

CANopen 네트워크 관리 (NMT)

처리 데이터의 전송과 디바이스의 구성을 위한 서비스와 프로토콜 외에도, 네트워크에 분산된 시스템의 운영은 개별 네트워크 노드의 통신 상태의 명령 제어를 위한 기능들이 필요합니다. CANopen 디바이스에 의한 데이터 전송이 많은 경우는 이벤트-제어(event-controlled)이기 때문에, 네트워크 노드의 통신 성능에 관한 꾸준한 모니터링 또한 필요합니다. CANopen 은 이러한 작업들을 위해 이른바 "네트워크 관리" 서비스와 프로토콜들을 제공합니다. 즉;

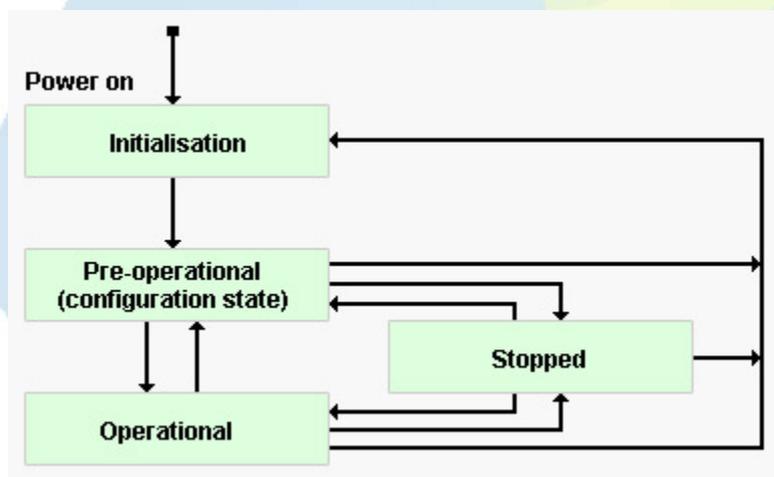
- 네트워크 노드의 통신 상태 제어와
- 노드 모니터링

CANopen 네트워크 노드의 상태와 NMT 메시지를 통한 상태 제어

CANopen 은 네트워크 노드의 통신 상태를 상황 도표로 설명합니다. 특정 CAN 메시지 (NMT 메시지)를 전송하여, 네트워크 관리 마스터는 CANopen 네트워크의 다른 노드들 (네트워크 관리 슬레이브)의 통신 상태를 제어합니다. 즉, 단일 명령으로 모든 노드들의 상태 또는 개별 노드의 상태 변경이 가능합니다

NMT 메시지는 가장 높은 우선 순위 메시지 식별자 (CAN-ID 0)로 전송됩니다. 데이터 필드는 단 두 개의 바이트들로 구성됩니다: 첫번째 데이터 바이트에는 필요한 타겟 상태가 코드화되며, 두 번째 데이터 바이트는 통신 상태로 변경될 노드의 번호를 지정합니다. 네트워크의 모든 노드들은 공통적으로 가상 노드-ID 0 과 함께 어드레스됩니다. 이런 방식으로, 예를 들면, 모든 노드들은 동시에 작동을 시작할 수 있도록 한꺼번에 "Operational" 상태로 설정될 수 있습니다.

상태(state) 그리고 상태 변경





특정 노드의 부분적인 리셋도 가능하게 하기 위해서, 이 상태는 3 개의 하위-상태(sub-state)들로 다시 분할됩니다: "Reset-Application", "Reset-Communication", "Initializing".

하드웨어 리셋 또는 전원을 켜 후에는, 노드가 초기화("Initializing") 상태가 됩니다. 기본적인 노드 초기화 완료 후에 (예, 호스트 컨트롤러, CAN 컨트롤러, 애플리케이션 소프트웨어 등), 노드는 이른바 "boot-up 메시지"를 전송하며, "Pre-operational" 상태로 변환됩니다. "Reset-Application" 상태에서, 제조업체-특정 파라미터들과 표준 디바이스 프로파일의 파라미터들이 그들의 Power-On 값들로 리셋됩니다 (최종적으로 저장된 값들과 일치). 그러면 노드는 "Reset-Communication" 상태로 변경됩니다. "Reset-Communication" 상태에서, 통신 프로파일의 파라미터들이 Power-On 값들로 리셋됩니다. 그런 후 노드 상태는 "Initializing" 로 변경됩니다.

Pre-Operational

이 상태는 주로 CANopen 디바이스의 구성에 사용됩니다. 따라서 처리 데이터의 교환 (PDO 를 통한)은 이 상태에서는 가능하지 않습니다. 디바이스 객체 사전(object dictionary)의 엔트리들은 "service data objects" (SDO)를 통해 액세스할 수 있습니다. SDO 메시지를 전송함으로써, 특정 디바이스의 객체 사전은 configuration 툴 등을 이용하여 수정될 수 있습니다.

SDO 메시지를 통한 통신 외에도, emergency, synchronization, time stamp 와 NMT 제어 메시지들은 당연히 Pre-operational 상태에서 전송되거나 수신될 수 있습니다. "Start-Remote-Node"를 전송하여, 노드는 "Operational" 상태로 변환됩니다.

Operational

이 상태에서는, 이른바 "process data object" (PDO) 를 통한 처리 데이터의 전송이 가능합니다.

Stopped

Node guarding 또는 heartbeat 메시지 이외에는, 이 상태에서 노드가 어떤 다른 메시지들을 전송하거나 수신할 수 없습니다.

Node guarding 과 heartbeat 메커니즘을 사용한 CANopen Device 모니터링

확실한 네트워크 노드의 작동을 위해, CANopen 은 두 가지 대안을 제공합니다:

- 소위 "NMT master" 로 불리는, 상위 order instance 에 의한 노드 상태의 주기적 조회 ("node guarding" 원칙) 또는
- 네트워크 노드에 의한 "heartbeat message"의 자동 전송 ("heartbeat 원칙)

디바이스 통신 상태를 모니터링하기 위해서는 노드당 한 개의 CAN-ID 가 필요합니다. 이를 위해 node-ID+ 1792 의 값을 지닌 낮은 우선순위 메시지 식별자들이 준비되어 있습니다.



Node guarding 에서, NMT-master 는 toggle-bit 와 함께 이것의 현재 통신 상태 (stopped, operational, pre-operational) 를 포함한 데이터 전보를 전송하기 위해 정의된 간격에서 차례로 CAN 원격 프레임을 이용하여 네트워크의 다른 노드들 (따라서 NMT-slave 로 일컬어짐) 을 요청합니다. 만약 노드가 특정 시간 (node life time)동안 NMT-master 의 요청에 응답하지 않으면, 이는 노드의 실패로 평가되고 "node-guarding event" 로 NMT-master 의 호스트 컨트롤러에 통보됩니다. 한편, NMT-slave 역시 그들의 "life time" 동안 NMT-master 로부터 요청을 받았는지 관찰합니다. 만약 그와 같은 요청이 노드의 이른바 "life time factor"보다 긴 시간 동안 없었다면, NMT-slave 는 NMT-master 자체가 실패라 여기고 이 사건을 "life guarding event"로 자신의 호스트 컨트롤러에 알립니다.

Heartbeat 원칙에 따른 노드 모니터링은, 통신 능력의 증거로써 노드가 자신의 통신 상태를 일정한 간격마다 자동으로 전송합니다. 소위 heartbeat producer 의 두 개의 heartbeat 메시지들 사이의 간격 ("heartbeat interval")은 object directory entry [1017]를 통해 구성됩니다. 0 의 값은 heartbeat 메커니즘을 무효화합니다. 최대 127 개 네트워크 노드들의 이른바 "heartbeat consumer time" 은 OD entry [1016]에 나와 있습니다. 이 시간 간격은 모니터링 노드에서 heartbeat 메시지의 도착을 위한 최대 시간 주기를 설명합니다.

요즘은 node guarding 이 더 이상 사용되지 않습니다. 이것은 대개는 높은 버스 적재 (모니터링 간격마다 2 개의 CAN 메시지들 때문) 때문이기도 하지만, NMT-master 의 crucial node healthy survey 의 바람직하지 못한 집중화를 가져올 수도 있기 때문입니다..