

# M-AR0230-DUAL-CAM-UVC 사용설명서 KOR

REVISION 1.1.0

MGSG CO.,LTD

## Revision history

Revision	Date	Description	Update by
<b>1.1.0</b>	2019.06.01	Initial creation	jhyoo
	2019.06.13	Update	jhyoo
	2019.10.03	Update	jhyoo
	2019.10.11	Update	jhyoo

## Table of Contents

- 1. Package**
- 2. Overview**
- 3. Examples**
- 4. Support**

## 1. Package

### 1.1 내용물



- M-AR0230-DUAL-CAM-UVC 보드
- Tripod 와 고정용 nut
- USB3.0 케이블 AM-BM 1.2meter 1EA

Figure 1.1.1 M-AR0230-DUAL-CAM-UVC 내용물

## 2. Overview

M-AR0230-DUAL-CAM-UVC 는 AR0230 이미지센서 두개가 장착된 카메라 개발보드입니다.  
 다양한 해상도로 설정 가능하며, 영상 저장 및 실시간 영상처리를 위한 용도로 사용 할 수 있습니다.  
 2개의 이미지센서 영상을 UVC로 출력하는, 다운로드 가능한 binary 파일과  
 회로도 및 USB3.0 영상 전송 예제(FPGA generated pattern image) 소스코드가 제공됩니다.

### 2.1 보드 구성

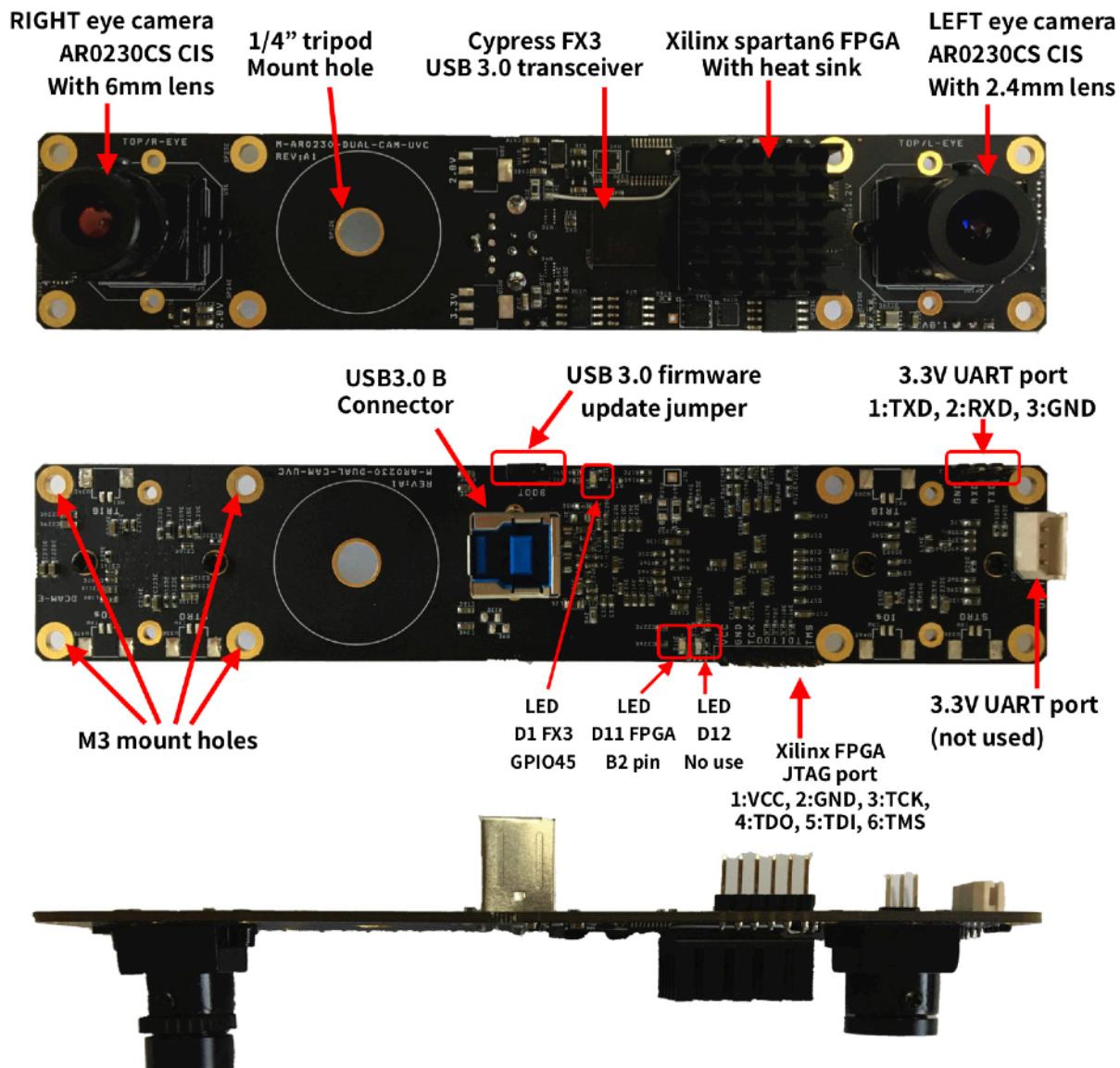


Figure 2.1.1 M-AR0230-DUAL-CAM-UVC 보드 구성

## 2.2 보드 크기

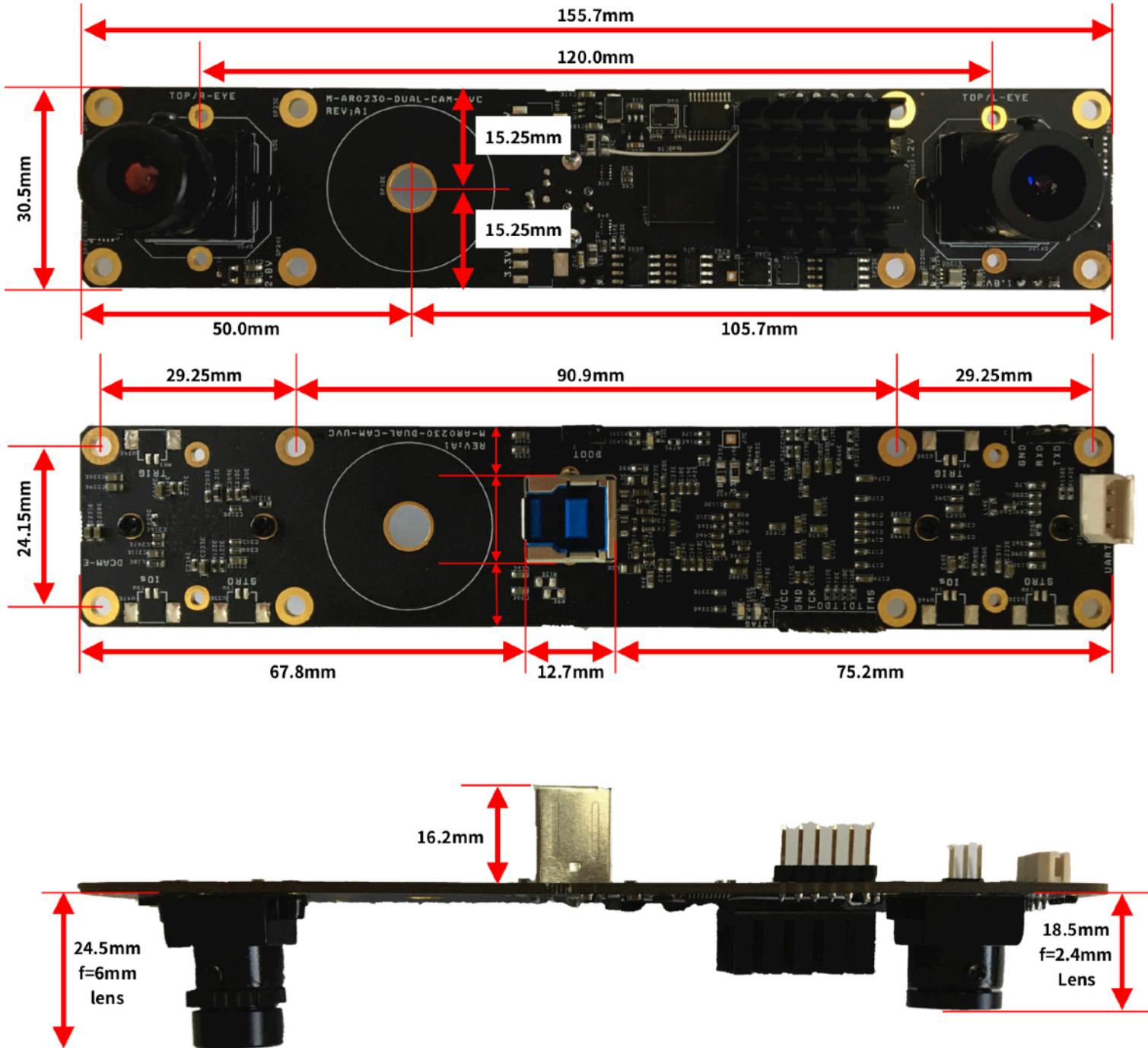


Figure 2.2.2 M-AR0230-DUAL-CAM-UVC 보드 치수

## 2.3 사양

- 기본 동작(구매 시 다운로드 되어있는 펌웨어)
  - UVC camera, 3840x1080@30fps, uncompressed color YUY2 format(binary 파일 제공)
  - 지원 가능 해상도 펌웨어, uncompressed color YUY2 format(binary 파일 제공)
  - UVC camera, 3840x1080@30fps(2 camera 1920x1080 해상도)
    - AR0230 sensor size : 5.76x3.24mm(sensor pixel size 3um)
  - UVC camera, 1920x540@30fps(2 camera 960x540 해상도)
    - AR0230 sensor size : 5.76x3.24mm(sensor pixel size 3um)
  - UVC camera, 2560x720@30fps(2 camera 1280x720 해상도)
    - AR0230 sensor size : 3.84x2.16mm(sensor pixel size 3um)
  - 기타 해상도 : 가로방향 1920 이하, 세로방향 1080 이하
- USB 3.0 super speed
  - USB 3.0 B connector
- 전원 입력
  - DC 5V USB bus power
- FPGA
  - XC6SLX16-2FTG256C , Xilinx Spartan6
- 두개의 CIS(CMOS image sensor)
  - AR0230CS, ONSEMI(Aptina) 1920x1080 CIS(CMOS image sensor)
- 두개의 M12(지름 12mm) 렌즈 홀더
- 두개의 M12 렌즈
  - 고정 초점거리 2.4mm, 6mm 혹은 유사한 초점거리(IR 차단 필터 있음), 추후 변경 가능
- Xilinx JTAG 연결 단자
  - 2.54mm 간격 6핀 male 커넥터(1:VCC, 2:GND, 3:TCK, 4:TDO, 5:TDI, 6:TMS)
- UART 단자
  - 2.54mm 간격 3핀 male 커넥터
    - 1 : TXD, 2: RXD, 3: GND
- LEDs
  - D1 : Blue LED, FX3 GPIO 45에 연결됨
  - D11 : Blue LED, FPGA B2 pin에 연결됨
  - D12 : no use

## 3. Examples

### 3.1 시스템 요구 및 기본사항

- Github 자료 링크 : <https://github.com/mgsgo/M-AR0230-DUAL-CAM-UVC>
  - 예제, 회로도 등
- Xilinx Spartan6 개발툴
  - Xilinx ISE 14.7
    - [https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/design-tools/v2012\\_4---14\\_7.html](https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/design-tools/v2012_4---14_7.html)
    - [https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/design-tools/14\\_7-windows.html](https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/design-tools/14_7-windows.html)
- Cypress FX3 SDK 1.3.4 개발툴
  - <https://www.cypress.com/documentation/software-and-drivers/ez-usb-fx3-software-development-kit>

### 3.2 영상 전송 기본 펌웨어 3840x1080@30fps 2camera 1920x1080 동작

- 기본 다운로드되어있는 3840x1080 해상도 카메라 영상 전송 예제(binary 파일 제공)
  - 2개의 이미지센서 초기화 후, video 신호 확인이 완료되면 LED D11을 켠다. 그리고 USB 3.0 통신을 시작한다.
  - 필요한 프로그램 : UVC 지원 프로그램
  - Python capture code 예제 Github 링크 : [https://github.com/mgsgo/M-AR0230-DUAL-CAM-UVC/tree/master/rev110/python\\_capture\\_code](https://github.com/mgsgo/M-AR0230-DUAL-CAM-UVC/tree/master/rev110/python_capture_code)
    - python 최신버전, opencv 최신버전 필요
  - 예제 실행 동영상 링크 : not yet prepared

- 기본 동작 케이블 연결
  - USB 3.0 펌웨어 점퍼 OFF
  - UAB 3.0 AM BM cable 연결

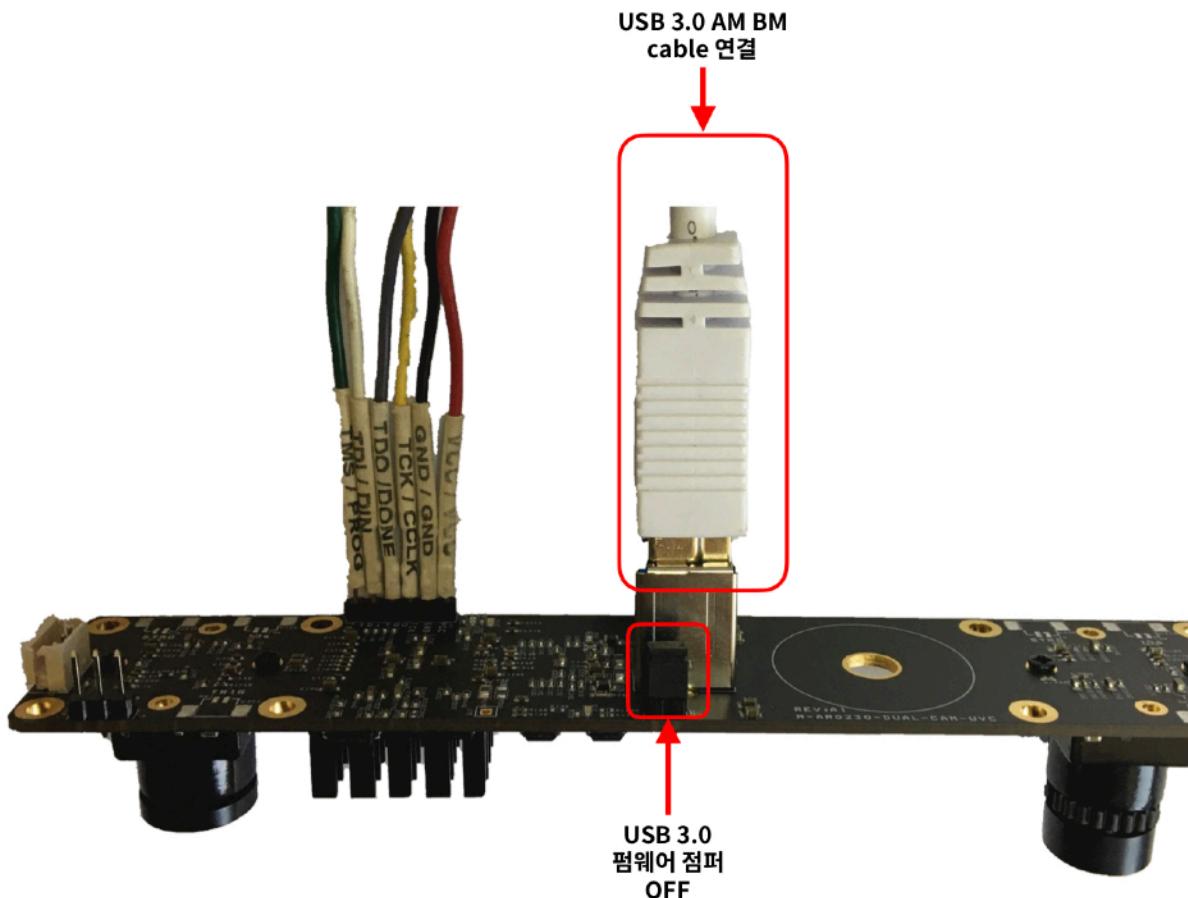


Figure 3.2.1 M-AR0230-DUAL-CAM-UVC 보드 케이블 연결

- cap3840x1080x30fps\_display1920x1080x2.py

- 3840x1080 해상도 영상 capture/1920x1080 해상도 영상 display x2 python code

```
import cv2
capture = cv2.VideoCapture(0)
capture.set(cv2.CAP_PROP_FRAME_WIDTH, 3840)
capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 1080)
while True:
    ret, frame = capture.read()
    image = frame.copy()

    image_1 = image[0:1080, 0:1920]
    image_2 = image[0:1080, 1920:3840]

    cv2.imshow("video1", image_1)
    cv2.imshow("video2", image_2)

    if cv2.waitKey(1) > 0: break
capture.release()
cv2.destroyAllWindows()
```

- cap3840x1080x30fps\_display960x540x2.py

- 3840x1080 해상도 영상 capture/960x540 해상도 영상 display x2 python code

```
import cv2
capture = cv2.VideoCapture(0)
capture.set(cv2.CAP_PROP_FRAME_WIDTH, 3840)
capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 1080)
while True:
    ret, frame = capture.read()
    image = frame.copy()

    image_1 = image[0:1080, 0:1920]
    image_2 = image[0:1080, 1920:3840]

    image_1 = cv2.resize(image_1, (960, 540))
    image_2 = cv2.resize(image_2, (960, 540))

    cv2.imshow("video1", image_1)
    cv2.imshow("video2", image_2)
    if cv2.waitKey(1) > 0: break
capture.release()
cv2.destroyAllWindows()
```

- cap3840x1080x30fps\_display960x540x2\_save3840x1080x1x30fps.py
  - 3840x1080 해상도 영상 capture/960x540 해상도 영상 display x2/3840x1080 해상도 영상x1 저장

python code

```
import cv2
capture = cv2.VideoCapture(0)
fourcc = cv2.VideoWriter_fourcc(*'mp4v')
capture.set(cv2.CAP_PROP_FRAME_WIDTH, 3840)
capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 1080)
```

```
writer_1 = cv2.VideoWriter('3840x1080_cam12.mp4', fourcc, 30.0, (3840,1080))
```

while True:

```
    ret, frame = capture.read()
    image = frame.copy()
    image_1 = image[0:1080, 0:1920]
    image_2 = image[0:1080, 1920:3840]
```

```
writer_1.write(image)
```

```
image_1 = cv2.resize(image_1, (960, 540))
image_2 = cv2.resize(image_2, (960, 540))
```

```
cv2.imshow("video1", image_1)
cv2.imshow("video2", image_2)
```

```
if cv2.waitKey(1) > 0: break
```

```
capture.release()
cv2.destroyAllWindows()
```

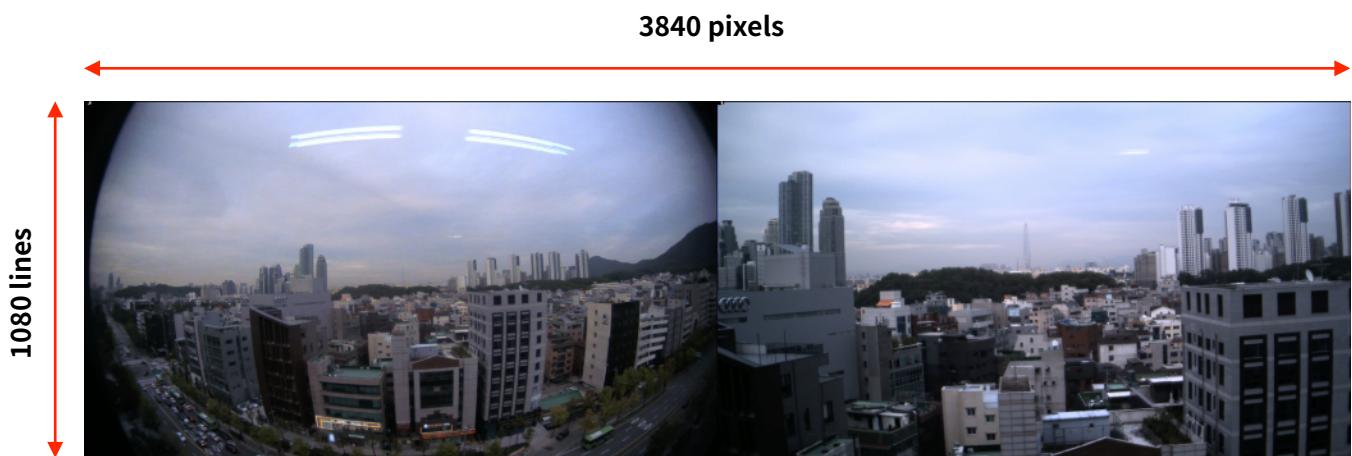


Figure 3.2.2 M-AR0230-DUAL-CAM-UVC 보드 영상 전송 기본 펌웨어(3840x1080) capture 결과

- cap3840x1080x30fps\_display960x540x2\_save1920x1080x2x30fps.py
  - 3840x1080 해상도 영상 capture/960x540 해상도 영상 display x2/1920x1080 해상도 영상x2 저장

python code

```
import cv2
capture = cv2.VideoCapture(0)
fourcc = cv2.VideoWriter_fourcc(*'mp4v')
capture.set(cv2.CAP_PROP_FRAME_WIDTH, 3840)
capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 1080)

writer_1 = cv2.VideoWriter('1920x1080_cam1.mp4', fourcc, 30.0, (1920,1080))
writer_2 = cv2.VideoWriter('1920x1080_cam2.mp4', fourcc, 30.0, (1920,1080))
while True:
    ret, frame = capture.read()
    image = frame.copy()
    image_1 = image[0:1080, 0:1920]
    image_2 = image[0:1080, 1920:3840]

    writer_1.write(image_1)
    writer_2.write(image_2)

    image_1 = cv2.resize(image_1, (960, 540))
    image_2 = cv2.resize(image_2, (960, 540))

    cv2.imshow("video1", image_1)
    cv2.imshow("video2", image_2)

    if cv2.waitKey(1) > 0: break
capture.release()
cv2.destroyAllWindows()
```

- 윈도우 프로그램에서의 UVC control
  - **선명도(auto exposure enable) 0일 때**
    - 밝기 : 이미지센서 exposure
    - 개인 : 이미지센서 개인

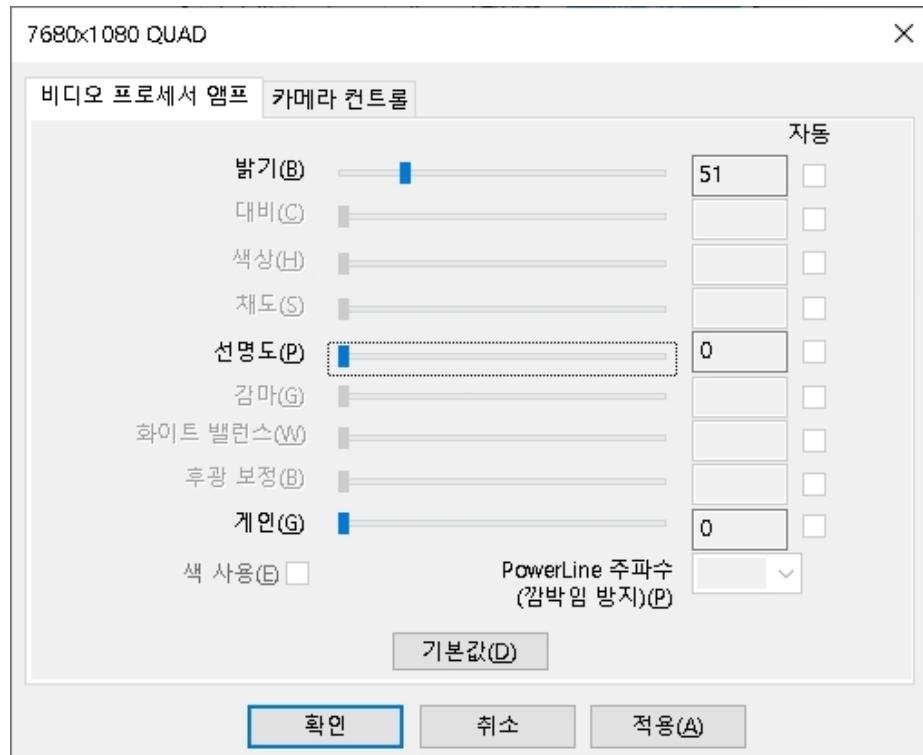


Figure 3.2.3 M-AR0230-DUAL-CAM-UVC 밝기/개인/선명도(auto exposure) 제어

- **선명도(auto exposure enable) 1일 때**

- 밝기 : 원하는 평균 밝기값
- 개인 : 상관없음

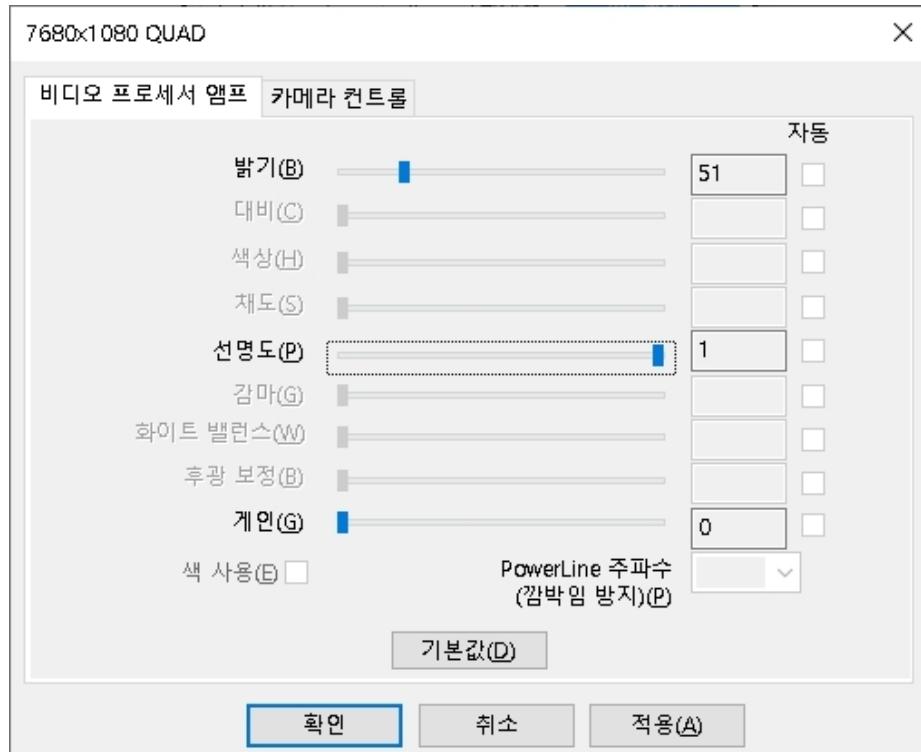


Figure 3.2.4 M-AR0230-DUAL-CAM-UVC 밝기/개인/선명도(auto exposure) 제어

### 3.3 펌웨어 업데이트

- USB 3.0 펌웨어 업데이트
  - 동영상 링크 : not yet prepared
- USB 3.0 펌웨어 업데이트시의 케이블 연결
  - USB 3.0 펌웨어 점퍼 ON
  - UAB 3.0 AM BM cable 연결

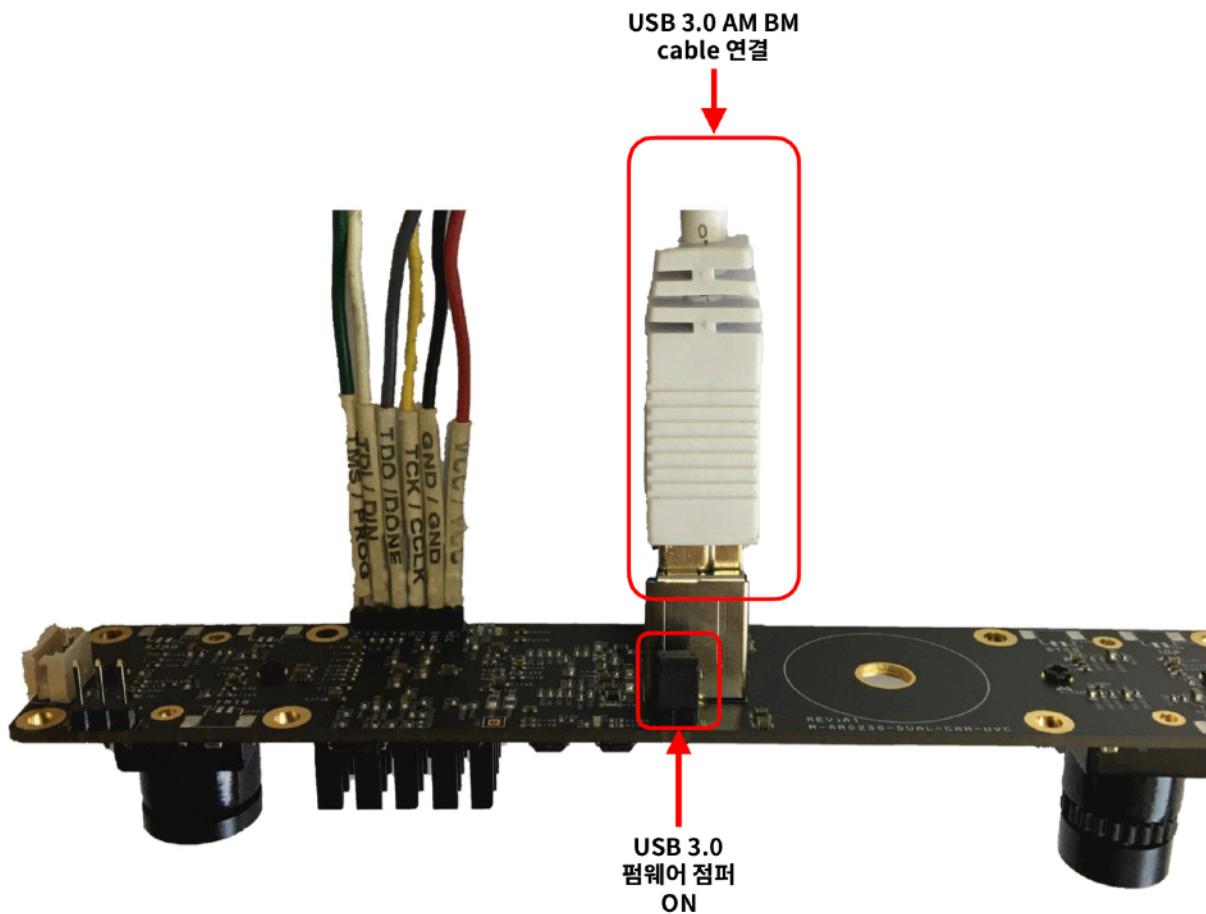


Figure 3.3.1 M-AR0230-DUAL-CAM-UVC 보드 USB 3.0 케이블 연결, USB 3.0 펌웨어 업데이트 점퍼 설정

- USB 3.0 펌웨어 업데이트 프로그램(USB control center)
  - USB control center 실행, device 선택
  - Program/FX3/I2C EEPROM 선택
  - USB control center 실행, device 선택, file 선택
  - Programming I2C EEPROM Succeeded

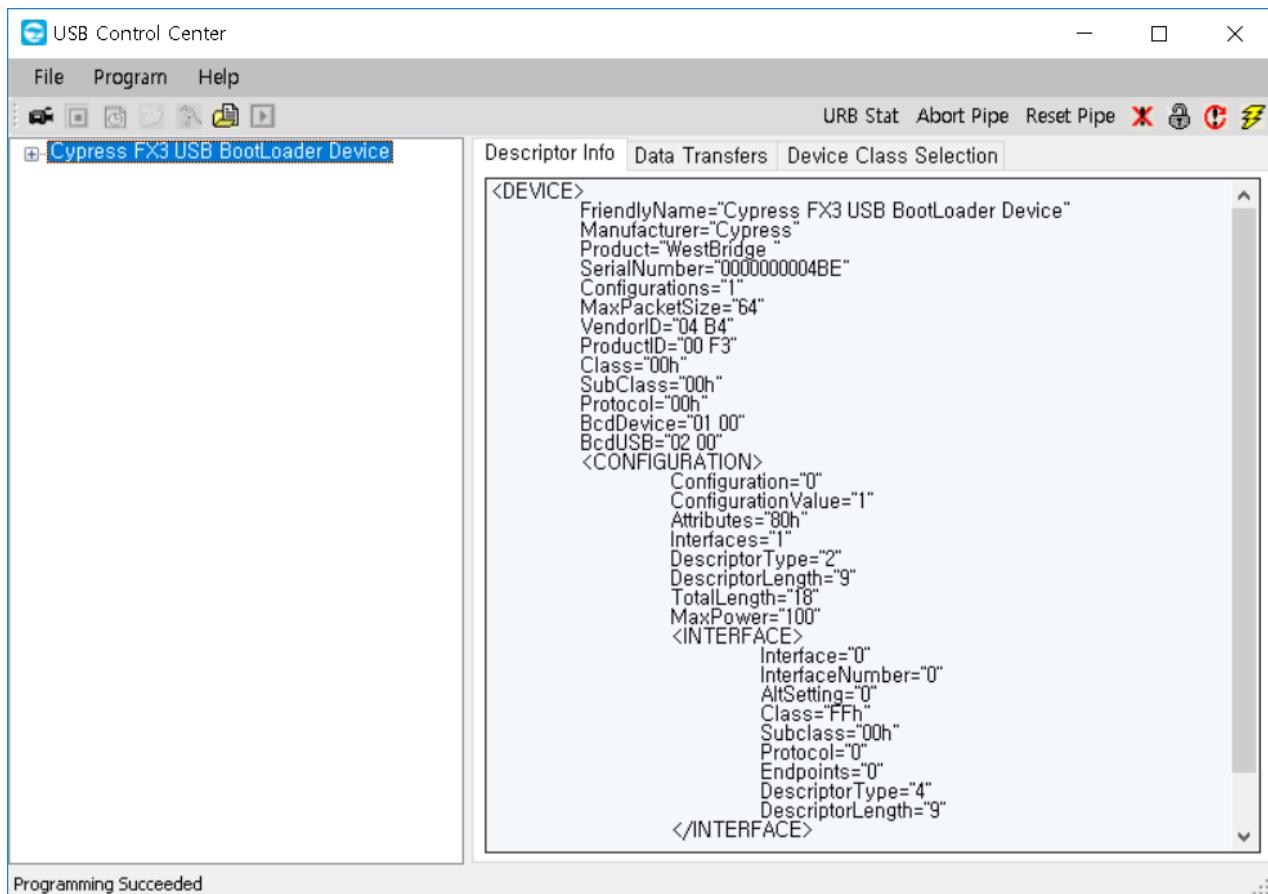


Figure 3.3.2 M-AR0230-DUAL-CAM-UVC 보드 USB 3.0 device 선택

## M-AR0230-DUAL-CAM-UVC USER MANUAL

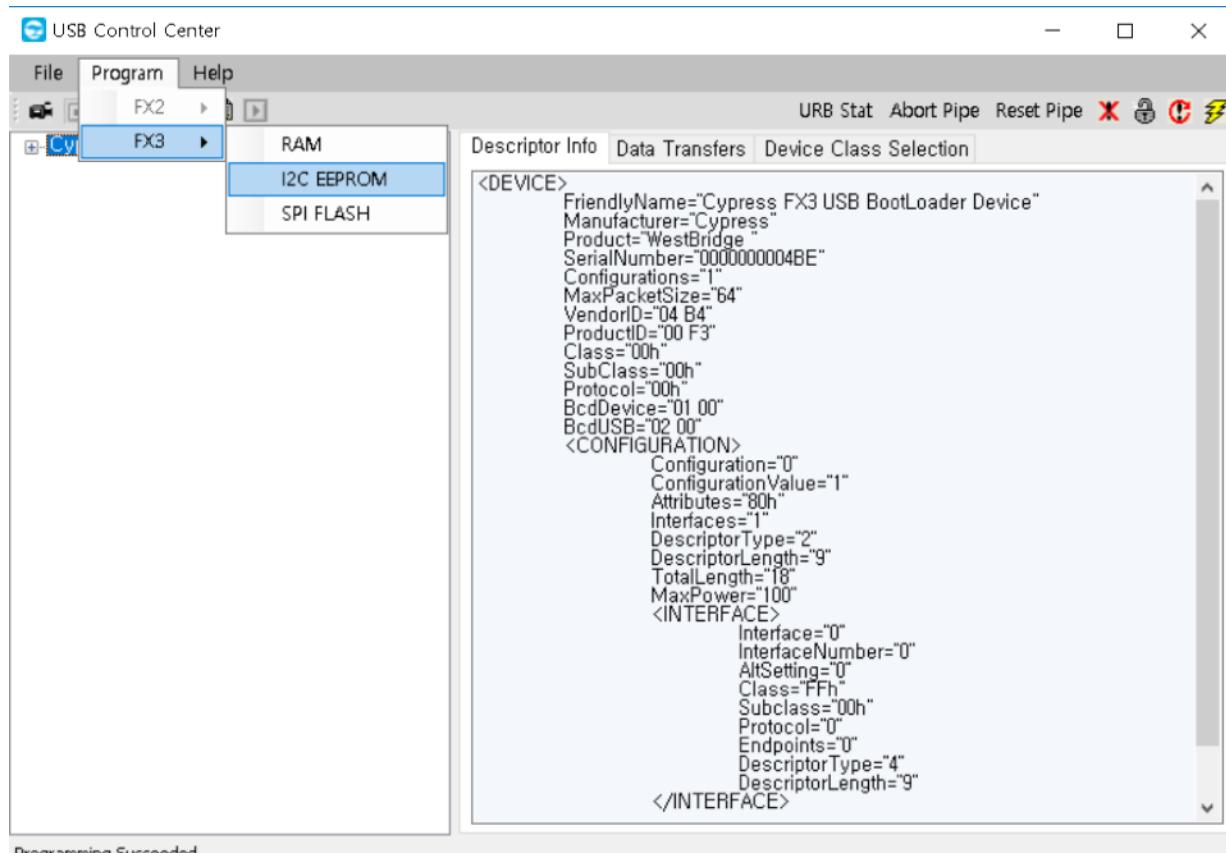


Figure 3.3.3 M-AR0230-DUAL-CAM-UVC USB 3.0 펌웨어 업데이트 EEPROM 선택

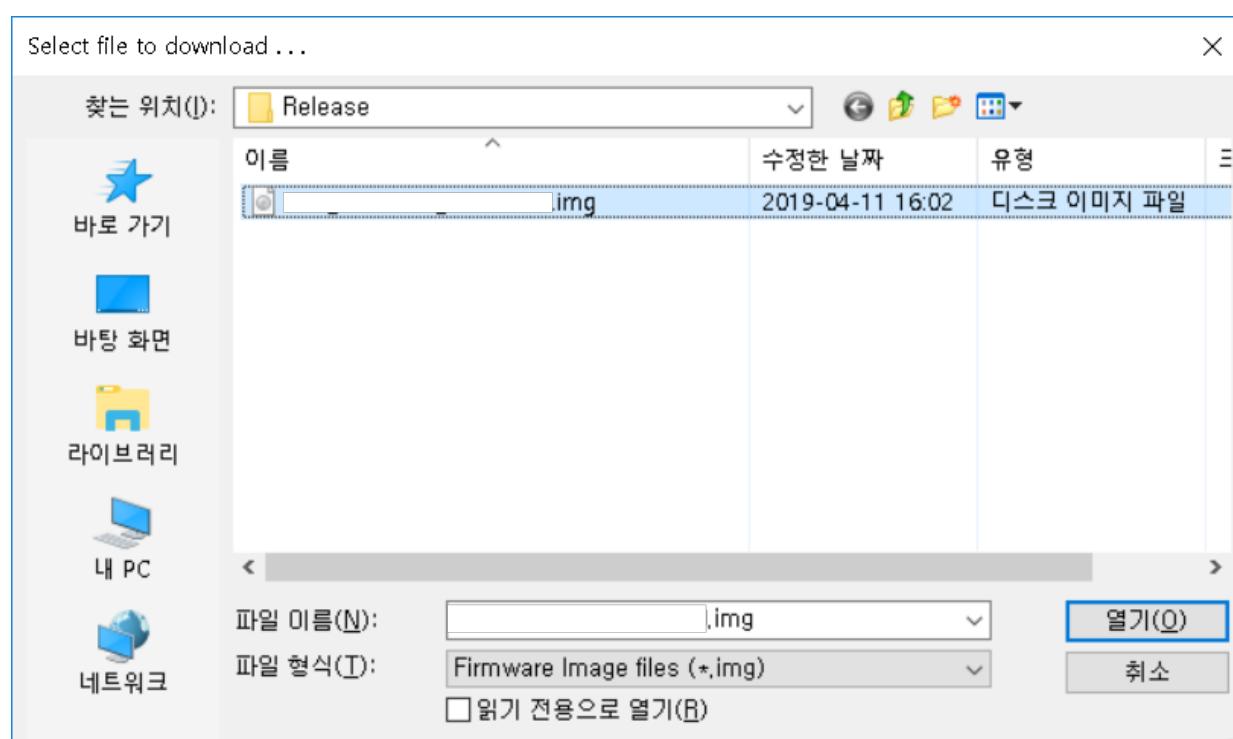


Figure 3.3.4 M-AR0230-DUAL-CAM-UVC 보드 USB 3.0 펌웨어 업데이트 device 선택, file 선택

## M-AR0230-DUAL-CAM-UVC USER MANUAL

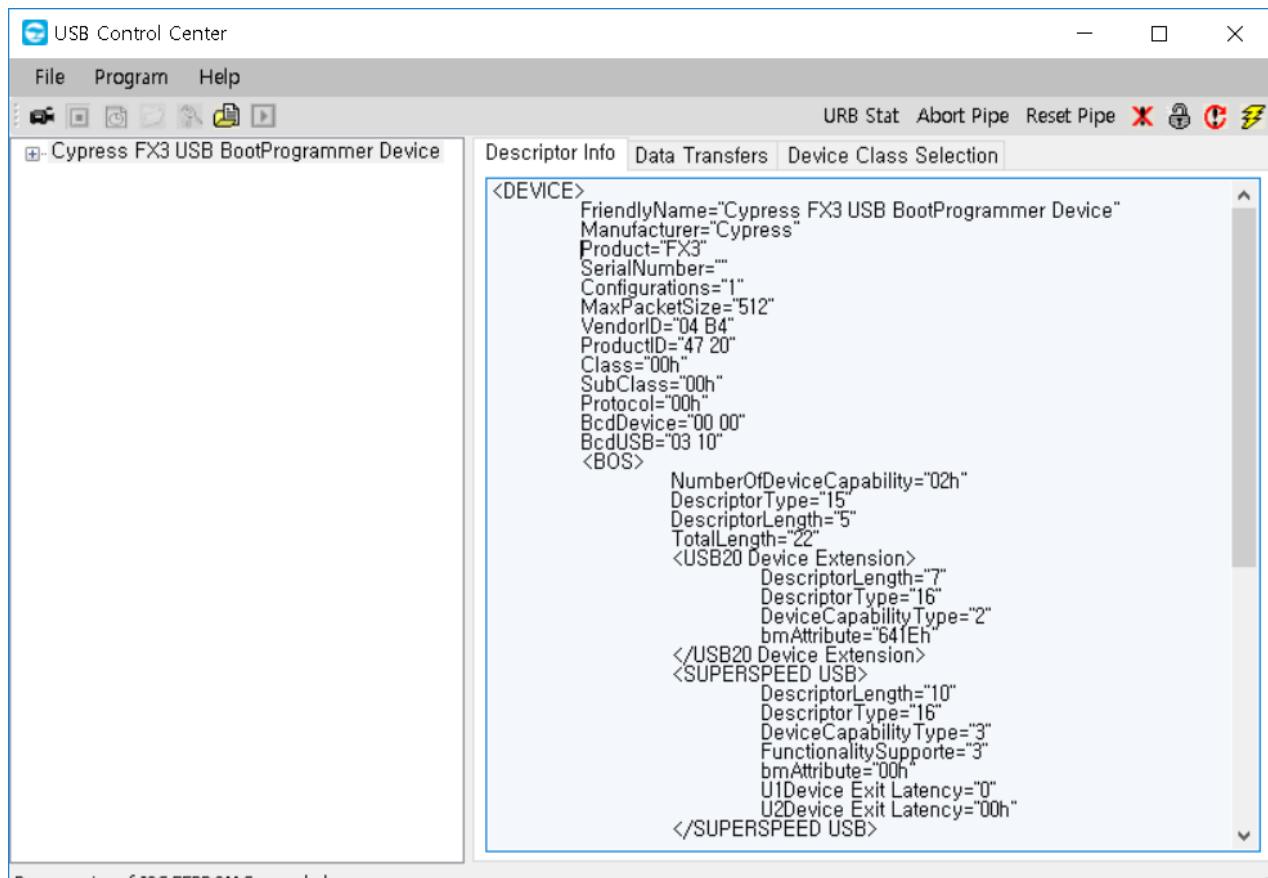


Figure 3.3.5 M-AR0230-DUAL-CAM-UVC 보드 USB 3.0 펌웨어 업데이트 Programming Succeeded

M-AR0230-DUAL-CAM-UVC USER MANUAL

- FPGA 펌웨어 업데이트
  - 동영상 링크 : not yet prepared
- FPGA 펌웨어 업데이트시의 케이블 연결
  - Xilinx JTAG cable 연결(1:VCC, 2:GND, 3:TCK, 4:TDO, 5:TDI, 6:TMS)
  - UAB 3.0 AM BM cable 연결

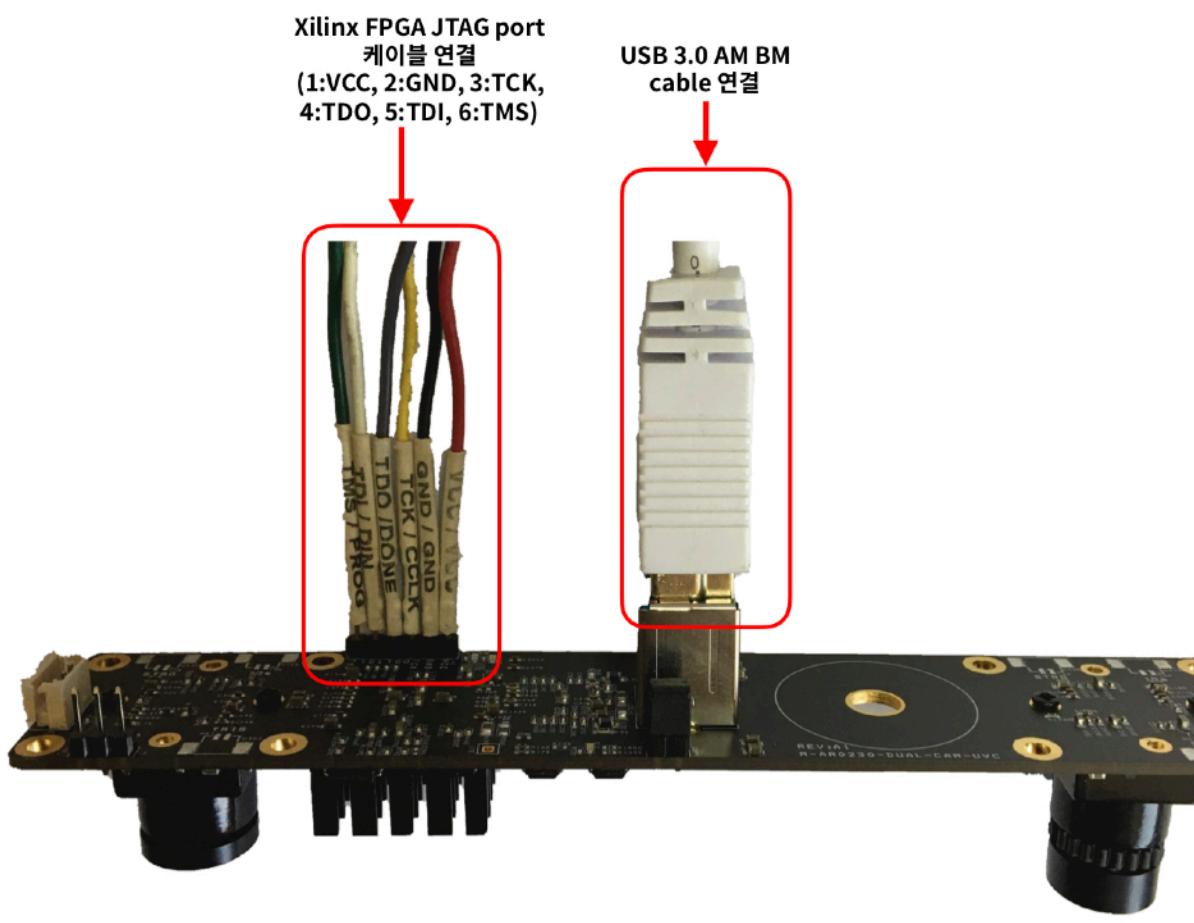


Figure 3.3.6 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 케이블 연결

- FPGA 펌웨어 업데이트 프로그램 실행 (Xilinx ISE iMPACT)
  - Xilinx iMPACT 실행, Initialize chain
  - Add SPI-BPI Flash
  - MCS file select
  - MCS file select(SPI PROM--W25Q64FV select)
  - MCS file select(SPI PROM--W25Q64FV select)
  - Program
  - Program(OK)
  - Program(Program Succeed)

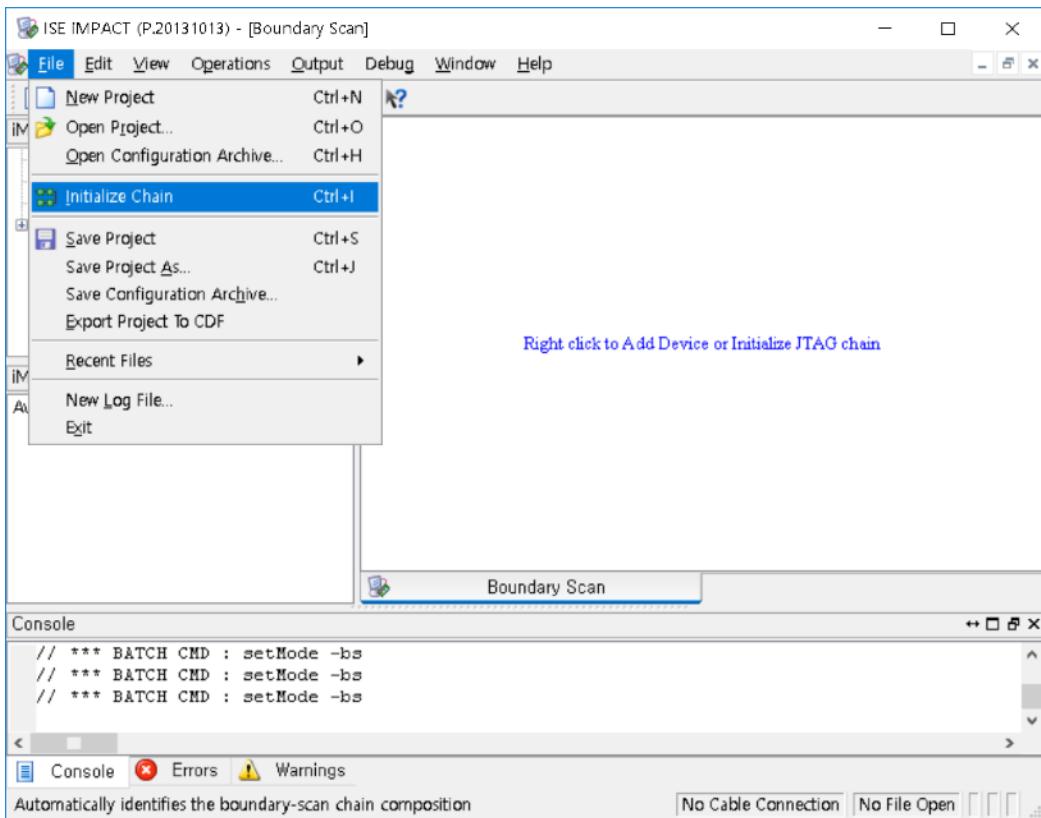


Figure 3.3.7 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 Initialize chain

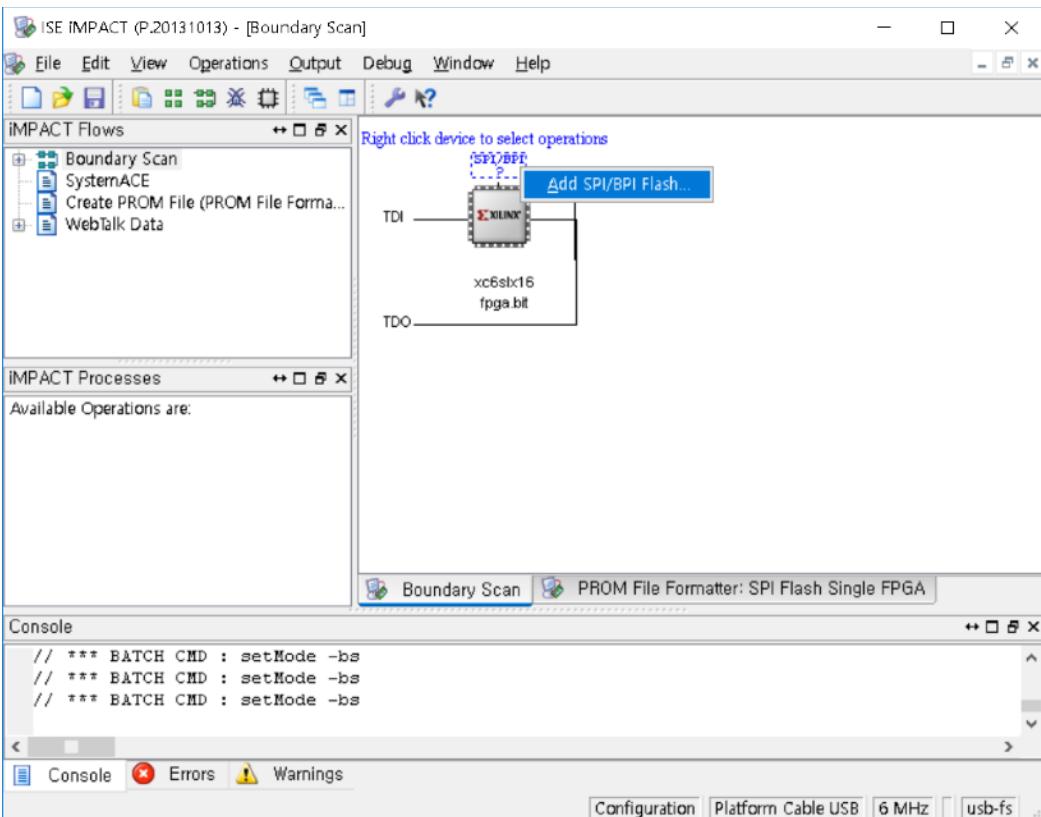


Figure 3.3.8 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 Add SPI-BPI Flash

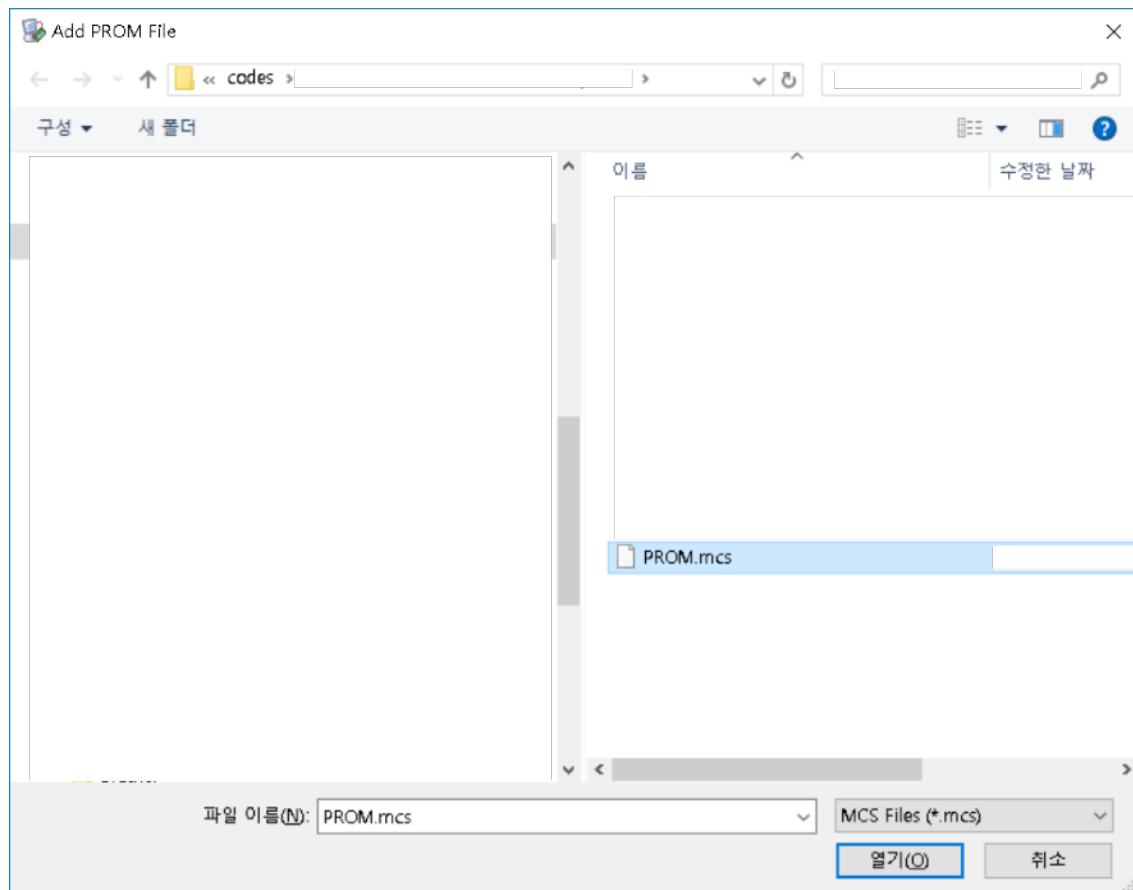


Figure 3.3.9 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 MCS file select

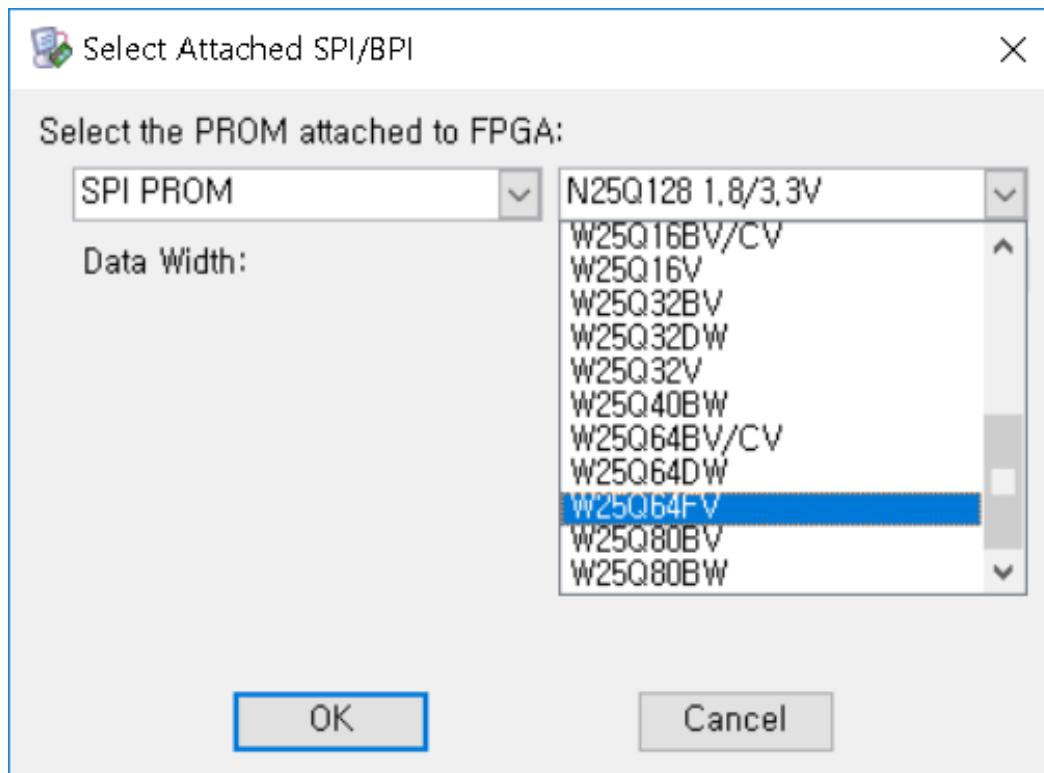


Figure 3.3.10 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 SPI PROM--W25Q64FV select

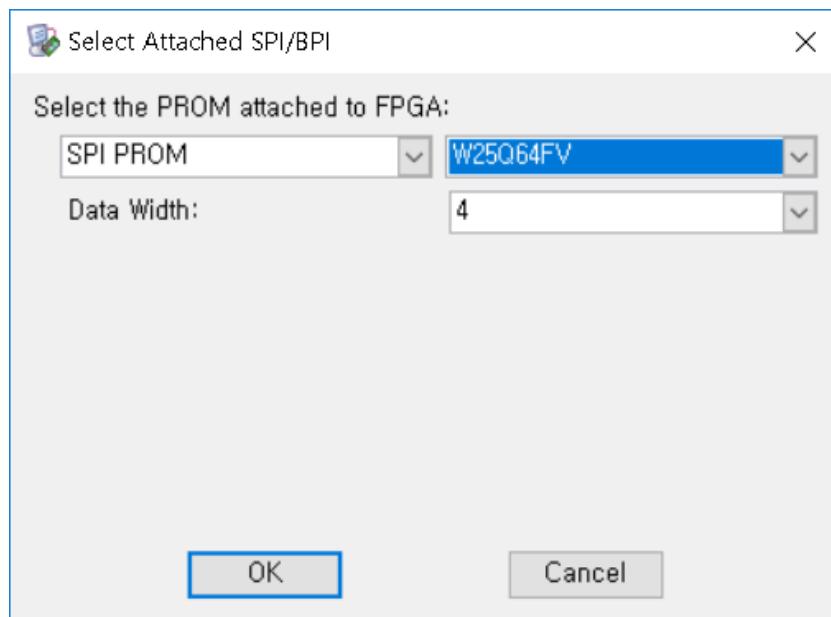


Figure 3.3.11 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 SPI PROM--W25Q64FV select

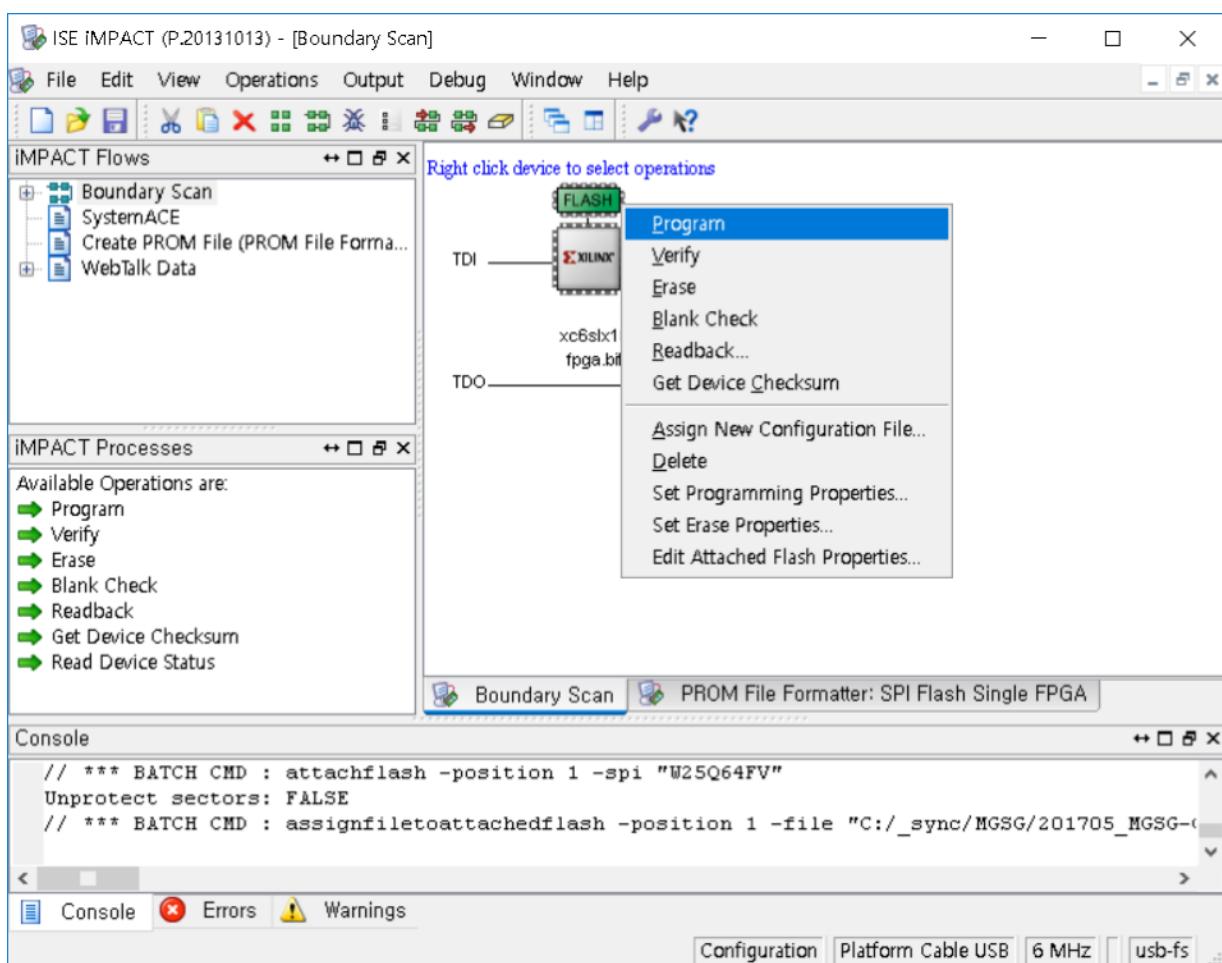


Figure 3.3.12 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 Program

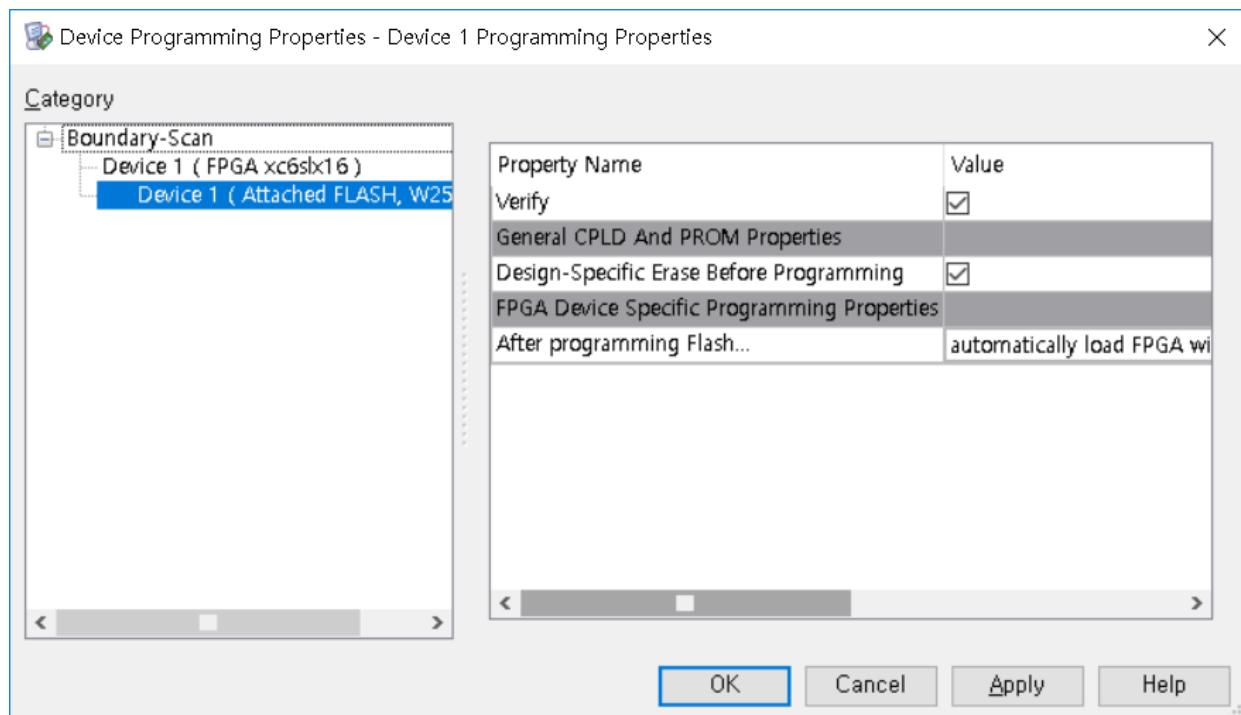


Figure 3.3.13 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 Program(OK)

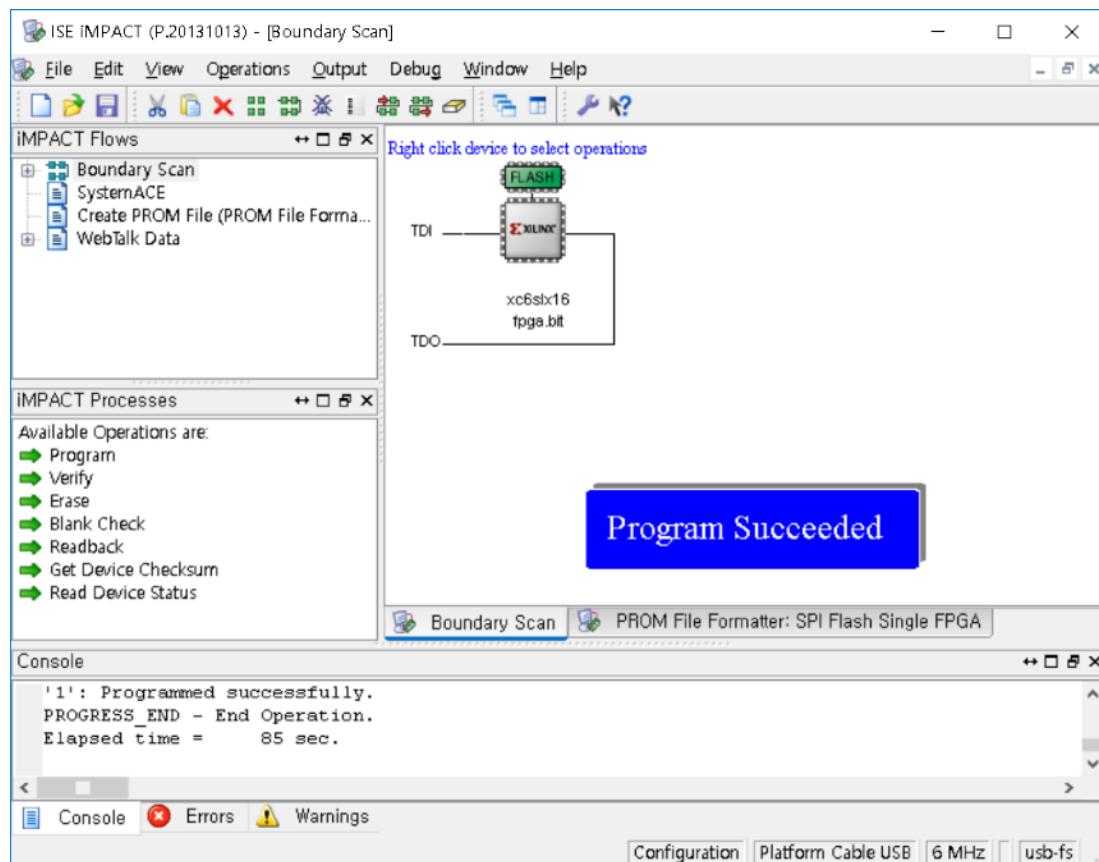


Figure 3.3.14 M-AR0230-DUAL-CAM-UVC 보드 FPGA 펌웨어 업데이트 Program Succeed

### 3.4 FPGA\_USB3.0\_UVC 영상 전송 예제

- USB 3.0 펌웨어 업데이트
  - Github 링크 : [https://github.com/mgsgo/M-AR0230-DUAL-CAM-UVC/tree/master/rev110/example\\_FPGA\\_USB3.0\\_UVC](https://github.com/mgsgo/M-AR0230-DUAL-CAM-UVC/tree/master/rev110/example_FPGA_USB3.0_UVC)
  - 동영상 링크 : not yet prepared
- FPGA\_USB3.0\_UVC 예제 케이블 연결
  - USB 3.0 펌웨어 점퍼 ON
  - UAB 3.0 AM BM cable 연결
  - Xilinx JTAG 연결

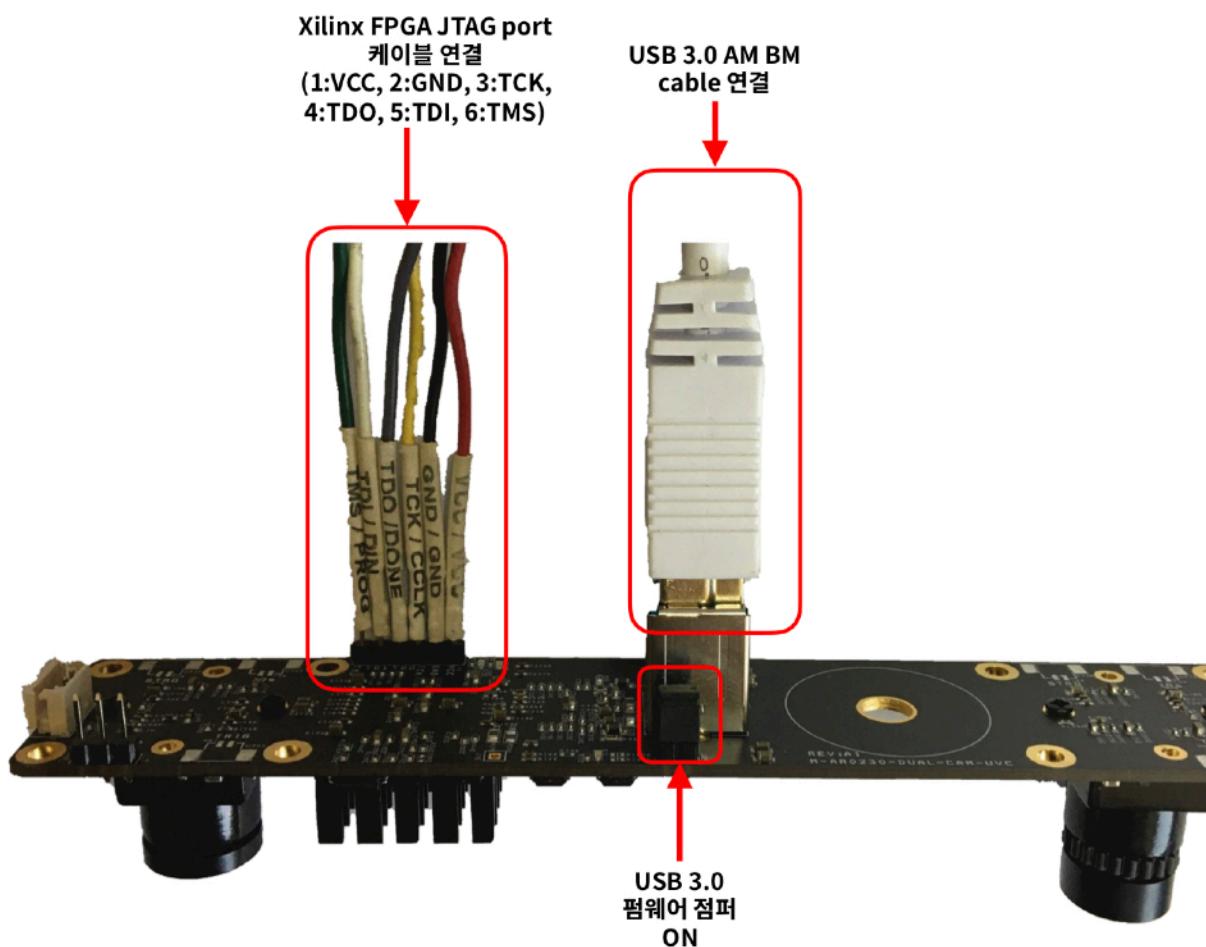


Figure 3.4.1 M-AR0230-DUAL-CAM-UVC 보드 FPGA\_USB3.0\_UVC 예제 케이블 연결/점퍼 설정

- FPGA 펌웨어 다운로드 (3.3 펌웨어 업데이트 참조)
- USB 3.0 펌웨어 다운로드 (3.3 펌웨어 업데이트 참조)
- Python capture code 실행(1920x1080x30fps)

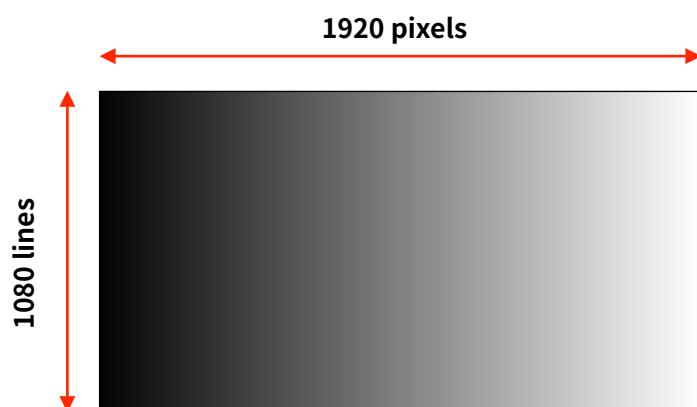


Figure 3.4.2 M-AR0230-DUAL-CAM-UVC 보드 FPGA\_USB3.0\_UVC 예제 영상 capture 결과

## 4. Support

### 4.1 구매 지원

각 유통업체를 통해서 구매 가능

### 4.2 기술지원

Contact to : [mgsg\\_opensource@gmail.com](mailto:mgsg_opensource@gmail.com)

Github page : [github.com/mgsgo/M-AR0230-DUAL-CAM-UVC](https://github.com/mgsgo/M-AR0230-DUAL-CAM-UVC)

Homepage : [www.mgsgo.com](http://www.mgsgo.com)