'

비트 1-0 EID : 확장 식별자 비트 <17:16>

이 비트는 수신 된 메시지에 대한 확장 식별자의 최상위 2 비트를 포함한다.

**REGISTER 4-6: RXBnEID8 – RECEIVE BUFFER n EXTENDED IDENTIFIER HIGH
(ADDRESS: 63h, 73h)**

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

비트 7-0 EID : 확장 식별자 비트 <15 : 8>

이 비트들은 수신 된 메시지에 대한 확장 식별자의 비트 15-8을 유지한다.

**REGISTER 4-7: RXBnEID0 – RECEIVE BUFFER n EXTENDED IDENTIFIER LOW
(ADDRESS: 64h, 74h)**

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

비트 7-0 EID : 확장 식별자 비트 <7 : 0>

이 비트들은 수신 된 메시지에 대한 확장 식별자의 최하위 8 비트를 보유한다.

**REGISTER 4-8: RXBnDLC – RECEIVE BUFFER n DATA LENGHT CODE
(ADDRESS: 65h, 75h)**

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

비트 7 비 구현 : '0'

비트 6 RTR : 확장 프레임 원격 전송 요청 비트 (RXBnSIDL.IDE = '1'인 경우에만 유효)

1 = 확장 프레임 원격 전송 요청 수신 됨

0 = 수신 된 확장 데이터 프레임

비트 5 RB1 : 예약 비트 1

비트 4 RB0 : 예약 비트 0

비트 3-0 DLC : 데이터 길이 코드 비트 <3 : 0>

수신 된 데이터 바이트 수를 나타냅니다.

REGISTER 4-9: RXBnDM – RECEIVE BUFFER n DATA BYTE M
(ADDRESS: 66h - 6Dh, 76h - 7Dh)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
RBnDm7	RBnDm6	RBnDm5	RBnDm4	RBnDm3	RBnDm2	RBnDm1	RBnDm0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

비트 7-0 RBnDm7 : RBnDm0 : 수신 버퍼 n 데이터 필드 바이트 m 수신 된 메시지의 데이터 바이트를 포함하는 8 바이트

4.5 메시지 허용 필터 및 마스크

메시지 수용 필터 및 마스크는 메시지 어셈블리 버퍼의 메시지를 수신 버퍼 중 하나에 로드해야 하는지 확인하십시오 (그림 4-5 참조).

유효한 메시지가 MAB로 수신되면 메시지의 식별자 필드가 필터 값과 비교됩니다.

일치하는 것이 있으면 해당 메시지가 적절한 수신 버퍼에 로드됩니다.

4.5.1 데이터 바이트 필터링

표준 데이터 프레임 (11 비트 식별자)을 수신 할 때, DSTEMP는 일반적으로 16 비트의 마스크와 확장 식별자와 연관된 필터를 데이터 필드의 처음 16 비트 (데이터 바이트 0 및 1)에 자동으로 적용합니다.

그림 4-4는 확장 된 데이터 프레임과 표준 데이터 프레임에 마스크와 필터를 적용하는 방법을 보여줍니다.

데이터 바이트 필터링은 첫 번째 데이터 바이트 (예 : DeviceNet TM)에서 필터링하는 상위 계층 프로토콜 (HLP)을 구현할 때 MCU의 로드를 줄입니다.

4.5.2 필터 매칭

필터 마스크 (레지스터 4-14 ~ 레지스터 4-17 참조)는 식별자에서 필터로 검사되는 비트를 결정하는 데 사용됩니다.

ID의 각 비트를 마스크 및 필터와 비교하여 메시지를 수신 버퍼에 로드해야 하는지 여부를 결정하는 진실성 테이블이 표 4-2에 나와 있습니다.

마스크는 수용 필터를 적용 할 비트를 본질적으로 결정합니다.

마스크 비트가 0으로 설정되면 필터 비트에 관계없이 해당 비트가 자동으로 수락됩니다.

TABLE 4-2: FILTER/MASK TRUTH TABLE

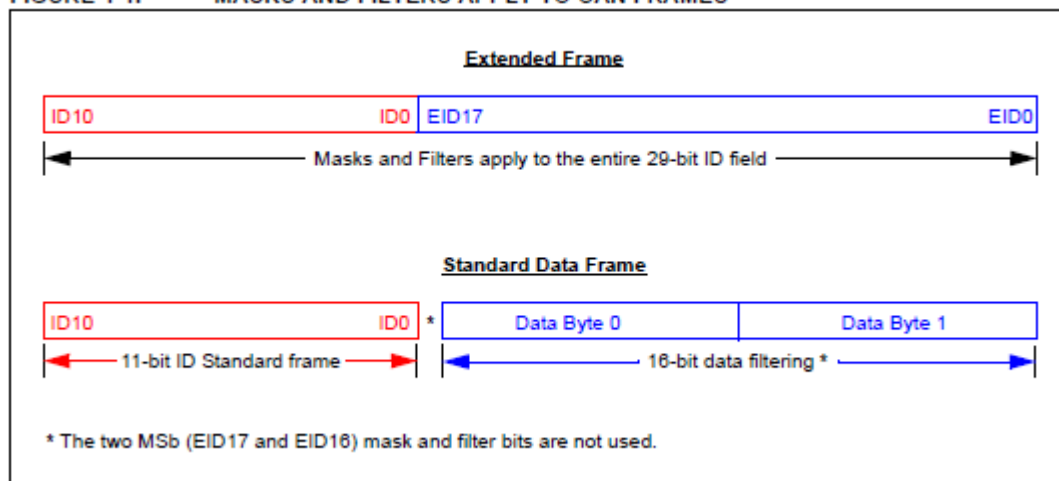
Mask Bit n	Filter Bit n	Message Identifier bit	Accept or Reject bit n
0	X	X	Accept
1	0	0	Accept
1	0	1	Reject
1	1	0	Reject
1	1	1	Accept

Note: X = don't care

수신 버퍼 블록 다이어그램 (그림 4-2)에 표시된 것처럼 수용 필터 RXF0 및 RXF1 (및 필터 마스크 RXM0)은 RXB0과 관련됩니다.

필터 RXF2, RXF3, RXF4, RXF5 및 마스크 RXM1은 RXB1과 연관됩니다.

FIGURE 4-4: MASKS AND FILTERS APPLY TO CAN FRAMES



4.5.3 FILHIT BITS

수신 된 메시지의 필터 일치는 연관된 RXBnCTRL 레지스터의 FILHIT 비트에 의해 결정될 수 있습니다.

버퍼 0의 경우 RXB0CTRL.FILHIT0, 버퍼 1의 경우 RXB1CTRL.FILHIT <2 : 0>.

수신 버퍼 1 (RXB1)에 대한 3 개의 FILHIT 비트는

다음과 같이 코딩 :

- 101 = 수락 필터 5 (RXF5)
- 100 = 수락 필터 4 (RXF4)
- 011 = 수락 필터 3 (RXF3)
- 010 = 수락 필터 2 (RXF2)
- 001 = 수락 필터 1 (RXF1)
- 000 = 수락 필터 0 (RXF0)

NOTE : 000 및 001은 RXB0CTRL의 BUKT 비트가 설정되어 RXB0 메시지가 RXB1로 롤오프할 수 있는 경우에만 발생할 수 있습니다.

RXB0CTRL은 BUKT 비트와 FILHIT <0> 비트의 두 복사본을 포함합니다.

BUKT 비트의 코딩은이 세 비트가

RXB1CTRL.FILHIT 비트와 비슷하게 사용할 수 있습니다.

RXB0 또는 RXB1로 롤 오버 한 후 필터 RXF0 및 RXF1의 히트를 구별합니다.

- 111 = 수락 필터 1 (RXB1)
- 110 = 수락 필터 0 (RXB1)
- 001 = 수락 필터 1 (RXB0)
- 000 = 수락 필터 0 (RXB0)

BUKT 비트가 클리어 인 경우, 6 개의 필터에 대응하는 6 개의 코드가있다.

BUKT 비트가 설정되면 여섯 개의 필터에 해당하는 여섯 개의 코드와 RXB1로 롤오버하는 RXF0 및 RXF1 필터에 해당하는 두 개의 추가 코드가 있습니다.

4.5.4 다중 필터 일치

둘 이상의 수락 필터가 일치하면 FILHIT 비트는 일치하는 가장 낮은 번호의 필터의 이진 값을 인코딩합니다.

예를 들어 필터 RXF2와 필터 RXF4가 일치하면 FILHIT에 RXF2 값이로드됩니다.

이것은 우선 순위가 더 낮은 번호가 매겨진 필터로 수용 필터의 우선 순위를 결정합니다.

메시지는 필터 번호의 오름차순으로 필터와 비교됩니다.

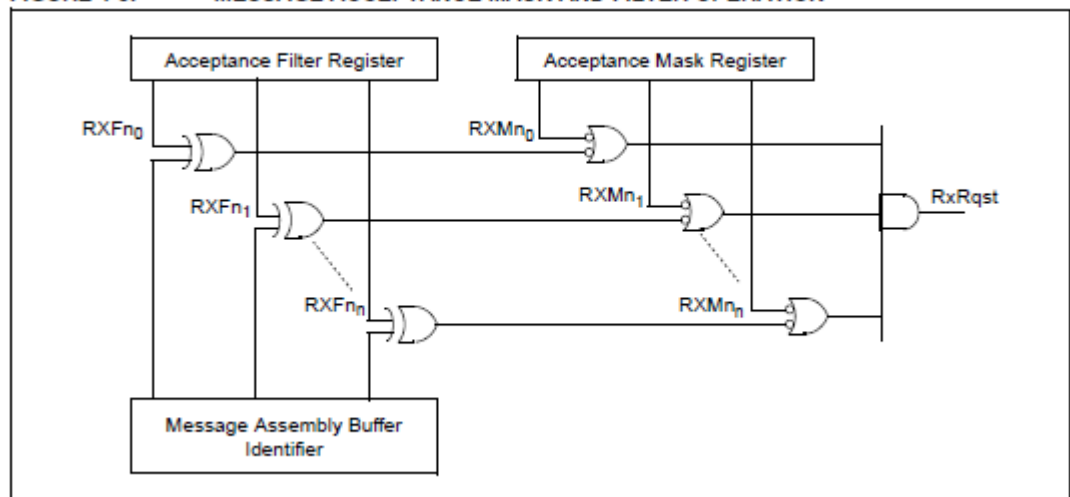
이것은 또한 메시지가 하나의 버퍼로만 수신되도록합니다.

이는 RXB0이 RXB1보다 우선 순위가 높음을 의미합니다.

4.5.5 마스크 및 필터 구성

마스크 및 필터 레지스터는 MCP2515가 구성 모드에있을 때만 수정할 수 있습니다 (섹션 10.0 "작동 모드"참조).

FIGURE 4-5: MESSAGE ACCEPTANCE MASK AND FILTER OPERATION



REGISTER 4-10: RXFnSIDH – FILTER n STANDARD IDENTIFIER HIGH
(ADDRESS: 00h, 04h, 08h, 10h, 14h, 18h)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 7							
bit 0							

Legend:							
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'					
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared		x = Bit is unknown			

비트 7-0 SID : 표준 식별자 필터 비트 <10 : 3>

이 비트는 수신 된 메시지의 표준 식별자 부분의 비트 <10 : 3>에 적용될 필터 비트를
보유합니다

REGISTER 4-11: RXFnSIDL – FILTER n STANDARD IDENTIFIER LOW
(ADDRESS: 01h, 05h, 09h, 11h, 15h, 19h)

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-5 SID : 표준 식별자 필터 비트 <2 : 0>

이들 비트는 수신 된 메시지의 표준 식별자 부분의 비트 <2 : 0>에 적용될 필터 비트를
보유한다.

비트 4 비 구현 : '0'

비트 3 EXIDE : 확장 식별자 가능 비트

1 = 필터는 확장 된 프레임에만 적용됩니다.

0 = 필터는 표준 프레임에만 적용됩니다.

비트 2 비 구현 : '0'

비트 1-0 EID : 확장 식별자 필터 비트 <17:16>

이 비트들은 수신 된 메시지의 확장 식별자 부분의 비트들 <17:16>에 적용될 필터 비트들을
보유한다

REGISTER 4-12: RXFnEID8 – FILTER n EXTENDED IDENTIFIER HIGH
(ADDRESS: 02h, 06h, 0Ah, 12h, 16h, 1Ah)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-0 EID : 확장 식별자 비트 <15 : 8>

이 비트는 수신 된 메시지의 확장 식별자 부분의 비트 <15 : 8>에 적용될 필터 비트를
보유합니다

REGISTER 4-13: RXFnEID0 – FILTER n EXTENDED IDENTIFIER LOW
(ADDRESS: 03h, 07h, 0Bh, 13h, 17h, 1Bh)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared

x = Bit is unknown

비트 7-0 EID : 확장 식별자 비트 <7 : 0>

이 비트는 수신 된 메시지의 확장 식별자 부분의 비트 <7 : 0>에 적용될 필터 비트를
보유합니다

REGISTER 4-14: RXMnSIDH – MASK n STANDARD IDENTIFIER HIGH
(ADDRESS: 20h, 24h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared

x = Bit is unknown

비트 7-0 SID : 표준 식별자 마스크 비트 <10 : 3>

이 비트는 수신 된 메시지의 표준 식별자 부분의 비트 <10 : 3>에 적용될 마스크 비트를
보유합니다

REGISTER 4-15: RXMnSIDL – MASK n STANDARD IDENTIFIER LOW
(ADDRESS: 21h, 25h)

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
SID2	SID1	SID0	—	—	—	EID17	EID16
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared

x = Bit is unknown

비트 7-5 SID : 표준 식별자 마스크 비트 <2 : 0>

이들 비트는 수신 된 메시지의 표준 식별자 부분의 비트 <2 : 0>에 적용될 마스크 비트를
보유합니다

비트 4-2 비 구현 : '0'

비트 1-0 EID : 확장 식별자 마스크 비트 <17:16>

이 비트는 수신 된 메시지의 확장 식별자 부분의 비트 <17:16>에 적용될 마스크 비트를
보유합니다

**REGISTER 4-16: RXMnEID8 – MASK n EXTENDED IDENTIFIER HIGH
(ADDRESS: 22h, 26h)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-0 EID : 확장 식별자 비트 <15 : 8>

이 비트는 수신 된 메시지의 확장 식별자 부분의 비트 <15 : 8>에 적용될 필터 비트를
보유합니다

**REGISTER 4-17: RXMnEID0 – MASK n EXTENDED IDENTIFIER LOW
(ADDRESS: 23h, 27h)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-0 EID : 확장 식별자 마스크 비트 <7 : 0>

이 비트는 수신 된 메시지의 확장 식별자 부분의 비트 <7 : 0>에 적용될 마스크 비트를
보유합니다

5.0 비트 타이밍

주어진 CAN 버스의 모든 노드는 동일한 공칭 비트 전송률을 가져야합니다.

CAN 프로토콜은 NRZ (Non Return to Zero) 코딩을 사용하며 데이터 스트림 내에서 클럭을
인코딩하지 않습니다.

따라서, 수신 클럭은 수신 노드에 의해 복구되고 송신기의 클럭에 동기화되어야한다.

발진기와 전송 시간은 노드마다 다를 수 있으므로 수신기는 수신기 클록을 동기화하고 유지하기 위해 데이터 전송 에지에 동기화 된 일부 유형의 PLL (Phase Lock Loop)이 있어야합니다.

데이터는 NRZ로 코딩되었으므로 DPLL (Digital Phase Lock Loop) 동기화를 유지하기 위해 적어도 6 비트 씩 에지가 발생하도록 비트 스템핑을 포함시켜야합니다.

MCP2515의 비트 타이밍은 들어오는 데이터와 동기화되고 전송 된 데이터의 공칭 타이밍을 제공하도록 구성된 aDPLL을 사용하여 구현됩니다.

DPLL은 각 비트 시간을 Time Quanta (TQ)라고하는 최소 기간으로 구성된 여러 세그먼트로 나눕니다.

비트 시간 프레임 내에서 실행되는 버스 타이밍 기능(예 : 로컬 오실레이터와의 동기화, 네트워크 전송 지연 보상 및 샘플 포인트 위치 지정)은 DPLL의 프로그래밍 가능한 비트 타이밍 로직에 의해 정의됩니다.

5.1 CAN 비트 시간

CAN 버스의 모든 장치는 동일한 비트 전송률을 사용해야합니다.

그러나 모든 장치에는 샘 마스터 오실레이터 클럭 주파수가 필요하지 않습니다.

개별 디바이스의 서로 다른 클럭 주파수의 경우, 각 세그먼트의 전송률 프리스케일러 및 시간 쿼텀 수를 적절하게 설정하여 비트 전송률을 조정해야합니다.

CAN 비트 시간은 겹치지 않는 세그먼트로 구성됩니다.

각 세그먼트는이 데이터 시트의 뒷부분에서 설명 할 Time Quanta (TQ)라고하는 정수 단위로 구성됩니다.

NBR (nominal bit rate)은 CAN 사양에서 재 동기화가없는 이상적인 송신기가 전송 한 초당 비트 수로 정의됩니다.

다음 방정식으로 설명 할 수 있습니다.

EQUATION 5-1:

$$NBR = f_{bit} = \frac{1}{t_{bit}}$$

공칭 비트 시간

NBT (Nominal Bit Time) (t 비트)는 겹치지 않는 세그먼트로 구성됩니다 (그림 5-1).

따라서 NBT는 다음 세그먼트의 합계입니다.

$$t_{bit} = t_{SyncSeg} + t_{PropSeg} + t_{PS1} + t_{PS2}$$

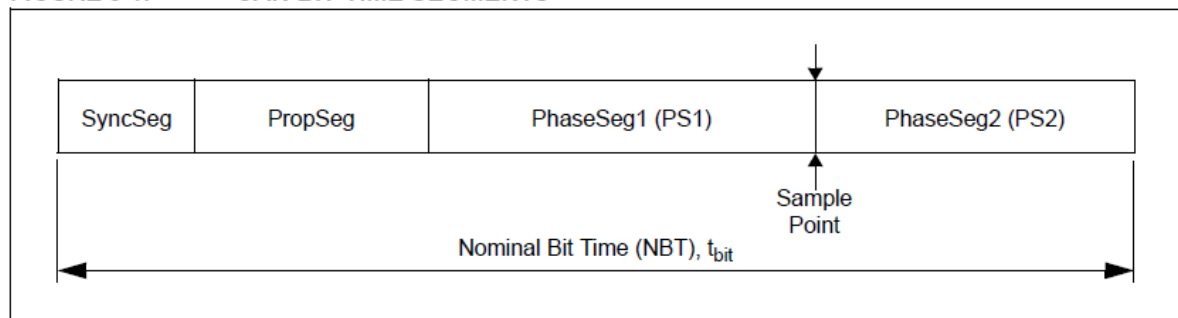
NBT와 관련하여 샘플 포인트, 동기화 점프 폭 (SJW) 및 정보 처리 시간 (IPT)은 나중에 설명합니다.

동기 구분 (SYNCHRONIZATION SEGMENT)

동기화 세그먼트 (SyncSeg)는 NBT의 첫 번째 세그먼트이며 버스의 노드를 동기화하는 데 사용됩니다.

비트 가장자리는 SyncSeg 내에서 발생할 것으로 예상됩니다. 이 세그먼트는 1 TQ로 고정됩니다.

FIGURE 5-1: CAN BIT TIME SEGMENTS



[선전 부분]

전파 세그먼트 (PropSeg)는 노드 간의 물리적 지연을 보상하기 위해 존재합니다.

전파 지연은 다음의 합계의 두 배로 정의됩니다.

신호는 버스 드라이버와 연관된 지연을 포함하여 버스 라인에서 전파됩니다.

PropSeg는 1 - 8 TQ에서 프로그래밍 할 수 있습니다.

위상 세그먼트 1 (PS1) 및 위상 세그먼트 2 (PS2)

2 개의 위상 세그먼트, PS1 및 PS2는 버스의 에지 위상 오류를 보상하는 데 사용됩니다.

PS1은 resynchronization에 의해 길어질 수 있습니다 (또는 PS2 단축됨).

PS1은 1 - 8 TQ에서 프로그래밍 할 수 있으며 PS2는 2 - 8 TQ에서 프로그래밍 할 수 있습니다.

[견본 포인트]

샘플 포인트는 비트 시간에서 논리 레벨을 읽고 해석합니다.

샘플 포인트는 PS1의 끝에 있습니다.

이 규칙의 예외는 샘플 모드가 비트 당 세 번 샘플링하도록 구성된 경우입니다.

이 경우, 비트가 PS1의 끝에서 여전히 샘플링되는 동안, 두 개의 추가 샘플이 PS1의 종료 이전에 한 번의 TQ 간격에서 취해지고, 비트의 값은 다수결에 의해 결정된다.

[정보 처리 시간]

정보 처리 시간 (IPT)은 로직이 샘플링 된 비트의 비트 레벨을 결정하는 데 필요한 시간입니다.

IPT는 샘플 포인트에서 시작하여 TQ로 측정되며 마이크로 칩 CAN 모듈의 경우 2 TQ로 고정됩니다.

PS2도 샘플 포인트에서 시작하고 비트 시간의 마지막 세그먼트이기 때문에 PS2 최소값이 IPT보다 작지 않아야 합니다.

Therefore:

$$PS2_{min} = IPT = 2TQ$$

[동시 점프 너비 (SYNCHRONIZATION JUMP WIDTH)]

SJW (Synchronization Jump Width)는 전송된 메시지와 동기화를 유지하기 위해 필요한대로 1 - 4 TQ (구성된대로)로 비트 클럭을 조정합니다.

동기화에 대한 자세한 내용은 이 데이터 시트 뒷부분에 나와 있습니다.

[시간 양자]

비트 시간을 구성하는 각 세그먼트는 TQ (Time Quanta)라는 정수 단위로 구성됩니다.

각 Time Quantum의 길이는 발진기주기 (tOSC)를 기반으로 합니다.

기본 TQ는 발진기주기의 두 배입니다.

그림 5-2는 비트주기가 TOSC 및 TQ에서 파생되는 방법을 보여줍니다.

TQ 길이는 하나의 TQ 클럭주기 (tBRPCLK)와 같으며, 이는 프로그래밍 가능 프리스케일러 (Baud Rate Prescaler, BRP)를 사용하여 프로그래밍 할 수 있습니다.

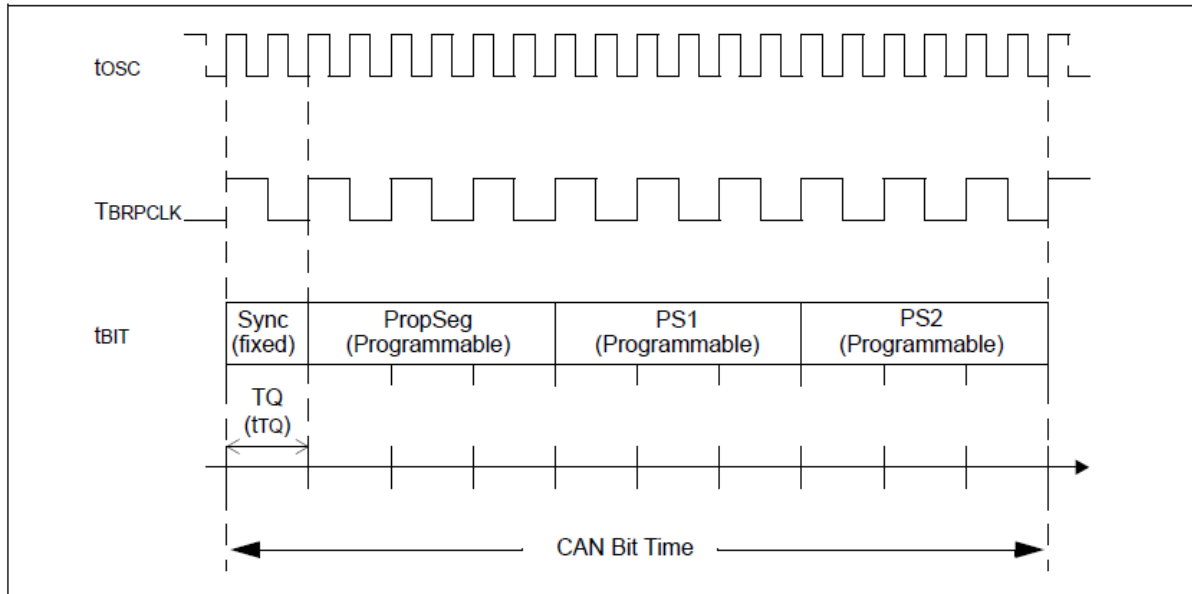
이것은 다음 방정식에 설명되어 있습니다.

EQUATION 5-2:

$$TQ = 2 \cdot BRP \cdot T_{OSC} = \frac{2 \cdot BRP}{F_{OSC}}$$

Where: BRP equals the configuration as shown in Register 5-1.

FIGURE 5-2: TQ AND THE BIT PERIOD



5.2 동기화

오실레이터 (oscillator) 간의 위상 변이를 보상하기 위해 각 CAN 컨트롤러는 수신 신호의 관련 신호 에지에 동기화 할 수 있어야합니다.

동기화는 DPLL 기능이 구현되는 프로세스입니다.

전송 된 데이터의 에지가 감지되면 로직은 에지의 위치를 예상 시간 (SyncSeg)과 비교합니다.

그러면 회로가 필요에 따라 PS1 및 PS2의 값을 조정합니다.

동기화에는 두 가지 메커니즘이 사용됩니다.

1. 하드 동기화.
2. 재 동기화.

5.2.1 하드 동시성

하드 동기화는 BUS IDLE 상태에서 메시지의 시작을 나타내는 과도한 에지가있을 때만 수행됩니다.

하드 동기화 후에 비트 시간 카운터가 SyncSeg로 다시 시작됩니다.

하드 동기화는 발생 된 에지를 재시작 된 비트 시간의 동기화 세그먼트 내에 놓도록 강제합니다.

동기화 규칙으로 인해 하드 동기화가 발생하면 해당 비트 시간 내에 재 동기화가 수행되지 않습니다.

5.2.2 재 동기

재 동기화의 결과로 PS1이 길어 지거나 PS2가 짧아 질 수 있습니다.

위상 버퍼 세그먼트의 길이를 줄이거나 늘이는 양은 동기화 점프 폭 (SJW)에 의해 주어진 상한선을가집니다.

SJW의 값은 PS1에 추가되거나 PS2에서 제거됩니다 (그림 5-3 참조).

SJW는 DPLL의 루프 필터링을 나타냅니다.

SJW는 1 TQ와 4 TQ 사이에서 프로그래밍 할 수 있습니다.

5.2.2.1 위상 오차

NRZ 비트 코딩 방법은 클럭을 인코딩하지 않습니다.
메시지에.

클럭킹 정보는 열성 - 우성 전환에서만 파생됩니다.

고정 된 최대 연속 비트 수만 같은 값 (비트 스테핑)
프레임 동안 비트 스트림에 대한 재 동기화를 보장한다.

에지의 위상 오차는 TQ로 측정 된 SyncSeg에 대한 에지의 위치에 의해 주어집니다.

위상 오차는 다음과 같이 TQ의 크기로 정의됩니다.

- 가장자리가 SYNCSEG 내에 있으면 $e = 0$ 입니다.
- 가장자리가 SAMPLE POINT보다 앞에 있으면 $e > 0$ 입니다.

(TQ가 PS1에 추가됨).

- 가장자리가 SAMPLE POINT 다음에 오면 $e < 0$

이전 비트 (TQ는 PS2에서 빼기).

5.2.2.2 위상 오차 없음 ($e = 0$)

위상 오차의 크기가 작거나 같으면

SJW의 프로그래밍 된 값에 대해 재 동기화의 효과는 하드 동기화와 동일합니다.

5.2.2.3 양의 위상 오차 ($e > 0$)

위상 오차의 크기가 SJW보다 크고, 위상 오차가 양이면, PS1은 SJW와 동일한 양만큼 길어진다.

5.2.2.4 음의 위상 오차 ($e < 0$)

위상 오차의 크기가 재 동기화 점프 폭보다 크고 위상 오차가 음수이면 PS2는 SJW와 동일한 양만큼 짧아집니다.

5.2.3 동기 부여 규칙

1. recessive-to-dominant 에지 만 동기화에 사용됩니다.

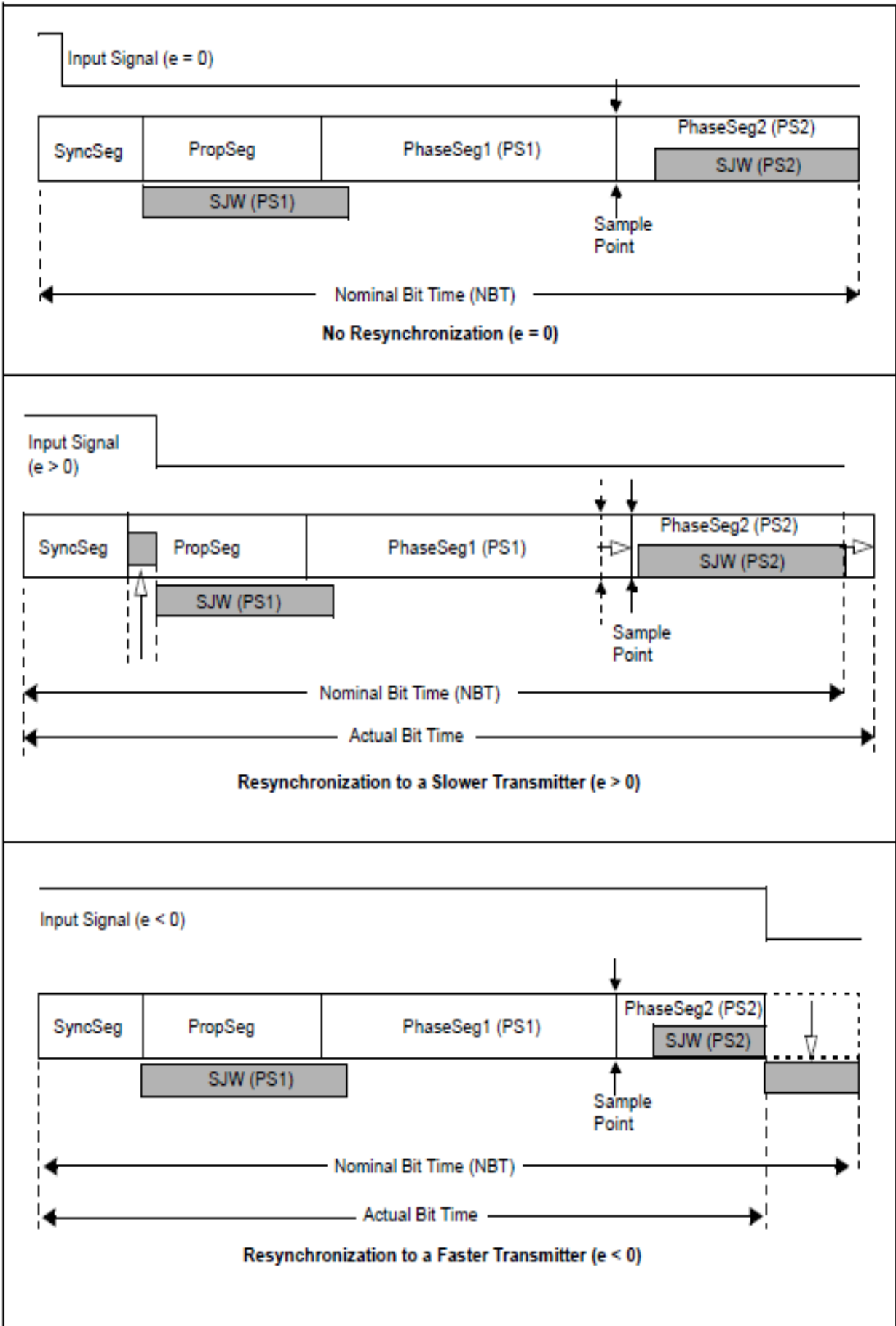
2. 1 비트 시간 내의 하나의 동기화 만이 허용.

3. 에지는 이전 샘플 포인트 (이전에 읽은 버스 값)에서 감지 된 값이 에지 직후의 버스 값과 다른 경우에만 동기화에 사용됩니다.

4. 송신 노드는 양의 위상 오차 ($e > 0$)로 재 동기화되지 않습니다.

5. 위상 오차의 절대 크기가 SJW보다 큰 경우 적절한 위상 세그먼트가 SJW와 동일한 양만큼 조정됩니다.

FIGURE 5-3: SYNCHRONIZING THE BIT TIME



5.3 시간 세그먼트 프로그래밍

시간 프로그래밍에 대한 몇 가지 요구 사항

세그먼트 :

- $\text{PropSeg} + \text{PS1} > \text{PS2}$
- $\text{PropSeg} + \text{PS1} > \text{TDELAY}$
- $\text{PS2} > \text{SJW}$

예를 들어, 125 kHz CAN 보오율

FOSC = 20 MHz가 요구되는 경우 :

$\text{TOSC} = 50\text{ns}$, $\text{BRP} < 5 : 0 > = 04\text{h}$, $\text{TQ} = 500\text{ns}$ 를 선택하십시오. 125 kHz를 얻으려면 비트 시간이 16 TQ 여야합니다.

일반적으로 비트 샘플링은 시스템 매개 변수에 따라 비트 시간의 약 60-70 %에서 수행되어야 합니다.

또한 일반적으로 TDELAY는 1-2TQ입니다.

$\text{SyncSeg} = 1 \text{ TQ}$ 및 $\text{PropSeg} = 2 \text{ TQ}$. 따라서 $\text{PS1} = 7 \text{ TQ}$ 로 설정하면 전환 후 샘플이 10 TQ에 배치됩니다.

이것은 PS2에 대해 6 TQ를 남겨 둡니다.

규칙에 따르면 PS2가 6이므로 SJW는 최대 4 TQ 일 수 있습니다.

그러나 대형 SJW는 일반적으로 세라믹 공진기 사용과 같이 서로 다른 노드의 클럭 생성이 부정확하거나 불안정한 경우에만 필요합니다.

따라서 SJW가 1이면 일반적으로 충분합니다.

5.4 발진기 허용 오차

비트 타이밍 요구 사항은 경험적으로 최대 125 kbit / 초의 전송 속도를 갖는 애플리케이션에 세라믹 공진기를 사용할 수 있도록 해줍니다.

CAN 프로토콜의 전체 버스 속도 범위의 경우, 수정 발진기가 필요합니다.

1.7 %의 최대 노드 간 발진기 변동은 허용됩니다.

5.5 비트 타이밍 구성 레지스터

구성 레지스터 (CNF1, CNF2, CNF3)는 CAN 버스 인터페이스의 비트 타이밍을 제어합니다.

이 레지스터는 MCP2515가 설정 모드에있을 때만 수정할 수 있습니다 (10.0 절 "동작 모드"참조).

5.5.1 CNF1

BRP <5 : 0> 비트는 보드 율 프리스케일러를 제어합니다.

이 비트는 최소 TQ 길이가 2 TOSC (BRP <5 : 0> = 'b000000'인 경우) 인 OSC1 입력 주파수에 상대적인 TQ 길이를 설정합니다.

SJW <1 : 0> 비트는 TQ 수의 관점에서 SJW를 선택합니다

5.5.2 CNF2

PRSEG <2 : 0> 비트는 전파 세그먼트의 길이를 설정합니다 (TQ 단위).

PHSEG1 <2 : 0> 비트는 PS1의 길이 (TQ 단위)를 설정합니다.

SAM 비트는 RXCAN 핀이 샘플링되는 횟수를 제어합니다.

이 비트를 '1'로 설정하면 버스는 샘플 포인트 이전의 TQ / 2에서 두 번, 그리고 정상 샘플 포인트 (PS1의 끝 부분에 있음)에서 세 번 샘플링됩니다.

버스의 값은 다수 표본으로 결정됩니다.

SAM 비트가 '0'으로 설정되면 RXCAN 핀은 샘플 포인트에서 한 번만 샘플링됩니다.

BTLMODE 비트는 PS2의 길이를 결정하는 방법을 제어합니다.

이 비트가 '1'로 설정되면 PS2의 길이는 CNF3의 PHSEG2 <2 : 0> 비트에 의해 결정됩니다 (5.5.3 "CNF3"참조). BTLMODE 비트가 '0'으로 설정된 경우 PS2의 길이는 PS1의 길이와 정보 처리 시간 (MCP2515의 경우 2TQ로 고정됩니다)보다 큼니다.

5.5.3 CNF3

PHSEG2 <2 : 0> 비트는 CNF2.BTLMODE 비트가 '1'로 설정된 경우 PS2의 길이를 설정합니다 (TQ 단위).

BTLMODE 비트가 '0'으로 설정되면, PHSEG2 <2 : 0> 비트 효과가 없다.

REGISTER 5-1: CNF1 – CONFIGURATION 1 (ADDRESS: 2Ah)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
bit 7 bit 0							

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-6 SJW : 동기 점프 폭 길이 비트 <1 : 0>

11 = 길이 = 4 × TQ

10 = 길이 = 3 × TQ

01 = 길이 = 2 × TQ

00 = 길이 = 1 × TQ

비트 5-0 BRP : 보드 윌 프리 스케일러 비트 <5 : 0>

$TQ = 2 \times (BRP + 1) / FOSC$

REGISTER 5-2: CNF2 – CONFIGURATION 1 (ADDRESS: 29h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BTLMODE	SAM	PHSEG12	PHSEG11	PHSEG10	PRSEG2	PRSEG1	PRSEG0
bit 7 bit 0							

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7 BTLMODE : PS2 비트 시간 길이 비트

1 = PHSEG22에 의해 결정된 PS2의 길이 : CNF3의 PHSEG20 비트

0 = PS2의 길이가 PS1 및 IPT (2TQ)보다 크다.

비트 6 SAM : 샘플 포인트 구성 비트

1 = 버스 라인은 샘플 포인트에서 3 번 샘플링됩니다.

0 = 버스 라인이 샘플 포인트에서 한 번 샘플링 됨

비트 5-3 PHSEG1 : PS1 길이 비트 <2 : 0>

$(PHSEG1 + 1) \times TQ$

비트 2-0 PRSEG : 전달 세그먼트 길이 비트 <2 : 0>

$(PRSEG + 1) \times TQ$

REGISTER 5-3: CNF3 - CONFIGURATION 1 (ADDRESS: 28h)

R/W-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
SOF	WAKFIL	—	—	—	PHSEG22	PHSEG21	PHSEG20
bit 7					bit 0		

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7 SOF : Start-of-Frame 신호 비트

CANCTRL.CLKEN = 1 인 경우 :

1 = SOF 신호에 대해 CLKOUT 핀이 활성화 됨

0 = 클럭 아웃 기능을 위해 CLKOUT 핀이 활성화 됨

CANCTRL.CLKEN = 0이면 Bit는 상관하지 않습니다.

비트 6 WAKFIL : 웨이크 업 필터 비트

1 = 웨이크 업 필터 사용

0 = 웨이크 업 필터가 비활성화 됨

비트 5-3 비 실행 : '0'

비트 2-0 PHSEG2 : PS2 길이 비트 <2 : 0>

$(PHSEG2 + 1) \times TQ$

PS2의 최소 유효 설정은 2 TQ입니다.

6.0 오류 탐지

CAN 프로토콜은 정교한 오류 탐지 메커니즘을 제공합니다.

다음 오류를 감지 할 수 있습니다.

6.1 CRC 오류

Cyclic Redundancy Check (CRC)를 사용하여 송신기는 프레임 시작부터 데이터 필드의 끝까지 비트 시퀀스에 대한 특수 검사 비트를 계산합니다.

이 CRC 시퀀스는 CRC 필드에서 전송됩니다.

수신 노드는 또한 동일한 공식을 사용하여 CRC 시퀀스를 계산하고 수신 된 시퀀스에 대한 비교를 수행한다.

불일치가 발견되면 CRC 오류가 발생하고 오류 프레임이 생성됩니다.

메시지가 반복됩니다.

6.2 응답 오류

메시지의 acknowledge 필드에서 송신기는 확인 비트 (열성 비트로 전송 된)에 도미넌트 비트가 포함되어 있는지 확인합니다.

그렇지 않은 경우 다른 노드가 프레임을 올바르게 수신하지 못합니다.

확인 오류가 발생했습니다. 오류 프레임이 생성되고 메시지를 반복해야 합니다.

6.3 서식 오류

노드가 네 개의 세그먼트 (프레임 끝, 프레임 간 공간, 승인 구분 기호 또는 CRC 구분 기호 포함) 중 하나에서 도미넌트 비트를 감지하면 양식 오류가 발생하고 오류 프레임이 생성됩니다.

메시지가 반복됩니다.

6.4 비트 오류

송신기가 반대편을 감지하면 비트 오류가 발생합니다.

(즉, 지배적 인 것을 전송하고 열성을 검출하거나, 열성을 전달하고 지배적 인 것을 검출)까지 전송할 수 있다.

예외 : 송신기가 열성 비트를 전송하고 중재 필드와 확인 슬롯 중에 도미넌트 비트가 감지되는 경우 비트 오류가 발생하지 않습니다.

정상적인 중재가 일어나기 때문에 생성됩니다.

6.5 물건 오류

프레임 시작과 CRC 분리 문자 사이에 동일한 극성을 가진 6 개의 연속 비트가 감지되면 비트 스테핑 규칙이 위반 된 것입니다.

물건 오류가 발생하고 오류 프레임이 생성됩니다.

메시지가 반복됩니다.

6.6 오류 상태

감지 된 오류는 오류 프레임을 통해 다른 모든 노드에 알려집니다.

오류 메시지의 전송이 중단되고 프레임이 가능한 한 빨리 반복됩니다.

또한 각 CAN 노드는 내부 오류 카운터의 값에 따라 세 가지 오류 상태 중 하나에 있습니다.

1. 오류 활성.
2. 오류 - 수동.
3. 버스 - 오프 (송신기 전용).

오류 활성 상태는 nodecan이 아무런 제한없이 메시지 (능동적 인 비트로 만들어진)와 오류 메시지를 전송하는 일반적인 상태입니다.

오류 - 패시브 상태에서 메시지 및 패시브 오류 프레임 (열성 비트로 만들어 짐)이 전송 될 수 있습니다.

버스 오프 상태는 스테이션이 버스 통신에 참여하는 것을 일시적으로 불가능하게 만듭니다.

이 상태에서는 메시지를 수신하거나 수신 할 수 없습니다.
전송되었습니다. 송신기 만 버스 접속이 가능합니다.

6.7 오류 모드 및 오류 카운터

MCP2515에는 수신 오류 카운터 (REC) (레지스터 6-2 참조)와 송신 오류 카운터 (TEC)의 두 가지 오류 카운터가 있습니다 (레지스터 6-1 참조).

두 카운터의 값은 MCU에서 읽을 수 있습니다.

이 카운터는 CAN 버스 사양에 따라 증가 / 감소합니다.

MCP2515는 두 오류 카운터가 모두 128의 오류 - 패시브 한계보다 낮습니다.

적어도 하나의 오류 카운터가 있으면 오류 패시브입니다.
128과 같습니다.

TEC가 버스 오프 한계 인 255를 초과하면 버스 오프가됩니다.

버스 오프 복구 시퀀스가 수신 될 때까지 디바이스는이 상태를 유지합니다.

버스 오프 복구 시퀀스는 128 개의 발생과 11 개의 연속적인 리세 시브 비트로 구성된다 (그림 6-1 참조).

NOTE : 버스가 꺼진 후 MCP2515는 버스가 128 x 11 비트 시간 동안 유힬 상태 인 경우 MCU의 개입없이 오류 활성 상태로 복구됩니다.

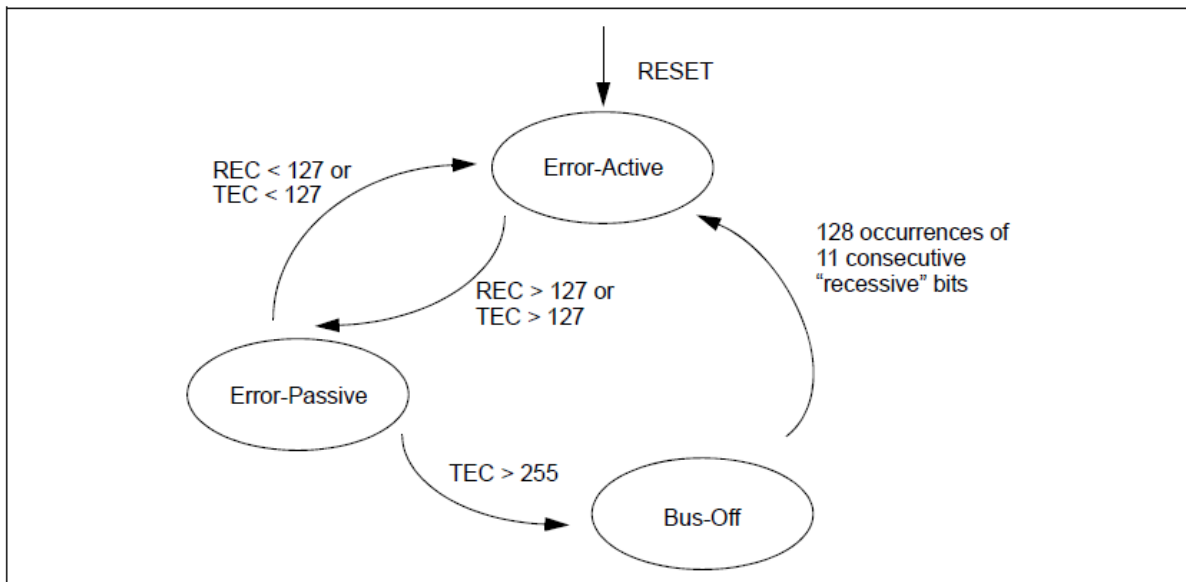
이것이 바람직하지 않은 경우, 에러 인터럽트 서비스 루틴이 문제를 해결해야 합니다.

MCP2515의 전류 오류 모드는 EFLG 레지스터 (레지스터 6-3 참조)를 통해 MCU에서 읽을 수 있습니다.

또한 오류 카운터 경고 플래그 비트 (EFLG : EWARN)가 있으며 오류 카운터 중 하나 이상이 96의 오류 경고 제한과 같거나 초과하면 설정됩니다.

두 오류 카운터가 모두 오류 경고 한계보다 작 으면 EWARN이 재설정됩니다.

FIGURE 6-1: ERROR MODES STATE DIAGRAM



**REGISTER 6-1: TEC – TRANSMIT ERROR COUNTER
(ADDRESS: 1Ch)**

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-0 TEC : 전송 오류 카운트 비트 <7 : 0>

**REGISTER 6-2: REC – RECEIVER ERROR COUNTER
(ADDRESS: 1Dh)**

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-0 REC : 수신 에러 카운트 비트 <7 : 0>

**REGISTER 6-3: EFLG – ERROR FLAG
(ADDRESS: 2Dh)**

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
RX1OVR	RX0OVR	TXBO	TXEP	RXEP	TXWAR	RXWAR	EWARN
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7 RX1OVR : 수신 버퍼 1 오버플로 플래그 비트

- RXB1 및 CANINTF.RX1IF = 1에 유효한 메시지가 수신되면 설정
- MCU로 재설정해야합니다.

비트 6 RX0OVR : 수신 버퍼 0 오버플로 플래그 비트

- RXB0 및 CANINTF.RX0IF = 1에 유효한 메시지가 수신 될 때 설정
- MCU로 재설정해야합니다.

비트 5 TXBO : 버스 오프 오류 플래그 비트

- TEC가 255에 도달하면 비트가 설정됩니다.
- 버스 복구 순서가 성공한 후 리셋

비트 4 TXEP : 전송 오류 - 패시브 플래그 비트

- TEC가 128 이상인 경우 설정
- TEC가 128보다 작 으면 재설정

비트 3 RXEP : 수신 오류 - 패시브 플래그 비트

- REC가 128보다 크거나 같을 때 설정
- REC가 128보다 작 으면 재설정

비트 2 TXWAR : 전송 오류 경고 플래그 비트

- TEC가 96 이상인 경우 설정
- TEC가 96 미만인 경우 재설정

비트 1 RXWAR : 수신 오류 경고 플래그 비트

- REC가 96 이상일 때 설정
- REC가 96 미만일 때 리셋

비트 0 EWARN : 오류 경고 플래그 비트

- TEC 또는 REC가 96 (TXWAR 또는 RXWAR = 1) 이상일 때 설정하십시오.
- REC와 TEC가 둘 다 96 미만일 때 리셋

7.0 중단

MCP2515에는 8 개의 인터럽트 소스가 있습니다.

CANINTE 레지스터는 각 인터럽트 소스에 대한 개별 인터럽트 가능 비트를 포함합니다.

CANINTF 레지스터는 각 인터럽트 소스에 해당하는 인터럽트 플래그 비트를 포함합니다.

인터럽트가 발생하면 INT 핀은 MCP2515에 의해 로우로 구동되고 MCU에 의해 인터럽트가 클리어 될 때까지 로우로 유지된다.

각 조건이 다른 경우 인터럽트를 지울 수 없습니다.

보통의 쓰기 작업보다는 CANINTF 레지스터의 플래그 비트를 재설정하는 데 bit modify 명령을 사용하는 것이 좋습니다.

이는 쓰기 명령 중에 변경되는 플래그를 의도하지 않게 변경하여 잠재적으로 인터럽트가 누락되는 것을 방지하기 위해 수행됩니다.

관련된 CANINTE 비트가 설정되어 있으면 CANINTF 플래그는 읽기 / 쓰기이며 인터럽트는 MCU가 이러한 비트 중 하나를 설정하여 생성 할 수 있습니다.

7.1 인터럽트 코드 비트

보류중인 인터럽트의 소스는 레지스터 10-2에 표시된대로 CANSTAT.ICOD (인터럽트 코드) 비트에 표시됩니다.

다중 인터럽트가 발생하는 경우 MCU가 모든 인터럽트를 리셋 할 때까지 INT는 로우로 유지됩니다.

CANSTAT.ICOD 비트는 현재 보류중인 최상위 우선 순위 인터럽트에 대한 코드를 반영합니다.

인터럽트는 ICODE 값이 낮을수록 인터럽트 우선 순위가 높아지도록 내부적으로 우선 순위가 지정됩니다.

우선 순위가 가장 높은 인터럽트 조건이 지워지면 대기중인 (있는 경우) 다음으로 우선 순위가 높은 인터럽트에 대한 코드가 ICODE 비트에 반영됩니다 (표 7-1 참조).

연관된 CANINTE 가능 비트가 설정된 인터럽트 소스 만 ICODE 비트에 반영됩니다.

TABLE 7-1: ICODE<2:0> DECODE

ICODE<2:0>	Boolean Expression
000	$\overline{\text{ERR}} \cdot \overline{\text{WAK}} \cdot \overline{\text{TX0}} \cdot \overline{\text{TX1}} \cdot \overline{\text{TX2}} \cdot \overline{\text{RX0}} \cdot \overline{\text{RX1}}$
001	ERR
010	$\overline{\text{ERR}} \cdot \overline{\text{WAK}}$
011	$\overline{\text{ERR}} \cdot \overline{\text{WAK}} \cdot \overline{\text{TX0}}$
100	$\overline{\text{ERR}} \cdot \overline{\text{WAK}} \cdot \overline{\text{TX0}} \cdot \overline{\text{TX1}}$
101	$\overline{\text{ERR}} \cdot \overline{\text{WAK}} \cdot \overline{\text{TX0}} \cdot \overline{\text{TX1}} \cdot \overline{\text{TX2}}$
110	$\overline{\text{ERR}} \cdot \overline{\text{WAK}} \cdot \overline{\text{TX0}} \cdot \overline{\text{TX1}} \cdot \overline{\text{TX2}} \cdot \overline{\text{RX0}}$
111	$\overline{\text{ERR}} \cdot \overline{\text{WAK}} \cdot \overline{\text{TX0}} \cdot \overline{\text{TX1}} \cdot \overline{\text{TX2}} \cdot \overline{\text{RX0}} \cdot \overline{\text{RX1}}$

참고 : ERR은 CANINE, EERIE와 관련됩니다.

7.2 송신 인터럽트

전송 인터럽트가 활성화되면 ($CANINTE.TXnIE = 1$), 관련 송신 버퍼가 비워지고 새로운 메시지를 로드 할 준비가 되면 INT 핀에서 인터럽트가 생성됩니다.

CANINTF.TXnIF 비트는 인터럽트의 소스를 나타내도록 설정됩니다.

인터럽트는 TXnIF 비트를 지우면 소거됩니다.

7.3 수신 인터럽트

수신 인터럽트가 활성화되면 ($CANINTE.RXnIE = 1$) 메시지가 성공적으로 수신되어 관련 수신 버퍼에 로드되면 인터럽트가 INT 핀에 생성됩니다.

이 인터럽트는 EOF 필드를 받은 직후에 활성화됩니다.

CANINTF.RXnIF 비트는 인터럽트의 원인을 나타내도록 설정됩니다.

인터럽트는 RXnIF 비트를 지워서 지워집니다.

7.4 메시지 오류 인터럽트

메시지의 송신이나 수신 중에 에러가 발생하면 메시지 에러 플래그 (CANINTF.MERRF)가 설정되고, CANINTE.MERRE 비트가 설정되면 인터럽트가 INT 핀에 생성됩니다.

이 기능은 청취 전용 모드와 함께 사용할 경우 전송 속도 결정을 용이하게하기 위해 사용됩니다.

7.5 버스 활동 웨이크 업 인터럽트

MCP2515가 절전 모드에 있고 버스 작동 웨이크 업 인터럽트가 활성화됩니다 ($CANINTE.WAKIE = 1$). INT 핀에서 인터럽트가 발생하고 CAN 버스에서 활동이 감지되면 CANINTF.WAKIF 비트가 설정됩니다.

이 인터럽트로 인해 MCP2515는 절전 모드를 종료합니다.

인터럽트는 WAKIF 비트를 지움으로써 리셋된다.

참고 : MCP2515는 수신 대기 모드로 작동합니다.

7.6 오류 인터럽트

오류 인터럽트가 활성화되면 (CANINTE.ERRIE = 1) 오버 플로우 조건이 발생하거나 송신기 또는 수신기의 오류 상태가 변경되면 INT 핀에 인터럽트가 생성됩니다.

오류 플래그 (EFLG) 레지스터는 다음 조건 중 하나를 나타냅니다.

7.6.1 수신자 오버플로

오버플로 상태는 MAB가 유효한 수신 메시지를 모으고 (메시지가 수락 필터의 기준을 충족) 필터와 관련된 수신 버퍼를 새 메시지로드에 사용할 수없는 경우에 발생합니다.

연관된 EFLG.RXnOVR 비트는 오버 플로우 조건을 나타내도록 설정됩니다.

이 비트는 MCU에 의해 제거되어야한다.

7.6.2 리시버 경고

REC가 MCU 경고 한계 인 96에 도달했습니다.

7.6.3 트랜스미터 경고

TEC가 MCU 경고 한계 인 96에 도달했습니다.

7.6.4 수신자 오류 - 수동

REC가 오류 패시브 한계 인 127을 초과하여 장치가 오류 패시브 상태가되었습니다.

7.6.5 송신기 오류 - 수동

TEC가 오류 - 수동 제한 인 127을 초과했으며 장치가 오류 패시브 상태가되었습니다.

7.6.6 BUS-OFF

TEC가 255를 초과했으며 장치가 버스 상태가되었습니다.

7.7 인터럽트 승인

인터럽트는 CANINTF 레지스터의 하나 이상의 상태 플래그와 직접 연관됩니다. 인터럽트는 플래그 중 하나가 설정되어있는 한 대기 중입니다.

일단 장치에 의해 인터럽트 플래그가 설정되면 인터럽트 조건이 제거 될 때까지 MCU가 플래그를 리셋 할 수 없습니다.

**REGISTER 7-1: CANINTE – INTERRUPT ENABLE
(ADDRESS: 2Bh)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MERRE	WAKIE	ERRIE	TX2IE	TX1IE	TX0IE	RX1IE	RX0IE
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

비트 7 MERRE : 메시지 오류 인터럽트 활성화 비트

1 = 메시지 수신 또는 전송 중 오류시 인터럽트

0 = 사용 불가

bit 6 WAKIE : 웨이크 업 인터럽트 활성화 비트

1 = CAN 버스 활동에 대한 인터럽트

0 = 사용 불가

bit 5 ERRIE : 오류 인터럽트 활성화 비트 (EFLG 레지스터의 다중 소스)

1 = EFLG 오류 조건 변경시 인터럽트

0 = 사용 불가

비트 4 TX2IE : 송신 버퍼 2 빈 인터럽트 활성화 비트

1 = TXB2가 비어있는 경우 인터럽트

0 = 사용 불가

비트 3 TX1IE : 송신 버퍼 1 빈 인터럽트 활성화 비트

1 = TXB1이 비게 될 때 인터럽트

0 = 사용 불가

비트 2 TX0IE : 송신 버퍼 0 빈 인터럽트 활성화 비트

1 = TXB0이 비게되면 인터럽트 됨

0 = 사용 불가

비트 1 RX1IE : 수신 버퍼 1 완전 인터럽트 활성화 비트

1 = RXB1에서 메시지 수신시 인터럽트

0 = 사용 불가

비트 0 RX0IE : 수신 버퍼 0 완전 인터럽트 활성화 비트

1 = RXB0에서 메시지 수신시 인터럽트

0 = 사용 불가

**REGISTER 7-2: CANINTF – INTERRUPT FLAG
(ADDRESS: 2Ch)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MERRF	WAKIF	ERRIF	TX2IF	TX1IF	TX0IF	RX1IF	RX0IF
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

비트 7 MERRF : 메시지 오류 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 6 WAKIF : 웨이크 업 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 5 ERRIF : 오류 인터럽트 플래그 비트 (EFLG 레지스터의 다중 소스)

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 4 TX2IF : 송신 버퍼 2 빈 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 3 TX1IF : 송신 버퍼 1 빈 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 2 TX0IF : 송신 버퍼 0 빈 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 1 RX1IF : 수신 버퍼 1 완전 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

비트 0 RX0IF : 수신 버퍼 0 완전 인터럽트 플래그 비트

1 = 인터럽트 보류 중 (인터럽트 조건을 리셋하기 위해 MCU에 의해 클리어되어야 함)

0 = 보류중인 인터럽트 없음

8.0 오실레이터

MCP2515는 크리스털 또는 세라믹 공진기를 OSC1 및 OSC2 핀에 연결하여 작동하도록 설계되었습니다.

MCP2515 오실레이터 설계에는 병렬 컷 크리스털이 필요합니다.

직렬 컷 크리스털을 사용하면 크리스털 제조업체 사양에서 벗어나는 빈도가 발생할 수 있습니다.

일반적인 발진기 회로가 그림 8-1에 나와 있습니다.

MCP2515는 그림 8-2와 그림 8-3에서와 같이 OSC1 핀에 연결된 외부 클럭 소스에 의해 구동 될 수도 있습니다.

8.1 발진기 시동 타이머

MCP2515는 MCP2515를 리셋 상태로 유지하는 오실레이터 기동 타이머 (OSC)를 사용하여 내부 상태 머신이 작동하기 전에 오실레이터가 안정화되었는지 확인합니다.

OST는 파워 업 또는 슬립 모드에서의 웨이크 업이 발생한 후 처음 128 개의 OSC1 클럭 사이클에 대해 리셋을 유지합니다.

OST가 만료 될 때까지 어떠한 SPI 프로토콜 동작도 시도되어서는 안된다는 점에 유의해야한다.

8.2 CLKOUT 핀

CLKOUT 핀은 시스템 설계자에게 제공되어 메인 시스템 클럭 또는 시스템의 다른 디바이스를위한 클록 입력으로 사용됩니다.

CLKOUT에는 FOSC를 1, 2, 4 및 8로 나눌 수있는 내부 프리 스케일러가 있습니다.

CLKOUT 기능이 활성화되고 프리스케일러가 CANCECTRL 레지스터를 통해 선택됩니다 (레지스터 10-1 참조).

참고 : CLKOUT의 최대 주파수는 25MHz로 지정됩니다 (표 13-5 참조)

CLKOUT 핀은 시스템 리셋시 활성화되어 가장 느린 속도 (8로 나누기)로 기본 설정되어 있습니다.

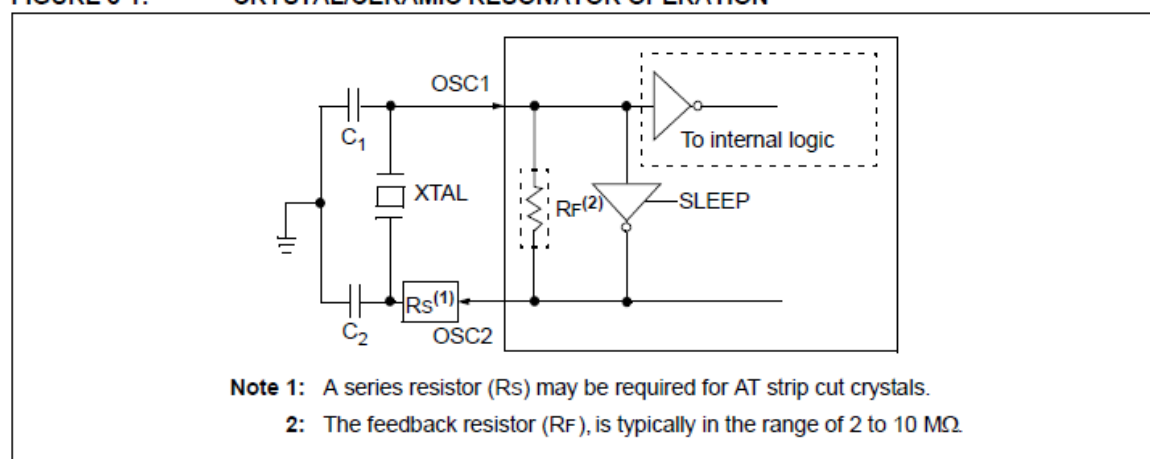
MCU 클럭으로 사용하십시오.

슬립 프 모드가 요청되면 MCP2515는 슬립 모드로 들어가기 전에 CLKOUT 핀에서 16 개의 추가 클럭 사이클을 구동합니다.

슬립 모드에서 CLKOUT 핀의 유휴 상태는 낮다. CLKOUT 기능이 비활성화되면 (CANCECTRL.CLKEN = '0') CLKOUT 핀은 하이 임피던스 상태가된다.

CLKOUT 기능은 thCLKOUT 및 tCLKOUT 타이밍이 CLKOUT 핀 기능이 활성화되거나 비활성화되거나 프리스케일러 값이 변경됩니다.

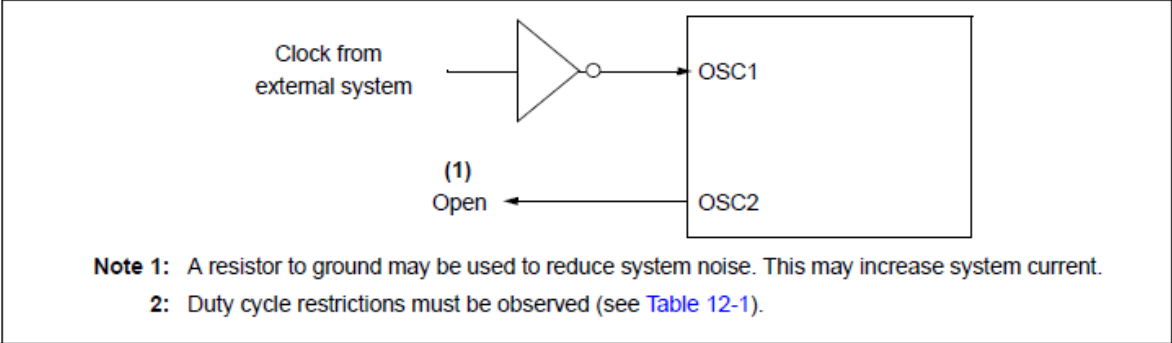
FIGURE 8-1: CRYSTAL/CERAMIC RESONATOR OPERATION



참고 1 : AT 스트립 컷 크리스탈에는 직렬 저항 (RS)이 필요할 수 있습니다.

2 : 피드백 저항 (RF)은 일반적으로 2 ~ 10 MΩ 입니다.

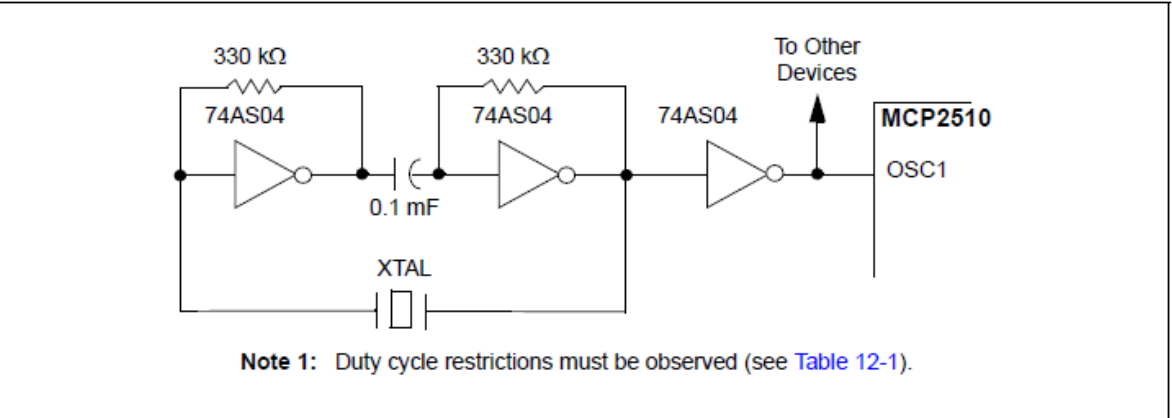
FIGURE 8-2: EXTERNAL CLOCK SOURCE



참고 1 : 접지에 대한 저항을 사용하여 시스템 잡음을 줄일 수 있습니다. 이로 인해 시스템 전류가 증가 할 수 있습니다.

2 : 듀티 사이클 제한 사항을 준수해야 합니다 (표 12-1 참조).

FIGURE 8-3: EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT⁽¹⁾



참고 1 : 듀티 사이클 제한 사항을 준수해야 합니다 (표 12-1 참조).

표 8-1 : 세라믹 공진기의 커패시터 선택

사용 된 일반적인 커패시터 값 :			
mode	freq	osc1	osc2
HS	8.0 MHz	27 pF	27 pF
	16.0 MHz	22 pF	22 pF
커패시터 값은 설계 지침 용입니다.			

이 콘덴서는 기본적인 스타트 업 및 작동을 위해 아래에 나열된 공진기로 테스트되었습니다.

이 값은 최적화되지 않았습니다.

수용 가능한 발진기 동작을 생성하기 위해 서로 다른 커패시터 값이 필요할 수 있다.

사용자는 예상되는 VDD 및 온도 범위에서 발진기의 성능을 테스트해야한다.

추가 정보는 표 8-2의 설명을 참조하십시오.

사용 된 공진기 :

4.0 MHz

8.0 MHz

16.0 MHz

표 8-2 : 크리스털 오실레이터의 커패시터 선택

OSC 타입	결정 주파수	테스트 한 대표적인 CapacitorValues :	
		C1	C2
HS	4 MHz	27 pF	27 pF
	8 MHz	22 pF	22 pF
	20MHz	15 pF	15 pF

커패시터 값은 설계 지침 용입니다.

이 축전기는 기본적인 시운전 및 작동을 위해 아래에 나열된 크리스털로 테스트되었습니다.

이 값은 최적화되지 않았습니다.

수용 가능한 발진기 동작을 생성하기 위해 서로 다른 커패시터 값이 필요할 수 있다.

사용자는 예상되는 VDD 및 온도 범위에서 발진기의 성능을 테스트해야 한다.

추가 정보는 이 표 다음의 참고 사항을 참조하십시오.

사용된 결정 (3) :

4.0 MHz

8.0 MHz

20.0 MHz

NOTE 1 : 커패시턴스가 높을수록 발진기의 안정성이 향상되지만 시동 시간도 증가합니다.

2 : 각 공진기 / 크리스탈은 고유한 특성을 가지고 있으므로 사용자는 공진기 / 수정 제조업체에 적절한 외부 부품 값을 문의해야 합니다.

3 : RS는 드라이브 레벨 사양이 낮은 크리스탈의 오버 드라이브를 피하기 위해 필요할 수 있습니다.

4 : 어플리케이션에서 예상되는 VDD 및 온도 범위에서 오실레이터 성능을 항상 확인하십시오.

9.0 리셋

MCP2515는 두 가지 리셋을 구별합니다 :

1. 하드웨어 리셋 - RESET 핀이 부족합니다.
2. SPI 리셋 - SPI 명령을 통해 리셋됩니다.

이러한 재설정은 모두 기능상으로 동일합니다.

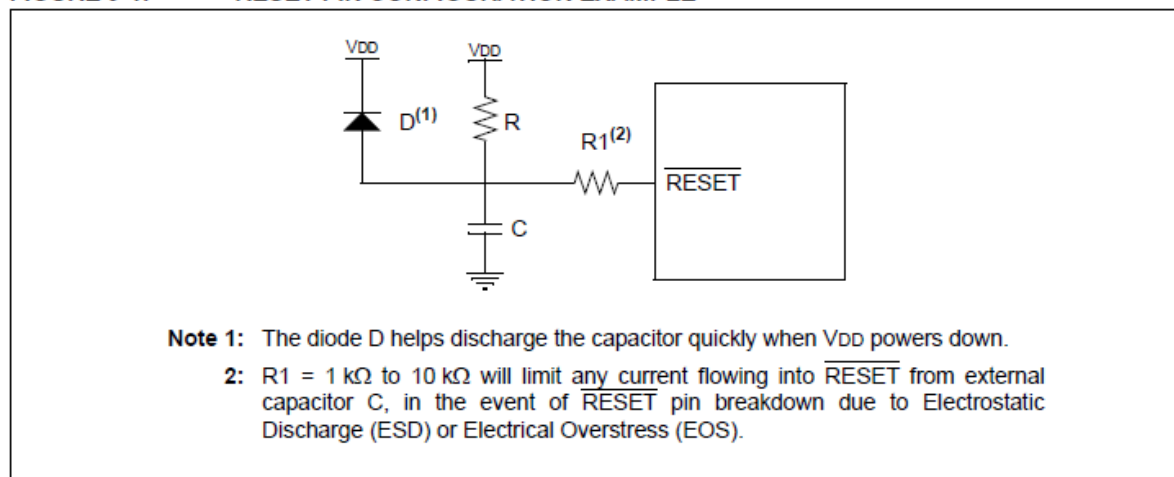
파워 업 후에이 두 리셋 중 하나를 제공하여 로직 및 레지스터가 기본 상태인지 확인하는 것이 중요합니다.

RESET 핀에 RC를 배치하여 하드웨어 리셋을 자동으로 수행 할 수 있습니다.

(그림 9-1 참조).

값은 VDD가 동작 전압에 도달 한 후 최소 $2\mu s$ 동안 디바이스가 리셋 상태로 유지되어야하며, 전기적 사양 (tRL)에 표시되어 있어야한다.

FIGURE 9-1: RESET PIN CONFIGURATION EXAMPLE



NOTE 1 : 다이오드 D는 VDD 전원이 꺼지면 커패시터를 빠르게 방전하는 데 도움이됩니다.

2 : R1 = 1k Ω ~ 10k Ω 은 정전기 방전 (ESD) 또는 전기 과부하 (EOS)로 인한 RESET 핀 고장의 경우 외부 커패시터 C에서 RESET으로 흐르는 전류를 제한합니다.

10.0 작동 방식

MCP2515에는 5 가지 작동 모드가 있습니다.

이러한 모드는 다음과 같습니다.

1. 구성 모드.
2. 정상 모드.
3. 수면 모드.
4. 듣기 전용 모드.
5. 루프백 모드.

동작 모드는 CANCTRL.REQOP 비트를 통해 선택된다 (레지스터 10-1 참조).

모드를 변경할 때 보류중인 모든 메시지 전송이 완료 될 때까지 모드가 실제로 변경되지 않습니다.

요청 된 모드는 CANSTAT.OPMODE 비트 (레지스터 10-2 참조)를 읽음으로써 검증되어야한다.

10.1 구성 모드

MCP2515는 정품 인증 전에 초기화해야합니다.

이는 장치가 구성 모드에있는 경우에만 가능합니다.

구성 모드는 CANTRL.REQOP 비트를 '100'으로 설정하여 전원을 켜 후 자동으로 선택되거나 재설정되거나 다른 모드에서 입력 될 수 있습니다.

구성 모드가 시작되면 모든 오류 카운터가 지워집니다.

구성 모드는 다음 레지스터를 수정할 수있는 유일한 모드입니다.

- CNF1, CNF2, CNF3
- TXRTSCTRL
- 필터 레지스터
- 마스크 레지스터

10.2 수면 모드

MCP2515는 장치의 전류 소비를 최소화하기 위해 사용되는 내부 절전 모드를 갖추고 있습니다.

SPI 인터페이스는 MCP2515가 슬립 모드에있을 때에도 읽기 용으로 활성화되어 모든 레지스터에 액세스 할 수 있습니다.

슬립 모드로 들어가려면 모드 요청 비트가 CANCTRL 레지스터에 설정됩니다 (REQOP <2 : 0>).

CANSTAT.OPMODE 비트는 작동 모드를 나타냅니다.

이 비트는 MCP2515에 슬립 명령을 보낸 후에 읽어야합니다.

MCP2515는 활성 상태이며이 비트가 슬립 모드로 들어갈 때까지 슬립 모드로 들어 가지 않았습니다.

내부 절전 모드에있을 때, 절전 모드 인터럽트는 활성 상태입니다 (활성화 된 경우).

이는 MCU가 슬립 모드에 놓이게하고 MCP2515를 사용하여 버스에서의 활동을 감지했을 때 웨이크 업 할 수 있도록한다.

슬립 모드에있을 때 MCP2515는 내부 오실레이터를 멈춘다.

MCP2515는 버스 동작이 발생하거나 MCU가 SPI 인터페이스를 통해 CANINTF.WAKIF 비트를 설정하여 웨이크 업 시도를 '생성'할 때 웨이크 업합니다 (웨이크 업 인터럽트가 발생하려면 CANINTE.WAKIE 비트도 설정해야합니다).

TXCAN 핀은 MCP2515가 절전 모드에있는 동안 열성 상태로 유지됩니다.

10.2.1 깨우기 기능

장치는 슬립 모드에있는 동안 RXCAN 핀의 활동을 모니터링합니다.

CANINTE.WAKIE 비트가 설정되면 장치는 깨어나고 인터럽트를 생성합니다.

절전 모드에서는 내부 오실레이터가 종료되기 때문에 모드로 설정하면 발진기에 약간의 시간이 걸립니다 장치가 시작되어 메시지 수신을 활성화합니다.

이 오실레이터 시작 타이머 (OST)는 128 TOSC로 정의됩니다.

장치는 절전 모드에서 깨어나 기 시작한 메시지와 장치가 '깨어 난'중에 발생하는 메시지를 무시합니다.

장치는 청취 전용 모드로 활성화됩니다.

DSTEMP가 버스에서 통신 할 수 있으려면 MCU가 정상 모드로 설정해야합니다.

이 디바이스는 내부 슬립 모드에서 로우 패스 필터 기능을 RXCAN 입력 라인에 적용하도록 프로그래밍 할 수있다.

이 기능은 CAN 버스 라인의 짧은 글리치로 인해 장치가 깨어나는 것을 방지하는 데 사용할 수 있습니다.

CNF3.WAKFIL 비트는 필터를 활성화하거나 비활성화합니다.

10.3 수신 전용 모드

듣기 전용 모드는 RXBnCTRL.RXM <1 : 0> 비트를 구성하여 MCP2515가 모든 메시지 (오류가있는 메시지 포함)를 수신 할 수있는 방법을 제공합니다.

이 모드는 버스 모니터 어플리케이션이나 '핫 플러그 (hot plugging)'상황에서 보드 율을 감지하는 데 사용할 수 있습니다.

자동 보드 감지의 경우 서로 통신하는 다른 노드가 두 개 이상 필요합니다.

유효한 메시지가 수신 될 때까지 다른 값을 테스트하여 경험적으로 보오율을 감지 할 수 있습니다.

듣기 전용 모드는 자동 모드입니다.이 모드에서는 오류 플래그 또는 확인 신호를 포함하여 메시지가 전송되지 않습니다.

필터와 마스크를 사용하여 특정 메시지 만 수신 레지스터에 로드 할 수있게하거나 마스크를 모든 0으로 설정하여 모든 식별자가있는 메시지를 전달할 수 있습니다.

오류 카운터는이 상태에서 재설정 및 비활성화됩니다.

청취 전용 모드는 CANCTRL 레지스터의 모드 요청 비트를 설정하여 활성화됩니다.

10.4 루프백 모드

루프백 모드는 실제로 CAN 버스에서 메시지를 전송하지 않고 송신 버퍼에서 수신 버퍼로 메시지를 내부 전송할 수 있게합니다.

이 모드는 시스템 개발 및 테스트에 사용할 수 있습니다.

이 모드에서는 ACK 비트가 무시되고 장치는 마치 다른 노드에서 들어오는 것처럼 들어오는 메시지를 허용합니다.

루프백 모드는 자동 모드입니다. 즉, 이 상태 (오류 플래그 또는 확인 신호 포함) 중에는 메시지가 전송되지 않습니다.

TXCAN 핀은 열성 상태에 있습니다.

필터와 마스크를 사용하여 특정 메시지 만 수신 레지스터에 로드 할 수 있습니다.

모든 메시지를 받아들이는 모드를 제공하기 위해 마스크를 모두 0으로 설정할 수 있습니다.

Loopback 모드는 CANCTRL 레지스터에 모드 요청 비트를 설정하여 활성화됩니다.

10.5 정상 모드

일반 모드는 MCP2515의 표준 작동 모드입니다.

이 모드에서 장치는 모든 버스 메시지를 능동적으로 모니터링하고 확인 비트, 오류 프레임 등을 생성합니다.

이것은 또한 MCP2515가 CAN 버스를 통해 메시지를 전송할 수 있는 유일한 모드입니다.

REGISTER 10-1: CANCTRL – CAN CONTROL REGISTER
(ADDRESS: XFh)

R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1
REQOP2	REQOP1	REQOP0	ABAT	OSM	CLKEN	CLKPRE1	CLKPRE0
bit 7							bit 0

Legend:							
R = Readable bit		W = Writable bit		U = Unimplemented bit, read as '0'			
-n = Value at POR		'1' = Bit is set		'0' = Bit is cleared		x = Bit is unknown	

비트 7-5 REQOP : 요구 동작 모드 비트 <2 : 0>

000 = 정상 작동 모드 설정

001 = 수면 모드 설정

010 = 루프백 모드 설정

011 = 듣기 전용 모드 설정

100 = 설정 모드 설정

REQOP 비트의 다른 모든 값은 유효하지 않으므로 사용해서는 안됩니다. 전원을 켜올 때
REQOP = b'111 '

비트 4 ABAT : 보류중인 모든 전송 중단 비트

1 = 보류중인 모든 송신 버퍼의 중단 요청

0 = 모든 전송을 중단하라는 요청을 종료합니다.

비트 3 OSM : 원 샷 모드 비트

1 = 사용함. 메시지는 한 번만 전송하려고 시도합니다.

0 = 비활성화 됨. 메시지는 필요한 경우 전송을 다시 시도합니다.

비트 2 CLKEN : CLKOUT 핀 인 에이블 비트

1 = CLKOUT 핀 가능

0 = CLKOUT 핀 디스 에이블 (핀은 하이 - 임피던스 상태 임)

비트 1-0 CLKPRE : CLKOUT 핀 프리 스케일러 비트 <1 : 0>

00 = FCLKOUT = 시스템 클록 / 1

01 = FCLKOUT = 시스템 클록 / 2

10 = FCLKOUT = 시스템 클록 / 4

11 = FCLKOUT = 시스템 클록 / 8

REGISTER 10-2: CANSTAT – CAN STATUS REGISTER
(ADDRESS: XEh)

R-1	R-0	R-0	U-0	R-0	R-0	R-0	U-0
OPMOD2	OPMOD1	OPMOD0	—	ICOD2	ICOD1	ICOD0	—
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

비트 7-5 OPMOD : 동작 모드 비트 <2 : 0>

000 = 장치가 정상 작동 모드입니다.

001 = 장치가 절전 모드입니다.

010 = 장치가 루프백 모드에 있습니다.

011 = 장치가 수신 대기 모드에 있습니다.

100 = 장치가 구성 모드에 있음

비트 4 비 구현 : '0'

비트 3-1 ICODE : 인터럽트 플래그 코드 비트 <2 : 0>

000 = 인터럽트 없음

001 = 오류 인터럽트

010 = 깨우기 인터럽트

011 = TXB0 인터럽트

100 = TXB1 인터럽트

101 = TXB2 인터럽트

110 = RXB0 인터럽트

111 = RXB1 인터럽트

비트 0 구현되지 않음 : '0'으로 읽음

11.0 등록 맵

MCP2515에 대한 레지스터 맵은 표 11-1에 나와 있습니다.

각 레지스터의 주소 위치는 열 (상위 4 비트) 및 행 (하위 4 비트) 값을 사용하여 결정됩니다.

레지스터는 데이터의 순차적 읽기 및 쓰기를 최적화하도록 배열되었습니다.

일부 특정 제어 및 상태 레지스터는 SPI 비트 수정 명령을 사용하여 개별 비트 수정을 허용합니다.

이 명령을 허용하는 레지스터는 표 11-1에서 음영 처리 된 위치로 표시됩니다.

MCP2515 제어 레지스터의 요약은 표 11-2에 나와 있습니다.

TABLE 11-1: CAN CONTROLLER REGISTER MAP

Lower Address Bits	Higher-Order Address Bits							
	0000 xxxx	0001 xxxx	0010 xxxx	0011 xxxx	0100 xxxx	0101 xxxx	0110 xxxx	0111 xxxx
0000	RXF0SIDH	RXF3SIDH	RXM0SIDH	TXB0CTRL	TXB1CTRL	TXB2CTRL	RXB0CTRL	RXB1CTRL
0001	RXF0SIDL	RXF3SIDL	RXM0SIDL	TXB0SIDH	TXB1SIDH	TXB2SIDH	RXB0SIDH	RXB1SIDH
0010	RXF0EID8	RXF3EID8	RXM0EID8	TXB0SIDL	TXB1SIDL	TXB2SIDL	RXB0SIDL	RXB1SIDL
0011	RXF0EID0	RXF3EID0	RXM0EID0	TXB0EID8	TXB1EID8	TXB2EID8	RXB0EID8	RXB1EID8
0100	RXF1SIDH	RXF4SIDH	RXM1SIDH	TXB0EID0	TXB1EID0	TXB2EID0	RXB0EID0	RXB1EID0
0101	RXF1SIDL	RXF4SIDL	RXM1SIDL	TXB0DLC	TXB1DLC	TXB2DLC	RXB0DLC	RXB1DLC
0110	RXF1EID8	RXF4EID8	RXM1EID8	TXB0D0	TXB1D0	TXB2D0	RXB0D0	RXB1D0
0111	RXF1EID0	RXF4EID0	RXM1EID0	TXB0D1	TXB1D1	TXB2D1	RXB0D1	RXB1D1
1000	RXF2SIDH	RXF5SIDH	CNF3	TXB0D2	TXB1D2	TXB2D2	RXB0D2	RXB1D2
1001	RXF2SIDL	RXF5SIDL	CNF2	TXB0D3	TXB1D3	TXB2D3	RXB0D3	RXB1D3
1010	RXF2EID8	RXF5EID8	CNF1	TXB0D4	TXB1D4	TXB2D4	RXB0D4	RXB1D4
1011	RXF2EID0	RXF5EID0	CANINTE	TXB0D5	TXB1D5	TXB2D5	RXB0D5	RXB1D5
1100	BFPCTRL	TEC	CANINTF	TXB0D6	TXB1D6	TXB2D6	RXB0D6	RXB1D6
1101	TXRTSCTRL	REC	EFLG	TXB0D7	TXB1D7	TXB2D7	RXB0D7	RXB1D7
1110	CANSTAT	CANSTAT	CANSTAT	CANSTAT	CANSTAT	CANSTAT	CANSTAT	CANSTAT
1111	CANCTRL	CANCTRL	CANCTRL	CANCTRL	CANCTRL	CANCTRL	CANCTRL	CANCTRL

Note: Shaded register locations indicate that these allow the user to manipulate individual bits using the Bit Modify command.

참고 : 음영 처리 된 레지스터 위치는 사용자가 비트 수정 명령을 사용하여 개별 비트를 조작할 수 있음을 나타냅니다.

TABLE 11-2: CONTROL REGISTER SUMMARY

Register Name	Address (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/RST Value
BFPCTRL	0C	—	—	B1BFS	B0BFS	B1BFE	B0BFE	B1BFM	B0BFM	--00 0000
TXRTSCTRL	0D	—	—	B2RTS	B1RTS	B0RTS	B2RTSM	B1RTSM	B0RTSM	--xx x000
CANSTAT	xE	OPMOD2	OPMOD1	OPMOD0	—	ICOD2	ICOD1	ICOD0	—	100- 000-
CANCTRL	xF	REQOP2	REQOP1	REQOP0	ABAT	OSM	CLKEN	CLKPRE1	CLKPRE0	1110 0111
TEC	1C	Transmit Error Counter (TEC)								0000 0000
REC	1D	Receive Error Counter (REC)								0000 0000
CNF3	28	SOF	WAKFIL	—	—	—	PHSEG22	PHSEG21	PHSEG20	00-- -000
CNF2	29	BTLMODE	SAM	PHSEG12	PHSEG11	PHSEG10	PRSEG2	PRSEG1	PRSEG0	0000 0000
CNF1	2A	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0	0000 0000
CANINTE	2B	MERRE	WAKIE	ERRIE	TX2IE	TX1IE	TX0IE	RX1IE	RX0IE	0000 0000
CANINTF	2C	MERRF	WAKIF	ERRIF	TX2IF	TX1IF	TX0IF	RX1IF	RX0IF	0000 0000
EFLG	2D	RX1OVR	RX0OVR	TXBO	TXEP	RXEP	TXWAR	RXWAR	EWARN	0000 0000
TXB0CTRL	30	—	ABTF	MLOA	TXERR	TXREQ	—	TXP1	TXP0	-000 0-00
TXB1CTRL	40	—	ABTF	MLOA	TXERR	TXREQ	—	TXP1	TXP0	-000 0-00
TXB2CTRL	50	—	ABTF	MLOA	TXERR	TXREQ	—	TXP1	TXP0	-000 0-00
RXB0CTRL	60	—	RXM1	RXM0	—	RXRTR	BUKT	BUKT	FILHIT0	-00- 0000
RXB1CTRL	70	—	RSM1	RXM0	—	RXRTR	FILHIT2	FILHIT1	FILHIT0	-00- 0000

12.0 SPI 인터페이스

12.1 개요

MCP2515는 많은 마이크로 컨트롤러에서 사용할 수 있는 SPI (Serial Peripheral Interface) 포트와 직접 인터페이스하도록 설계되었으며 모드 0,0 및 모드 1,1을 지원합니다.

명령과 데이터는 SI 핀을 통해 디바이스로 보내지며, SCK의 상승 에지에서 데이터가 클럭 인된다.

데이터는 SCK의 하강 에지에서 MCP2515 (SO 라인)에 의해 구동된다.

어떤 동작이 수행되는 동안 CS 핀은 로우로 유지되어야한다.

표 12-1은 모든 연산의 명령어 바이트를 보여줍니다.

모드 0,0 및 모드 1,1 작동에 대한 자세한 입력 및 출력 타이밍 다이어그램은 그림 12-10 및 그림 12-11을 참조하십시오.

참고 : MCP2515는 CS를 명령 / 명령 바이트로 낮추고 첫 번째 바이트를 기대합니다.

이는 CS를 올린 다음 다시 낮추어 다른 명령을 호출해야 함을 의미합니다.

12.2 리셋 명령

재설정 명령은 MCP2515의 내부 레지스터를 다시 초기화하고 구성 모드를 설정하는 데 사용할 수 있습니다.

이 명령은 SPI 인터페이스를 통해 RESET 핀과 동일한 기능을 제공합니다.

리셋 명령은 CS를 로우로 풀고 명령 바이트를 전송 한 다음 CS를 올려서 장치를 선택해야하는 싱글 바이트 명령입니다.

리셋 명령을 전송하는 것이 좋습니다.

(또는 RESET 핀을 내리십시오).

12.3 읽기 명령어

읽기 명령어는 CS 핀을 낮춰서 시작됩니다.

그런 다음 읽기 명령어가 MCP2515로 전송되고 그 뒤에 8 비트 주소 (A7에서 A0까지)가 전송됩니다.

다음으로, 선택된 어드레스의 레지스터에 저장된 데이터는 SO 핀에서 밖으로 이동 될 것이다.

데이터의 각 바이트가 이동되면 내부 주소 포인터가 자동으로 다음 주소로 증가합니다.

따라서, 클록 펄스를 계속해서 제공함으로써 다음 연속 레지스터 어드레스를 판독하는 것이 가능하다.

이 방법을 사용하여 임의의 수의 연속 레지스터 위치를 순차적으로 읽을 수 있습니다.

판독 동작은 CS 핀을 상승시킴으로써 종료된다

(그림 12-2).

12.4 RX 버퍼 명령 읽기

Read RX Buffer 명령 (그림 12-3)은 수신 버퍼의 주소를 빠르게 지정하는 방법을 제공합니다.

이 명령어는 SPI 오버 헤드를 주소 바이트 인 1 바이트 씩 줄입니다.

명령 바이트에는 실제로 주소 포인터 위치를 결정하는 네 개의 가능한 값이 있습니다.

커맨드 바이트가 보내지면, 컨트롤러는 어드레스 위치에서 판독 명령과 동일한 데이터를 클록 아웃한다 (즉, 순차 판독이 가능하다).

이 명령은 명령이 끝날 때 CS가 발생하면 관련 수신 플래그 (CANINTF.RXnIF)를 자동으로 지워 SPI 오버 헤드를 줄입니다.

12.5 쓰기 명령

쓰기 명령은 CS 핀을 낮춰서 시작됩니다.

그런 다음 쓰기 명령어가 MCP2515로 전송 된 다음 주소와 최소 1 바이트의 데이터가옵니다.

CS가 로우로 유지되는 한 데이터 바이트로 계속해서 클럭킹함으로써 순차 레지스터에 쓸 수 있습니다.

데이터는 실제로 D0 비트에 대한 SCK 라인의 상승 에지에서 레지스터에 기록됩니다.

8 비트가 로드되기 전에 CS 라인이 하이가되면 해당 데이터 바이트에 대해 쓰기가 중단되고 명령의 이전 바이트가 기록됩니다.

바이트 쓰기 순서에 대한 자세한 설명은 그림 12-4의 타이밍 다이어그램을 참조하십시오.

12.6 로드 TX 버퍼 명령어

Load TX Buffer 명령 (그림 12-5)은 다음을 제거합니다.

통상의 기입 커맨드에 필요한 8 비트의 주소.

8 비트 명령은 주소 포인터를 여섯 개의 주소 중 하나에 설정하여 세 개의 전송 버퍼 중 하나의 "ID"또는 "데이터"주소를 가리키는 전송 버퍼에 빠르게 씁니다.

12.7 전송 요청 (RTS)

명령 RTS 명령은 하나 이상의 전송 버퍼에 대한 메시지 전송을 시작하는 데 사용할 수 있습니다.

MCP2515는 CS 핀을 낮춤으로써 선택된다.

그런 다음 RTS 명령 바이트가 전송됩니다.

그림 12-6에 표시된 명령의 마지막 3 비트는 전송할 수 있는 전송 버퍼를 나타냅니다.

이 명령은 TxBnCTRL.TXREQ 비트를 각각의 버퍼 (들).

마지막 3 비트 중 일부 또는 전부는 단일 명령으로 설정할 수 있습니다.

nnn = 000으로 RTS 명령을 보내면 명령이 무시됩니다.

12.8 상태 명령 읽기

상태 읽기 명령은 메시지 수신 및 전송에 자주 사용되는 일부 상태 비트에 단일 명령 액세스를 허용합니다.

MCP2515는 CS 핀을 낮춤으로써 선택되며 그림 12-8과 같은 읽기 상태 명령 바이트가 MCP2515로 전송됩니다.

명령 바이트가 보내지면 MCP2515는 상태가 들어있는 8 비트의 데이터를 반환합니다.

처음 8 비트가 전송 된 후에 추가 클럭이 전송되면 MCP2515는 CS 핀이 로우로 유지되고 SCK에 클럭이 제공되는 한 상태 비트를 계속 출력합니다.

이 명령에서 반환 된 각 상태 비트는 적절한 레지스터 주소와 함께 표준 읽기 명령을 사용하여 읽을 수도 있습니다.

12.9 RX 상태 명령

RX 상태 명령 (그림 12-9)은 메시지 및 메시지 유형 (표준, 확장, 원격)과 일치하는 필터를 신속하게 결정하는 데 사용됩니다.

명령 바이트가 전송 된 후 컨트롤러는 상태 데이터가 들어있는 8 비트의 데이터를 반환합니다.

8 비트가 전송 된 후에 더 많은 클럭이 전송되면 컨트롤러는 CS 핀이 로우 상태를 유지하고 클럭이 제공되는 한 동일한 상태 비트를 계속 출력합니다.

12.10 비트 수정 명령

비트 수정 명령어는 특정 상태 및 제어 레지스터에서 개별 비트를 설정하거나 지우는 수단을 제공합니다.

이 명령은 모든 레지스터에서 사용할 수 없습니다.

이 명령을 사용할 수 있는 레지스터를 결정하려면 11.0 "레지스터 맵"을 참조하십시오.

NOTE : 비트 수정 가능하지 않은 레지스터에 대해 비트 수정 명령을 실행하면 마스크가 FFh가됩니다.

이것은 바이트 라이트를 레지스터에 허용하고 비트 수정은 허용하지 않습니다.

CS 핀을 낮춤으로써 부품을 선택하면 비트 수정 명령 바이트가 MCP2515로 전송됩니다.

이 명령은 레지스터 마스크 바이트의 마지막 데이터 바이트의 주소를 따른다.

마스크 바이트는 레지스터의 어느 비트가 변경 될지를 결정합니다.

마스크 바이트의 '1'은 레지스터의 비트가 변경되는 것을 허용하지만 '0'은 변경되지 않습니다.

데이터 바이트는 레지스터의 수정 된 비트가 어떤 값으로 변경 될지 결정합니다.

데이터 바이트의 '1'은 비트를 설정하고 '0'은 해당 비트의 마스크가 '1'로 설정된 경우 비트를 지웁니다 (그림 12-7 참조).

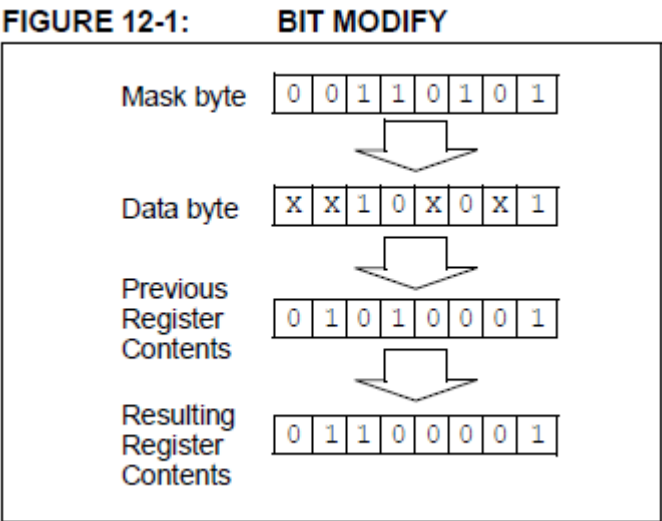


표 12-1 : SPI 지침 세트

instruction 이름	instruction 형태	설명
RESET	1100 0000	내부 레지스터를 기본 상태로 재설정하고 구성 모드를 설정합니다.
READ	0000 0011	선택한 주소에서 시작하여 레지스터에서 데이터를 읽습니다.
Read RX Buffer	1001 0nm0	수신 버퍼를 읽을 때 'n, m'으로 표시된 것처럼 4 개의 위치 중 하나에 주소 포인터를 배치하여 정상 읽기 명령의 오버 헤드를 줄입니다. NOTE : 관련 RX 플래그 비트 (CANINTF.RXnIF)는 CS를 하이로 전환 한 후에 지워진다.
WRITE	0000 0010	선택한 주소부터 시작하여 레지스터에 데이터를 씁니다.
Load TX Buffer	0100 0abc	전송 버퍼를로드 할 때 주소 포인터를 'a, b, c'로

		표시된 6 개 위치 중 하나에 배치하여 일반 쓰기 명령의 오버 헤드를 줄입니다.
RTS (Message Request-To-Send)	1000 0nnn	<p>컨트롤러에 전송 버퍼 중 하나에 대한 메시지 전송 시퀀스를 시작하도록 지시합니다.</p>
Read Status	1010 0000	전송을위한 여러 상태 비트를 읽는 빠른 폴링 명령 기능을 수신합니다.
RX Status	1011 0000	받은 메시지의 필터 일치 및 메시지 유형 (표준, 확장 및 / 또는 원격)을 나타내는 빠른 폴링 명령입니다.
Bit Modify	0000 0101	<p>특정 레지스터의 개별 비트를 설정하거나 지울 수 있습니다.</p> <p>NOTE :이 명령으로 모든 레지스터를 비트 수정 할 수있는 것은 아닙니다.</p> <p>비트 변조 불가능한 레지스터에 대해이 명령을 실행하면 마스크가 FFh가됩니다.</p> <p>적용되는 레지스터 목록은 11.0 "레지스터 맵"의 레지스터 맵을 참조하십시오.</p>

FIGURE 12-2: READ INSTRUCTION

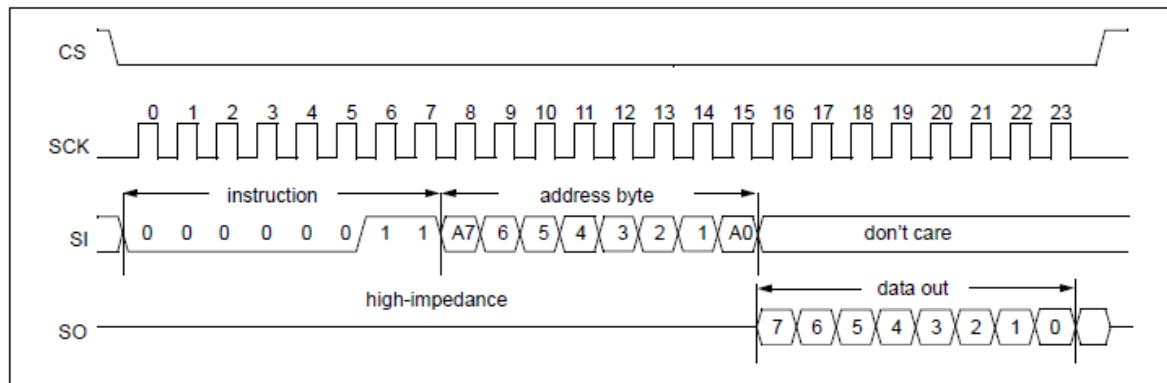


FIGURE 12-3: READ RX BUFFER INSTRUCTION

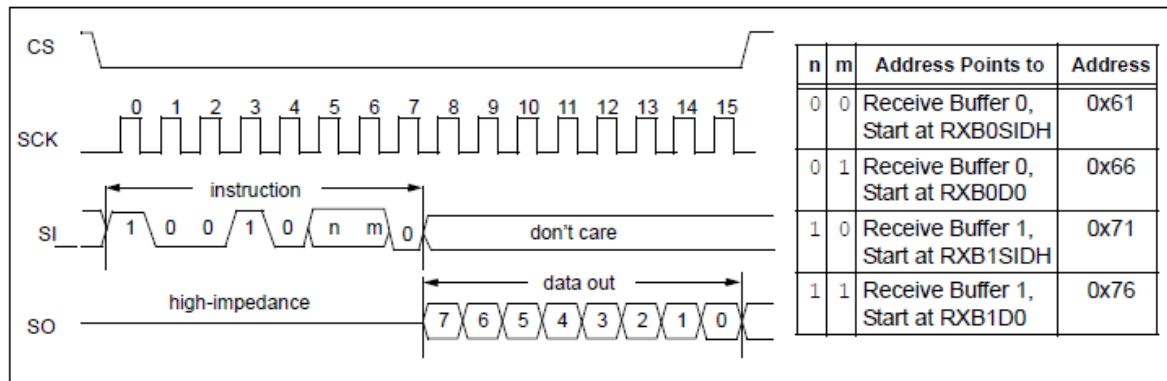


FIGURE 12-4: BYTE WRITE INSTRUCTION

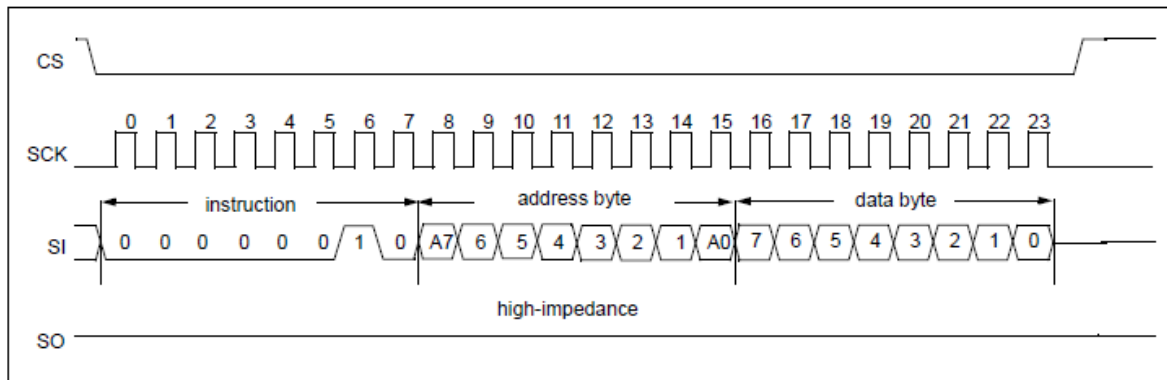


FIGURE 12-5: LOAD TX BUFFER

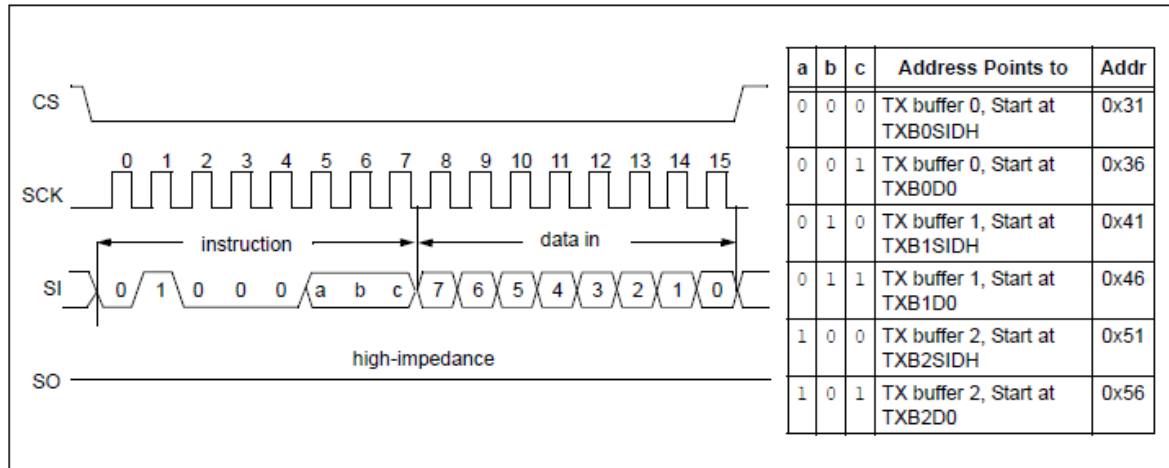


FIGURE 12-6: REQUEST-TO-SEND (RTS) INSTRUCTION

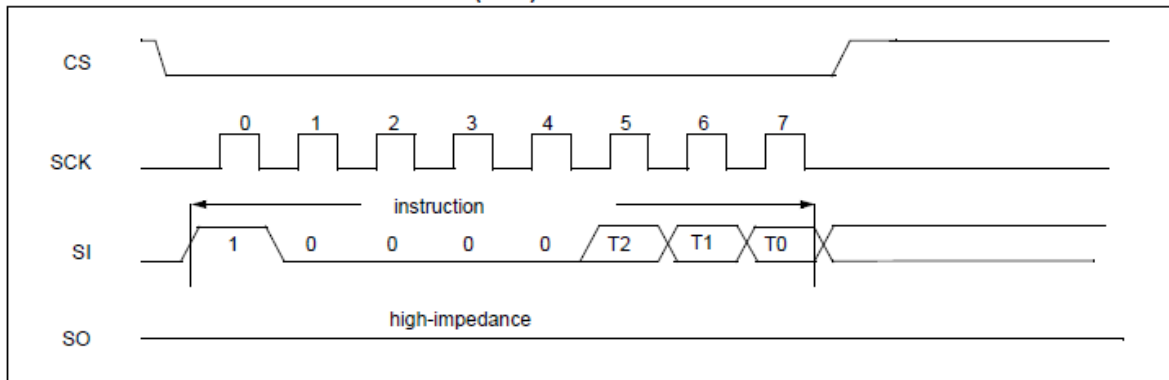
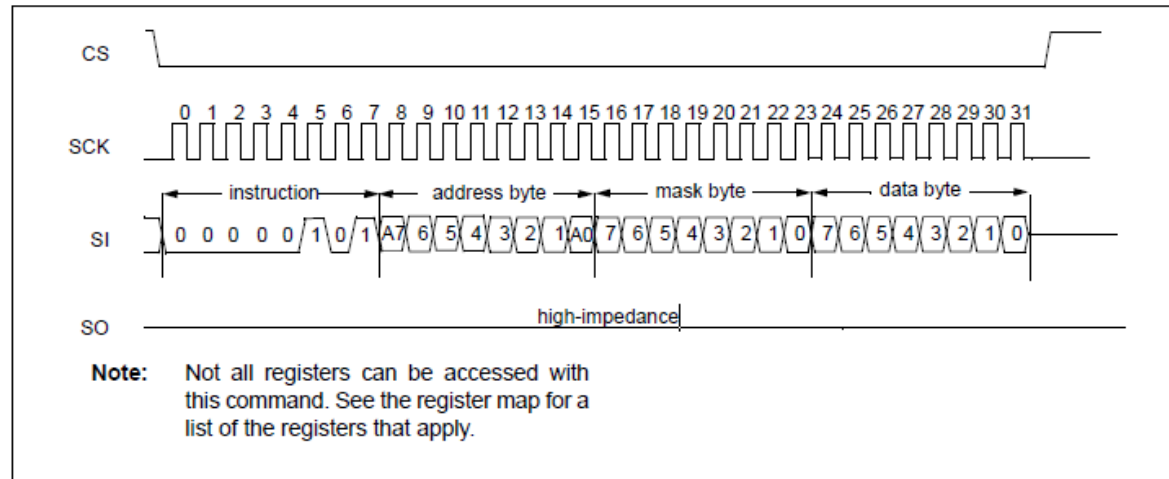


FIGURE 12-7: BIT MODIFY INSTRUCTION



NOTE :이 명령으로 모든 레지스터에 액세스 할 수있는 것은 아닙니다.

적용되는 레지스터 목록은 레지스터 맵을 참조하십시오.

FIGURE 12-8: READ STATUS INSTRUCTION

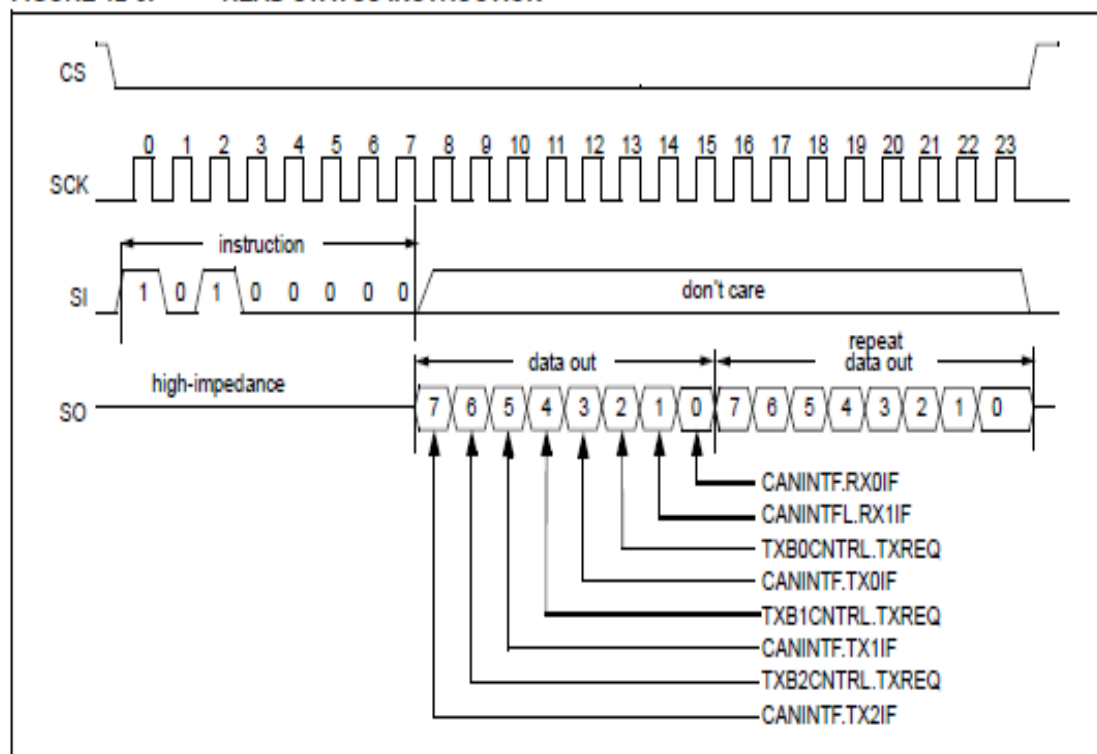


FIGURE 12-9: RX STATUS INSTRUCTION

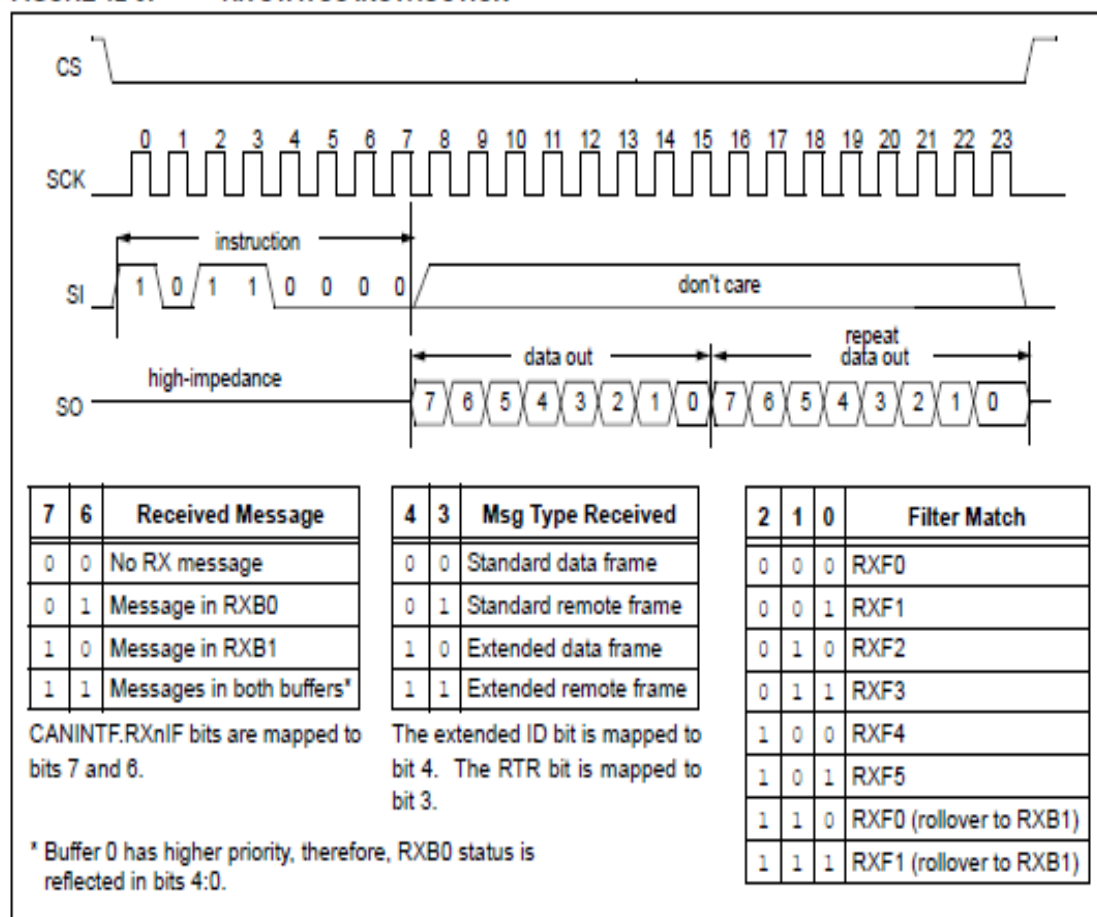


FIGURE 12-10: SPI INPUT TIMING

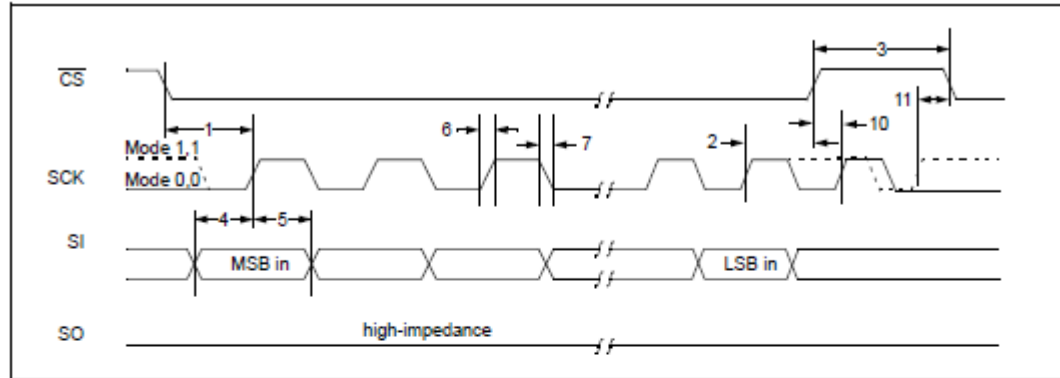
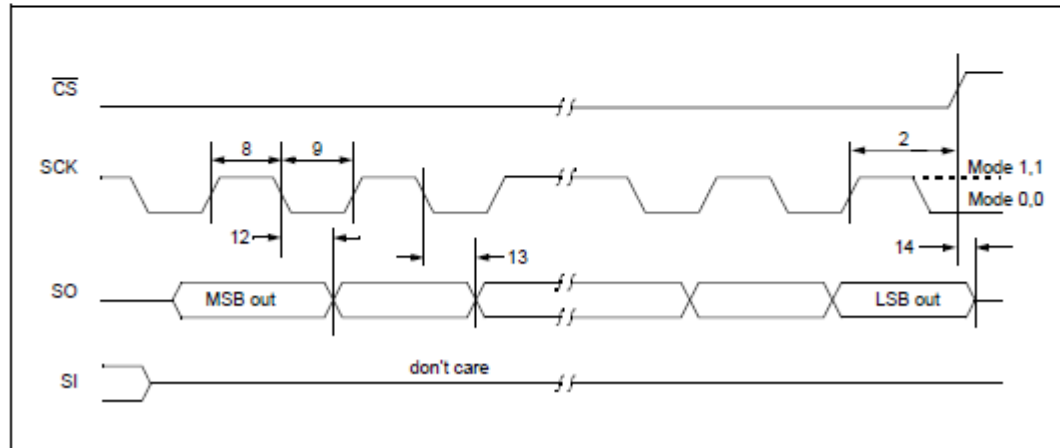


FIGURE 12-11: SPI OUTPUT TIMING



13.1 절대 최대 정격 †

VDD 7.0V

모든 입출력 w.r.t. VSS..... ~ 0.6V ~ VDD + 1.0V

보관 온도 - 65 ° C ~ + 150 ° C

주변 온도. 전원이 공급 된 상태에서 - 65 ° C ~ + 125 ° C

리드의 납땜 온도 (10 초) + 300 ° C

† notice : "최대 정격"에 나열된 것 이상으로 스트레스를 받으면 장치가 영구적으로 손상 될 수 있습니다.

이는 스트레스 등급으로, 본 사양의 작동 목록에 표시된 것 이상의 조건에서 장치의 기능 작동은 함축되어 있지 않습니다.

장시간 최대 정격 조건에 노출되면 장치의 신뢰성에 영향을 미칠 수 있습니다.

TABLE 13-1: DC CHARACTERISTICS

DC Characteristics			Industrial (I): Extended (E):		T _{AMB} = -40°C to +85°C T _{AMB} = -40°C to +125°C		V _{DD} = 2.7V to 5.5V V _{DD} = 4.5V to 5.5V
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions	
	V _{DD}	Supply Voltage	2.7	5.5	V		
	V _{RET}	Register Retention Voltage	2.4	—	V		
	V _{IH}	High-Level Input Voltage RXCAN SCK, $\overline{\text{CS}}$, SI, $\overline{\text{TXnRTS}}$ Pins OSC1 RESET	2 0.7 V _{DD} 0.85 V _{DD} 0.85 V _{DD}	V _{DD} + 1 V _{DD} + 1 V _{DD} V _{DD}	V V V V		
	V _{IL}	Low-Level Input Voltage RXCAN, $\overline{\text{TXnRTS}}$ Pins SCK, $\overline{\text{CS}}$, SI OSC1 RESET	-0.3 -0.3 V _{SS} V _{SS}	.15 V _{DD} 0.4 .3 V _{DD} .15 V _{DD}	V V V V		
	V _{OL}	Low-Level Output Voltage TXCAN $\overline{\text{RXnBF}}$ Pins SO, CLKOUT $\overline{\text{INT}}$	— — — —	0.6 0.6 0.6 0.6	V V V V	I _{OL} = +6.0 mA, V _{DD} = 4.5V I _{OL} = +8.5 mA, V _{DD} = 4.5V I _{OL} = +2.1 mA, V _{DD} = 4.5V I _{OL} = +1.6 mA, V _{DD} = 4.5V	
	V _{OH}	High-Level Output Voltage TXCAN, $\overline{\text{RXnBF}}$ Pins SO, CLKOUT $\overline{\text{INT}}$	V _{DD} - 0.7 V _{DD} - 0.5 V _{DD} - 0.7	— — —	V V V	I _{OH} = -3.0 mA, V _{DD} = 4.5V I _{OH} = -400 μ A, V _{DD} = 4.5V I _{OH} = -1.0 mA, V _{DD} = 4.5V	
	I _{LI}	Input Leakage Current All I/O except OSC1 and TXnRTS pins OSC1 Pin	-1 -5	+1 +5	μ A μ A	$\overline{\text{CS}} = \overline{\text{RESET}} = \text{V}_{DD}$, V _{IN} = V _{SS} to V _{DD}	
	C _{INT}	Internal Capacitance (All Inputs and Outputs)	—	7	pF	T _{AMB} = 25°C, f _C = 1.0 MHz, V _{DD} = 0V (Note 1)	
	I _{DD}	Operating Current	—	10	mA	V _{DD} = 5.5V, F _{OSC} = 25 MHz, F _{CLK} = 1 MHz, SO = Open	
	I _{DD5}	Standby Current (Sleep mode)	— —	5 8	μ A μ A	$\overline{\text{CS}}$, $\overline{\text{TXnRTS}} = \text{V}_{DD}$, Inputs tied to V _{DD} or V _{SS} , -40°C to +85°C $\overline{\text{CS}}$, $\overline{\text{TXnRTS}} = \text{V}_{DD}$, Inputs tied to V _{DD} or V _{SS} , -40°C to +125°C	

Note 1: This parameter is periodically sampled and not 100% tested.

NOTE 1 :이 매개 변수는 주기적으로 샘플링되며 100 % 테스트되지 않습니다.

TABLE 13-2: OSCILLATOR TIMING CHARACTERISTICS

Oscillator Timing Characteristics ^(Note)			Industrial (I): T _{AMB} = -40°C to +85°C		V _{DD} = 2.7V to 5.5V	
			Extended (E): T _{AMB} = -40°C to +125°C		V _{DD} = 4.5V to 5.5V	
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
	Fosc	Clock-In Frequency	1	40	MHz	4.5V to 5.5V
			1	25	MHz	2.7V to 5.5V
	Tosc	Clock-In Period	25	1000	ns	4.5V to 5.5V
			40	1000	ns	2.7V to 5.5V
	T _{DUTY}	Duty Cycle (External Clock Input)	0.45	0.55	—	T _{OSH} /(T _{OSH} + T _{OSL})

Note: This parameter is periodically sampled and not 100% tested.

Note :이 매개 변수는 주기적으로 샘플링되며 100 % 테스트되지 않습니다.

TABLE 13-3: CAN INTERFACE AC CHARACTERISTICS

CAN Interface AC Characteristics			Industrial (I): T _{AMB} = -40°C to +85°C		V _{DD} = 2.7V to 5.5V	
			Extended (E): T _{AMB} = -40°C to +125°C		V _{DD} = 4.5V to 5.5V	
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
	T _{WF}	Wake-up Noise Filter	100	—	ns	

TABLE 13-4: RESET AC CHARACTERISTICS

RESET AC Characteristics			Industrial (I): T _{AMB} = -40°C to +85°C		V _{DD} = 2.7V to 5.5V	
			Extended (E): T _{AMB} = -40°C to +125°C		V _{DD} = 4.5V to 5.5V	
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
	tr _L	RESET Pin Low Time	2	—	μs	

TABLE 13-5: CLKOUT PIN AC CHARACTERISTICS

CLKOUT Pin AC/DC Characteristics			Industrial (I): T _{AMB} = -40°C to +85°C		V _{DD} = 2.7V to 5.5V	
			Extended (E): T _{AMB} = -40°C to +125°C		V _{DD} = 4.5V to 5.5V	
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
	t _H CLKOUT	CLKOUT Pin High Time	15	—	ns	T _{OSC} = 40 ns (Note 1)
	t _L CLKOUT	CLKOUT Pin Low Time	15	—	ns	T _{OSC} = 40 ns (Note 1)
	t _r CLKOUT	CLKOUT Pin Rise Time	—	5	ns	Measured from 0.3 V _{DD} to 0.7 V _{DD} (Note 1)
	t _f CLKOUT	CLKOUT Pin Fall Time	—	5	ns	Measured from 0.7 V _{DD} to 0.3 V _{DD} (Note 1)
	t _d CLKOUT	CLOCKOUT Propagation Delay	—	100	ns	Note 1
15	t _H SO _F	Start-Of-Frame High Time	—	2 T _{OSC}	ns	Note 1
16	t _d SO _F	Start-Of-Frame Propagation Delay	—	2 T _{OSC} + 0.5 T _Q	ns	Measured from CAN bit sample point. Device is a receiver. CNF1.BRP<5:0> = 0 (Note 2)

Note 1: All CLKOUT mode functionality and output frequency is tested at device frequency limits, however, CLKOUT prescaler is set to divide by one. This parameter is periodically sampled and not 100% tested.

2: Design guidance only, not tested.

참고 1 : 모든 CLKOUT 모드 기능 및 출력 주파수는 장치 주파수 한계에서 테스트되지만 CLKOUT prescaler는 1로 분주하도록 설정됩니다.

이 매개 변수는 주기적으로 샘플링되며 100 % 테스트되지 않습니다.

2 : 설계 지침 만 테스트되었으며 테스트되지 않았습니다.

FIGURE 13-1: START-OF-FRAME PIN AC CHARACTERISTICS

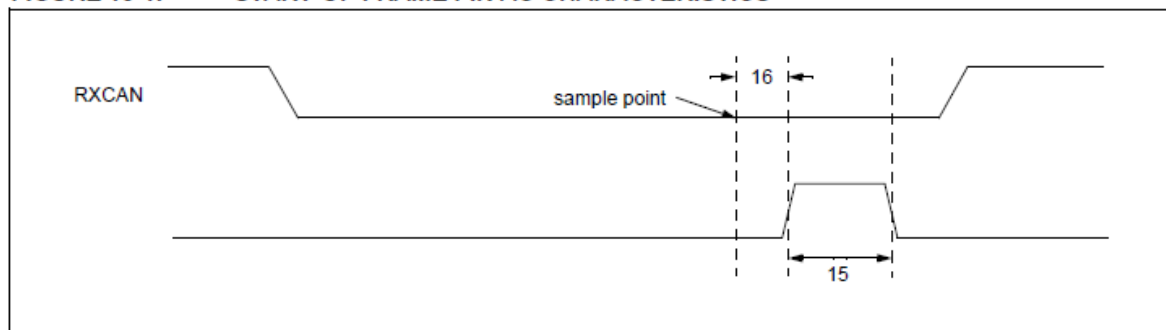


TABLE 13-6: SPI INTERFACE AC CHARACTERISTICS

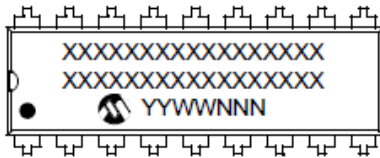
SPI Interface AC Characteristics			Industrial (I): T _{AMB} = -40°C to +85°C V _{DD} = 2.7V to 5.5V Extended (E): T _{AMB} = -40°C to +125°C V _{DD} = 4.5V to 5.5V			
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
	FCLK	Clock Frequency	—	10	MHz	
1	T _{css}	$\overline{\text{CS}}$ Setup Time	50	—	ns	
2	T _{csH}	$\overline{\text{CS}}$ Hold Time	50	—	ns	
3	T _{cSD}	$\overline{\text{CS}}$ Disable Time	50	—	ns	
4	T _{su}	Data Setup Time	10	—	ns	
5	T _{HO}	Data Hold Time	10	—	ns	
6	T _R	CLK Rise Time	—	2	μs	Note 1
7	T _F	CLK Fall Time	—	2	μs	Note 1
8	T _{HI}	Clock High Time	45	—	ns	
9	T _{LO}	Clock Low Time	45	—	ns ns	
10	T _{CLD}	Clock Delay Time	50	—	ns	
11	T _{CLE}	Clock Enable Time	50	—	ns	
12	T _V	Output Valid from Clock Low	—	45	ns	
13	T _{HO}	Output Hold Time	0	—	ns	
14	T _{DIS}	Output Disable Time	—	100	ns	

Note 1: This parameter is not 100% tested.

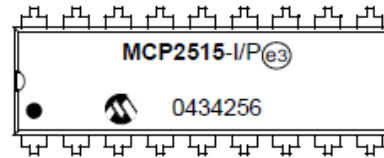
14.0 PACKAGING INFORMATION

14.1 Package Marking Information

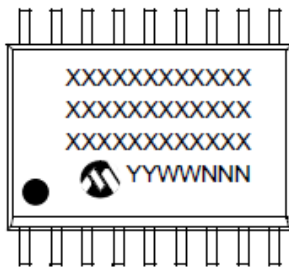
18-Lead PDIP (300 mil)



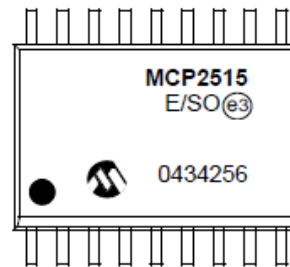
Example:



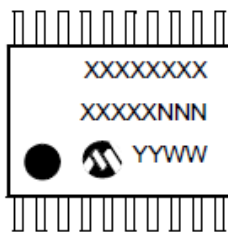
18-Lead SOIC (300 mil)



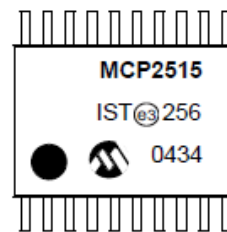
Example:



20-Lead TSSOP (4.4 mm)



Example:



범례 : XX ... X 고객 별 정보

Y 년 코드 (역년의 마지막 자리)

YY 연도 코드 (역년의 마지막 2 자리)

WW 주 코드 (1 월 1 일 주는 '01'주)

NNN 영숫자 추적 가능성 코드

(e3) 매트 주석 (Sn)을위한 무연 JEDEC 지정 부호

*이 패키지는 무연입니다. Pb-free JEDEC 부호 (e3)

이 패키지의 외부 포장에서 찾을 수 있습니다.

NOTE : 전체 마이크로 칩 부품 번호를 한 행에 표시 할 수없는 경우, 다음 행으로 이월되어 고객 별 정보에 사용 가능한 문자 수를 제한합니다.