

빠른 시작: MXCHIP AZ3166 DevKit를 IoT Central에 연결

2021. 06. 02. • 읽는 데 17분 걸림 •  

이 문서의 내용

[사전 요구 사항](#)

[개발 환경 준비](#)

[클라우드 구성 요소 만들기](#)

[디바이스 준비](#)

[디바이스 상태 확인](#)

[원격 분석 보기](#)

[디바이스에서 직접 메서드 호출](#)

[디바이스 정보 보기](#)

[문제 해결 및 디버그](#)

[리소스 정리](#)

[다음 단계](#)

적용 대상: [임베디드 디바이스 개발](#)

총 완료 시간: 30분



이 빠른 시작에서는 Azure RTOS를 사용하여 MXCHIP AZ3166 IoT DevKit(이하 MXCHIP DevKit)를 Azure IoT에 연결합니다.

다음 작업을 완료합니다.

- C에서 MXCHIP DevKit를 프로그래밍하기 위한 임베디드 개발 도구 세트 설치
- 이미지를 빌드하고 MXCHIP DevKit로 플래시
- Azure IoT Central을 사용하여 클라우드 구성 요소 만들기, 속성 보기, 디바이스 원격 분석 보기 및 직접 명령 호출

사전 요구 사항

- Microsoft Windows 10을 실행하는 PC
- 리포지토리를 복제하기 위한 [Git](#)
- 하드웨어

- [MXCHIP AZ3166 IoT DevKit](#) (MXCHIP DevKit)
- Wi-Fi 2.4GHz
- USB 2.0 A~마이크로 USB 수 케이블


개발 환경 준비

개발 환경을 설정하려면 먼저 빠른 시작에 필요한 모든 자산이 포함되어 있는 GitHub 리포지토리를 복제합니다. 그런 다음, 프로그래밍 도구 세트를 설치합니다.

빠른 시작에 사용할 리포지토리를 복제합니다.

모든 샘플 디바이스 코드, 설치 스크립트 및 오프라인 버전 설명서를 다운로드하려면 다음 리포지토리를 복제합니다. 이전에 다른 빠른 시작에서 이 리포지토리를 복제한 경우 다시 복제할 필요가 없습니다.

리포지토리를 복제하려면 다음 명령을 실행합니다.

shell	 복사
<pre>git clone --recursive https://github.com/azure-rtos/getting-started.git</pre>	

도구 설치

복제된 리포지토리에는 필요한 도구를 설치하고 구성하는 설치 스크립트가 포함되어 있습니다. 다른 포함된 디바이스 빠른 시작에서 이 도구를 설치한 경우 다시 설치할 필요가 없습니다.

❗ 참고

설치 스크립트에서 설치하는 도구는 다음과 같습니다.

- **CMake** : 빌드
- **ARM GCC** : 컴파일
- **Termite** : 연결된 디바이스의 직렬 포트 출력 모니터링

도구를 설치하려면

1. 파일 탐색기에서 리포지토리의 다음 경로로 이동하여 *get-toolchain.bat* 이라는 설치 스크립트를 실행합니다.

getting-started\tools\get-toolchain.bat

2. 설치 후 새 콘솔 창을 열어 설치 스크립트에서 변경한 구성을 인식합니다. 이 콘솔을 사용하여 빠른 시작의 나머지 프로그래밍 작업을 완료합니다. Windows용 Windows CMD, PowerShell 또는 Git Bash를 사용할 수 있습니다.
3. 다음 코드를 실행하여 CMake 버전 3.14 이상이 설치되어 있는지 확인합니다.

```
shell
```

[복사](#)

```
cmake --version
```

클라우드 구성 요소 만들기

IoT Central 애플리케이션 만들기

디바이스를 Azure IoT에 연결하는 방법에는 여러 가지가 있습니다. 이 섹션에서는 Azure IoT Central을 사용하여 디바이스를 연결하는 방법에 대해 알아봅니다. IoT Central은 IoT 솔루션을 만들고 관리하는 데 드는 비용과 복잡성을 줄이는 IoT 애플리케이션 플랫폼입니다.

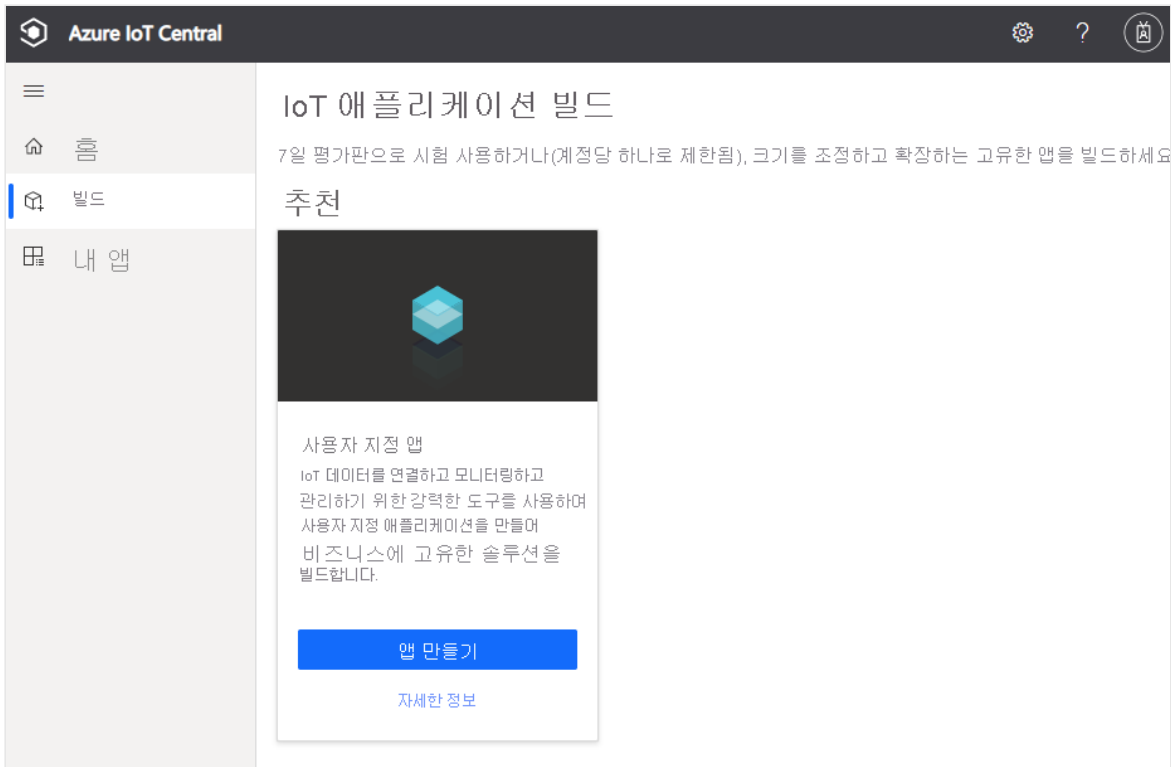
새 애플리케이션을 만들려면

1. [Azure IoT Central 포털](#)의 측면 탐색 메뉴에서 **빌드**를 선택합니다.

❗ 참고

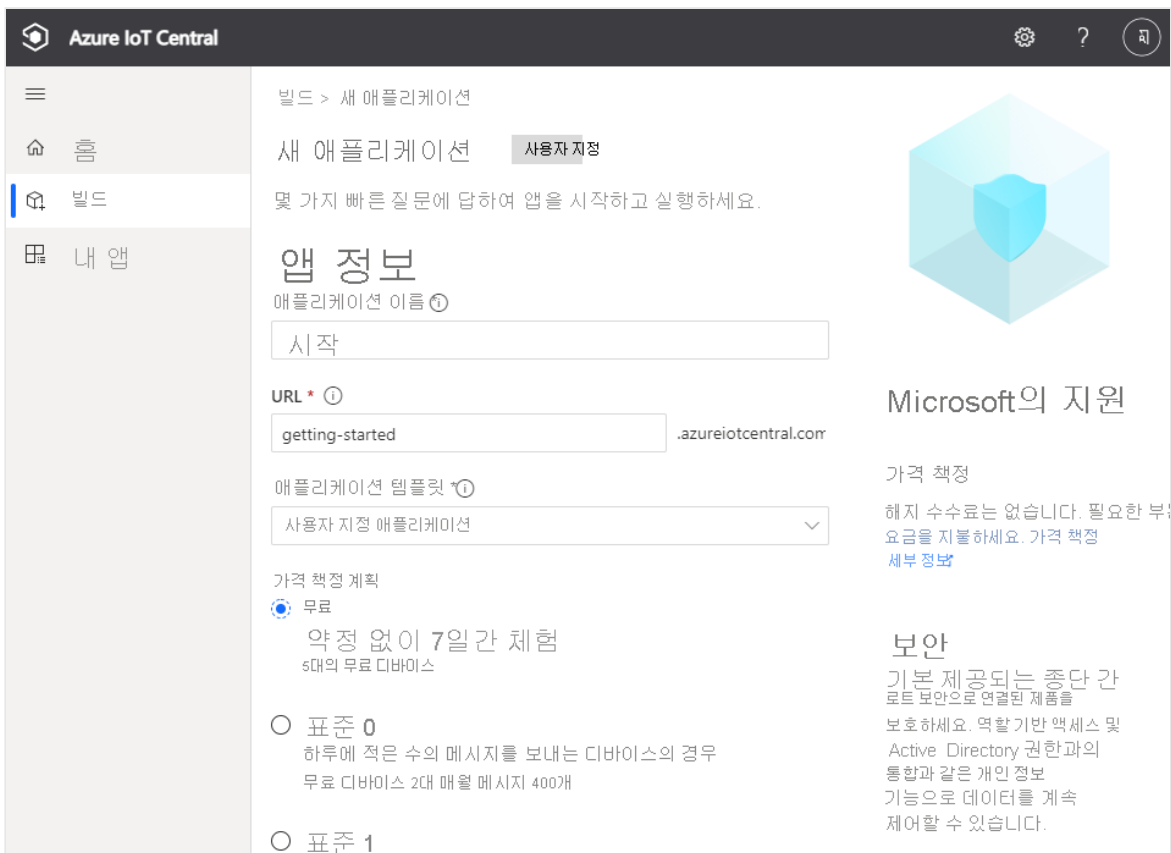
기존 IoT Central 애플리케이션이 있는 경우 이를 사용하여 새 애플리케이션을 만드는 대신 이 문서의 단계를 완료할 수 있습니다. 이 경우 기존 디바이스 ID를 사용하려는 경우 새 디바이스를 만들거나 디바이스를 삭제한 후 다시 만드는 것이 좋습니다.

2. **사용자 지정 앱** 타일에서 **앱 만들기**를 선택합니다.



3. 애플리케이션 이름 및 URL을 추가합니다.

4. **체험** 가격 책정 계획을 선택하여 7일 평가판을 활성화합니다.



5. **만들기** 를 선택합니다. IoT Central에서 애플리케이션을 프로비저닝하면 자동으로 새 애플리케이션 대시보드로 리디렉션됩니다.

새 디바이스 만들기

이 섹션에서는 IoT Central 애플리케이션 대시보드를 사용하여 새 디바이스를 만듭니다. 이후 섹션에서 새로 만든 디바이스에 대한 연결 정보를 사용하여 물리적 디바이스를 안전하게 연결합니다.

디바이스를 만들려면

1. 애플리케이션 대시보드의 측면 탐색 메뉴에서 **디바이스** 를 선택합니다.
2. **모든 디바이스** 창에서 **디바이스 만들기** 를 선택하여 **새 디바이스 만들기** 창을 엽니다. 이미 하나 이상의 디바이스가 있는 기존 애플리케이션을 다시 사용하는 경우 **+ 새로 만들기** 를 선택하여 창을 엽니다.
3. 디바이스 템플릿을 **할당되지 않음** 으로 둡니다.
4. 원하는 디바이스 이름 및 디바이스 ID를 입력합니다.

새 디바이스 만들기

×

새 디바이스를 만들려면 디바이스 템플릿, 이름 및 고유 ID를 선택합니다. 자세한 [정보](#)

디바이스 이름 *

?

mydevice

디바이스 ID *

?

mydevice

디바이스 템플릿 *

할당되지 않음

이 디바이스를 시뮬레이트하시겠습니까?

시뮬레이트된 디바이스는 실제 디바이스를 연결하기 전에 애플리케이션의 동작을 테스트할 수 있는 원격 분석을 생성합니다.

아니요

만들기

취소

5. **만들기** 단추를 선택합니다.
6. **모든 디바이스** 목록에 새로 만든 디바이스가 표시됩니다. 디바이스 이름을 선택하여 세부 정보를 표시합니다.
7. 오른쪽 위의 메뉴 모음에서 **연결** 을 선택하여 다음 섹션에서 디바이스를 구성하는 데 사용되는 연결 정보를 표시합니다.

디바이스 연결

ID 범위 ⓘ

디바이스 ID ⓘ

mydevice

이 디바이스에 대한 연결 형식을 선택합니다. 필요한 경우 나중에 변경할 수 있습니다.

인증 유형

공유 액세스 서명(SAS) ▼

SAS(공유 액세스 서명)는 보안 토큰과 키를 사용하여 IoT Central에 연결합니다. 기본 등록 그룹의 SAS 키를 사용하여 디바이스를 등록하세요. 자세한 정보

기본 키 ⓘ

보조 키 ⓘ

닫기

8. **연결** 대화 상자에 표시된 다음 연결 문자열 매개 변수에 대한 연결 값을 적어 둡니다. 이러한 값은 다음 단계에서 구성 파일에 추가합니다.

- ID scope
- Device ID
- Primary key

디바이스 준비

MXCHIP DevKit를 Azure에 연결하려면 Wi-Fi 및 Azure IoT 설정에 대한 구성 파일을 수정하고, 이미지를 다시 빌드하고, 해당 이미지를 디바이스로 플래시합니다.

구성 추가

1. 텍스트 편집기에서 다음 파일을 엽니다.

getting-started\MXChip\AZ3166\app\azure_config.h

2. Wi-Fi 상수를 로컬 환경의 다음 값으로 설정합니다.

상수 이름	값
WIFI_SSID	{사용자의 Wi-Fi SSID}
WIFI_PASSWORD	{사용자의 Wi-Fi 암호}
WIFI_MODE	{파일에 열거된 Wi-Fi 모드 값 중 하나}

3. Azure IoT 디바이스 정보 상수를 Azure 리소스를 만든 후에 저장한 값으로 설정합니다.

상수 이름	값
IOT_DPS_ID_SCOPE	{사용자의 ID 범위 값}
IOT_DPS_REGISTRATION_ID	{사용자의 디바이스 ID 값}
IOT_DEVICE_SAS_KEY	{사용자의 기본 키 값}

4. 파일을 저장하고 닫습니다.

이미지 빌드

1. 콘솔 또는 파일 탐색기에서 다음 경로에 있는 *rebuild.bat* 스크립트를 실행하여 이미지를 빌드합니다.

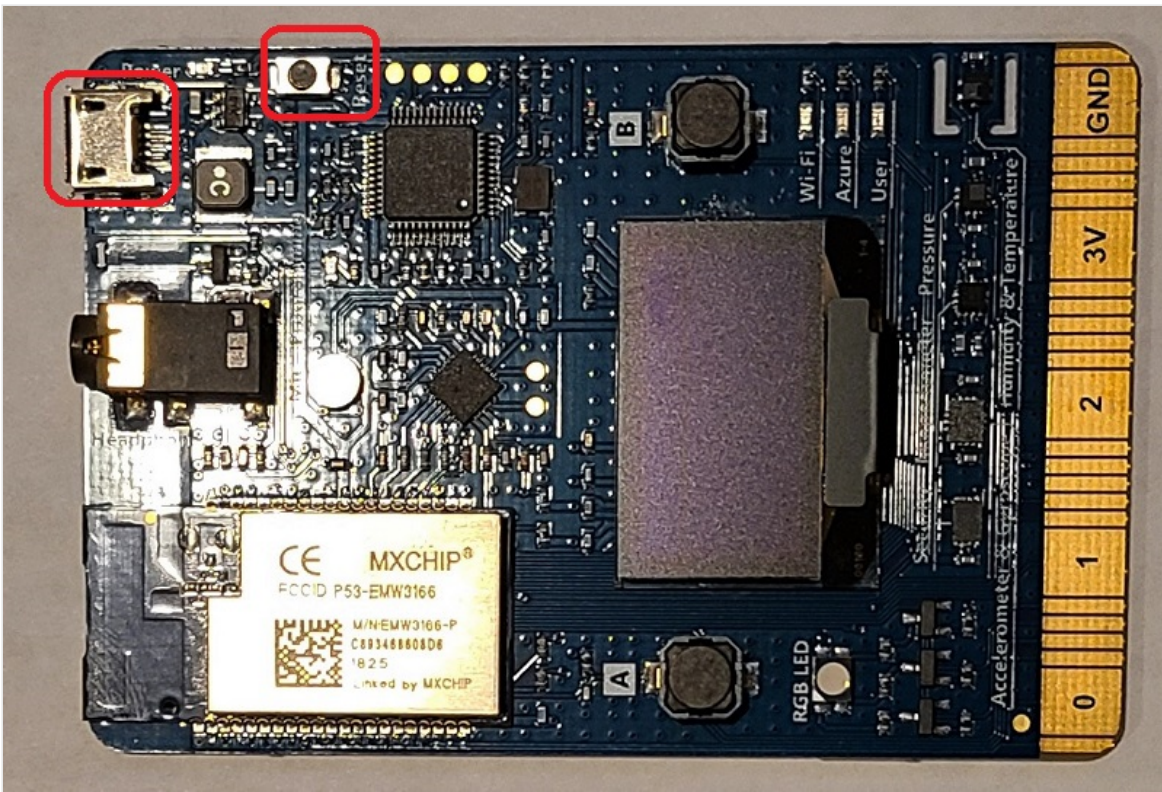
getting-started\MXChip\AZ3166\tools\rebuild.bat

2. 빌드가 완료되면 이진 파일이 다음 경로에 만들어졌는지 확인합니다.

getting-started\MXChip\AZ3166\build\app\mxchip_azure_iot.bin

이미지 플래시

1. MXCHIP DevKit에서 **다시 설정** 단추 및 마이크로 USB 포트를 찾습니다. 이러한 구성 요소는 다음 단계에서 사용합니다. 다음 그림에는 두 가지 구성 요소가 모두 강조 표시되어 있습니다.



2. 마이크로 USB 케이블을 MXCHIP DevKit의 마이크로 USB 포트에 연결한 다음, 컴퓨터에 연결합니다.
3. 파일 탐색기에서 이전 섹션에서 만든 이진 파일을 찾습니다.
4. `mxchip_azure_iot.bin` 이진 파일을 복사합니다.
5. 파일 탐색기에서 컴퓨터에 연결된 MXCHIP DevKit 디바이스를 찾습니다. 디바이스는 드라이브 레이블이 **AZ3166** 인 드라이브로 시스템에 표시됩니다.
6. 이진 파일을 MXCHIP DevKit의 루트 폴더에 붙여넣습니다. 깜박임이 자동으로 시작되고 몇 초 후에 완료됩니다.

❗ 참고

깜박임 프로세스 중에 녹색 LED에서 MXCHIP DevKit를 컵니다.

디바이스 연결 세부 정보 확인

Termite 앱을 사용하여 통신을 모니터링하고 디바이스가 올바르게 설정되었는지 확인할 수 있습니다.

1. **Termite** 를 시작합니다.

💡 팁

Termite를 DevKit에 연결할 수 없는 경우 **ST-LINK 드라이버** 를 설치하고 다시 시도합니다. 추가 단계는 **문제 해결**을 참조하세요.

2. **설정** 을 선택합니다.

3. **직렬 포트 설정** 대화 상자에서 다음 설정을 확인하고, 필요한 경우 업데이트합니다.

- **전송 속도**: 115,200
- **포트**: MXCHIP DevKit가 연결된 포트입니다. 드롭다운에 여러 포트 옵션이 있는 경우 사용할 올바른 포트를 찾을 수 있습니다. Windows **디바이스 관리자** 를 열고, **포트** 를 확인하여 사용할 포트를 식별합니다.

4. **확인**을 선택합니다.

5. 디바이스의 **다시 설정** 단추를 누릅니다. 이 단추는 디바이스에 대한 레이블을 표시하고 마이크로 USB 커넥터 근처에 있습니다.

6. **Termite** 앱에서 다음 검사점 값을 확인하여 디바이스가 초기화되고 Azure IoT에 연결되었는지 확인합니다.

출력	복사
<pre>Starting Azure thread Initializing WiFi MAC address: C8:93:46:8A:4C:43 Connecting to SSID 'iot' SUCCESS: WiFi connected to iot Initializing DHCP IP address: 192.168.0.18 Mask: 255.255.255.0 Gateway: 192.168.0.1 SUCCESS: DHCP initialized</pre>	

```

Initializing DNS client
  DNS address: 75.75.75.75
SUCCESS: DNS client initialized

Initializing SNTP client
  SNTP server 0.pool.ntp.org
  SNTP IP address: 38.229.71.1
  SNTP time update: May 19, 2021 20:36:6.994 UTC
SUCCESS: SNTP initialized

Initializing Azure IoT DPS client
  DPS endpoint: global.azure-devices-provisioning.net
  DPS ID scope: ***
  Registration ID: mydevice
SUCCESS: Azure IoT DPS client initialized

Initializing Azure IoT Hub client
  Hub hostname: ***.azure-devices.net
  Device id: mydevice
  Model id: dtmi:azurertos:devkit:gsgmxchip;1
Connected to IoT Hub
SUCCESS: Azure IoT Hub client initialized

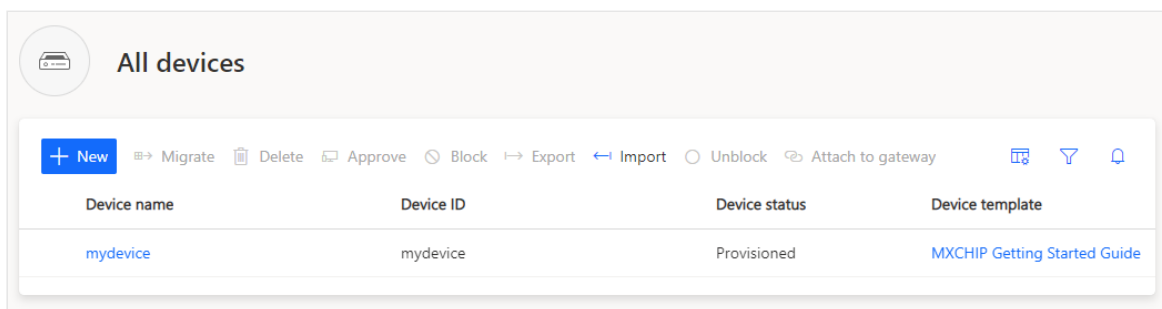
```

다음 단계에서 디바이스 출력을 모니터링하려면 Termite를 열어 둡니다.

디바이스 상태 확인

IoT Central 포털에서 디바이스 상태를 보려면

1. 애플리케이션 대시보드의 측면 탐색 메뉴에서 **디바이스** 를 선택합니다.
2. **디바이스 상태** 가 **프로비저닝됨** 으로 업데이트되었는지 확인합니다.
3. **디바이스 템플릿** 이 **MXCHIP 시작 가이드** 로 업데이트되었는지 확인합니다.

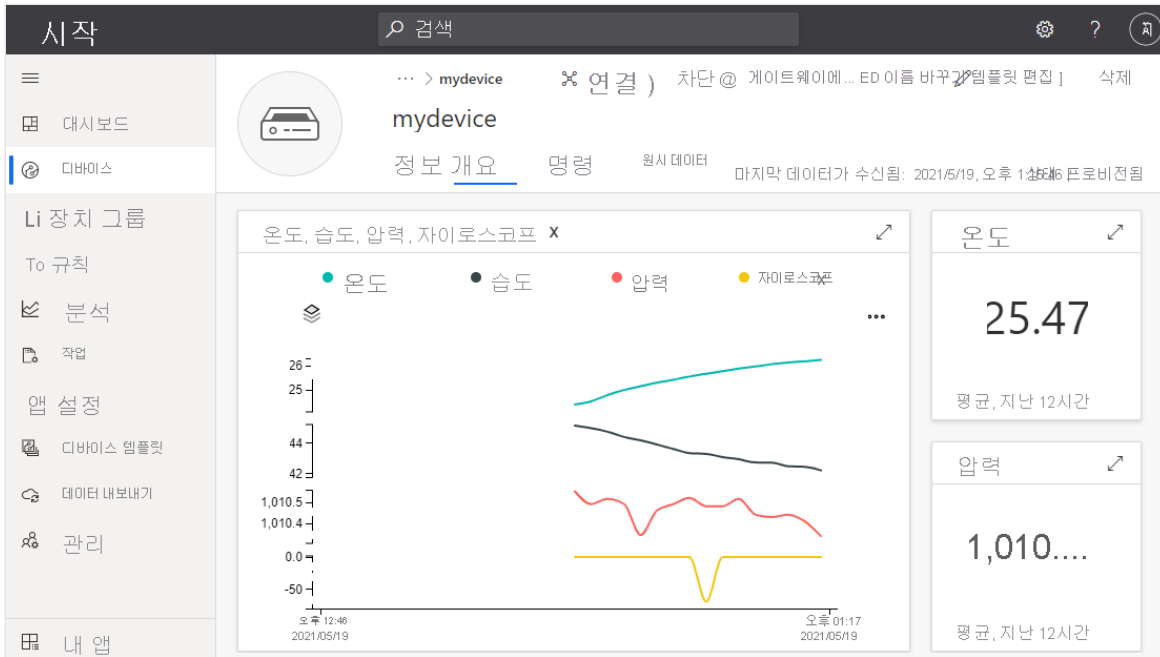


원격 분석 보기

IoT Central을 사용하면 디바이스에서 클라우드로의 원격 분석 흐름을 볼 수 있습니다.

IoT Central 포털에서 원격 분석을 보려면

1. 애플리케이션 대시보드의 측면 탐색 메뉴에서 **디바이스** 를 선택합니다.
2. 디바이스 목록에서 디바이스를 선택합니다.
3. **개요** 탭에서 디바이스에서 메시지를 클라우드에 보낼 때 원격 분석을 봅니다.



❗ 참고

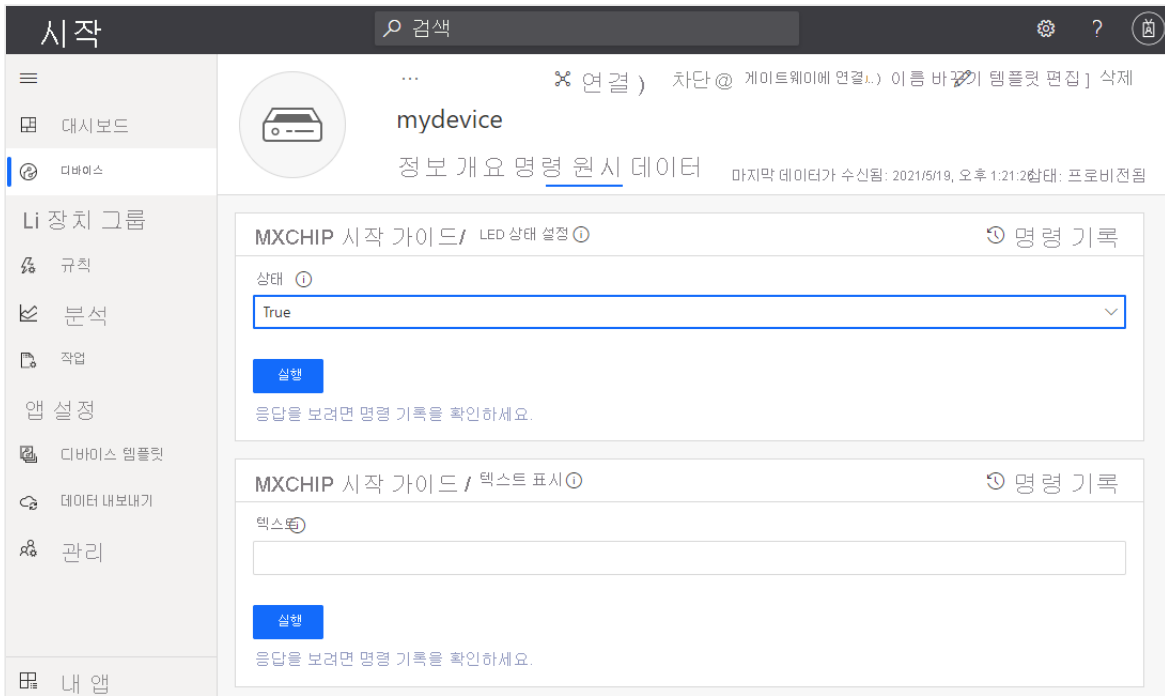
Termite 앱을 사용하여 디바이스에서 원격 분석을 모니터링할 수도 있습니다.

디바이스에서 직접 메서드 호출

IoT Central을 사용하여 디바이스에서 구현한 직접 메서드를 호출할 수도 있습니다. 직접 메서드에는 이름이 있으며, 필요에 따라 JSON 페이로드, 구성 가능한 연결 및 메서드 시간 제한이 있을 수 있습니다. 이 섹션에서는 LED를 설정하거나 해제하는 데 사용할 수 있는 메서드를 호출합니다.

IoT Central 포털에서 메서드를 호출하려면

1. 디바이스 페이지에서 **명령** 탭을 선택합니다.
2. **상태** 드롭다운에서 **True** 를 선택한 다음, **실행** 을 선택합니다. LED 표시등이 켜집니다.



3. **상태** 드롭다운에서 **False** 를 선택한 다음, **실행** 을 선택합니다. LED 표시등이 꺼집니다.

디바이스 정보 보기

IoT Central에서 디바이스 정보를 볼 수 있습니다.

디바이스 페이지에서 **정보** 탭을 선택합니다.



문제 해결 및 디버깅

디바이스 코드 빌드, 디바이스 플래시 또는 연결에 문제가 발생하는 경우 [문제 해결](#)을 참조하세요.

애플리케이션 디버깅에 대한 자세한 내용은 [Visual Studio Code를 사용하여 디버깅](#) 을 참조하세요.

리소스 정리

이 빠른 시작에서 만든 Azure 리소스가 더 이상 필요하지 않은 경우 IoT Central 포털에서 해당 리소스를 삭제할 수 있습니다.

전체 Azure IoT Central 샘플 애플리케이션과 모든 해당 디바이스 및 리소스를 제거하려면 다음을 수행합니다.

1. **관리 > 애플리케이션** 을 차례로 선택합니다.
2. **삭제** 를 선택합니다.

다음 단계

이 빠른 시작에서는 Azure RTOS 샘플 코드가 포함된 사용자 지정 이미지를 빌드한 다음, 해당 이미지를 MXCHIP DevKit 디바이스로 플래시했습니다. 또한 IoT Central 포털을 사용하여 Azure 리소스를 만들고, MXCHIP DevKit를 Azure에 안전하게 연결하고, 원격 분석을 보고, 메시지를 보냈습니다.

다음 단계로, 다음 문서에서 IoT 디바이스 SDK를 사용하여 디바이스를 Azure IoT에 연결하는 방법을 자세히 살펴봅니다.

IoT Hub에 MXCHIP AZ3166 devkit 연결

IoT Central에 시뮬레이션된 디바이스 연결

IoT Hub에 시뮬레이션된 디바이스 연결

① 중요

Azure RTOS는 기본 MCU/MPU 하드웨어 보호 메커니즘을 사용하여 통신을 보호하고 코드 및 데이터 격리를 만드는 구성 요소를 OEM에 제공합니다. 그러나 각 OEM은 궁극적으로 디바이스에서 진화하는 보안 요구 사항을 충족하는지 확인해야 합니다.

이 페이지가 도움이 되었나요?

 Yes  No

권장 콘텐츠

Azure IoT Central 에너지 솔루션이란?

Azure IoT Central 애플리케이션 템플릿을 사용하여 에너지 솔루션을 빌드하는 방법을 알아 봅니다.

Azure 산업용 IoT 개요

이 문서는 산업용 IoT에 대한 개요를 제공합니다. IoT의 연결된 공장, 작업 현장 연결성 및 보안 구성 요소에 대해 설명합니다.

OPC 쌍이란? - Azure

이 문서는 OPC 쌍에 대한 개요를 제공합니다. OPC 쌍은 REST API를 통한 산업용 디바이스의 검색, 등록 및 원격 제어를 제공합니다.

Azure IoT Central에서 데이터 내보내기

새 데이터 내보내기를 사용하여 IoT 데이터를 Azure 및 사용자 지정 클라우드 대상으로 내보내는 방법입니다.

자세히 표시 ▼