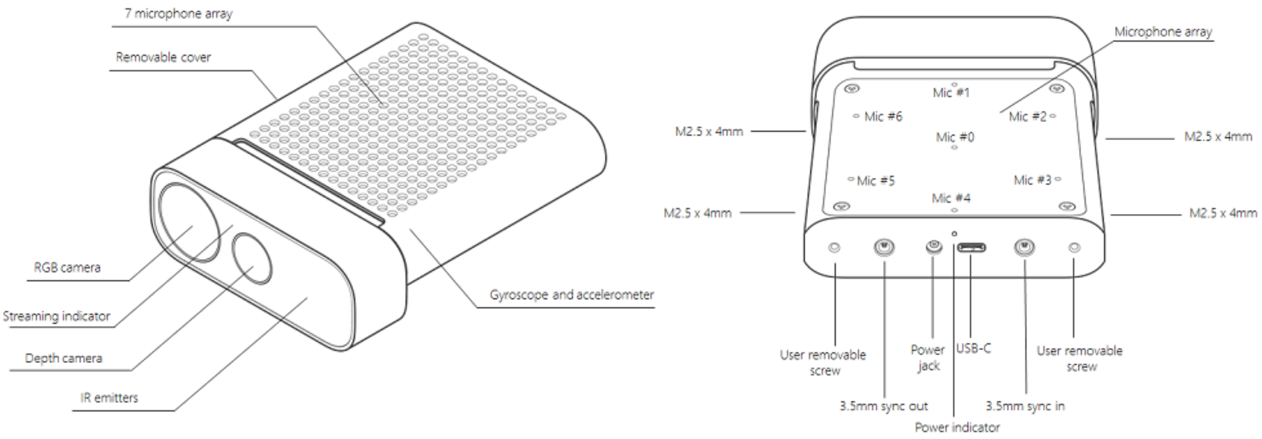
# Azure Kinect DK hardware specifications

06/26/2019 • 5 minutes to read • +1

## In this article

This article provides details about how Azure Kinect hardware integrates Microsoft's latest sensor technology into a single, USB-connected accessory.



## Terms

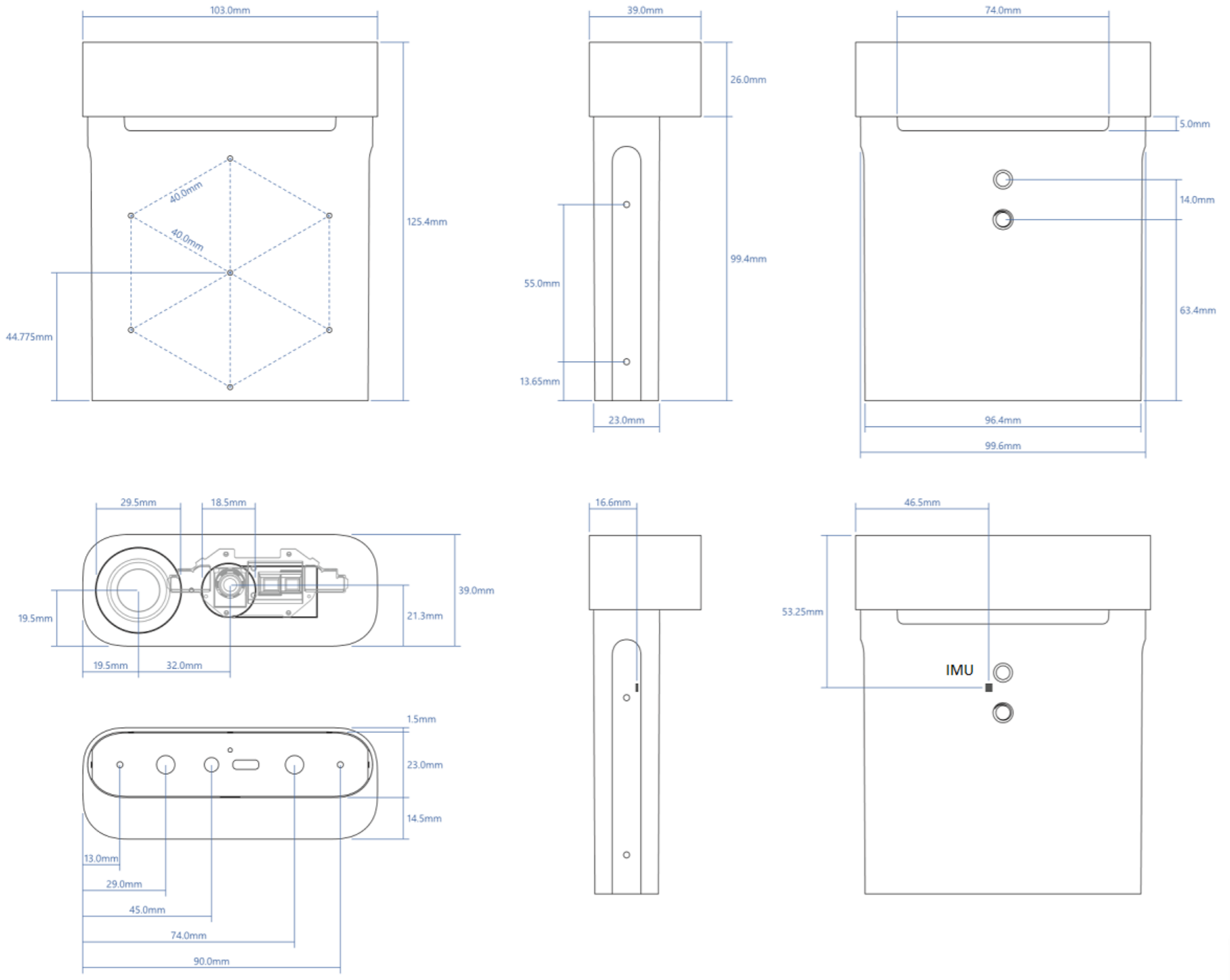
These abbreviated terms are used throughout this article.

- NFOV (Narrow field-of-view depth mode)
- WFOV (Wide field-of-view depth mode)
- FOV (Field-of-view)
- FPS (Frames-per-second)
- IMU (Inertial Measurement Unit)
- FoI (Field of Interest)

## Product dimensions and weight

The Azure Kinect device consists of the following size and weight dimensions.

- **Dimensions**: 103 x 39 x 126 mm
- **Weight**: 440 g



# Operating environment

Azure Kinect DK is intended for developers and commercial businesses operating under the following ambient conditions:

- **Temperature**: 10-25⁰C
- **Humidity**: 8-90% (non-condensing) Relative Humidity

ⓘ **Note**

Use outside of the ambient conditions could cause the device to fail and/or function incorrectly. These ambient conditions are applicable for the environment immediately around the device under all operational conditions. When used with an external enclosure, active temperature control and/or other cooling solutions are recommended to ensure the device is maintained within these ranges. The device design features a cooling channel in between the front section and rear sleeve. When you implement the device, make sure this cooling channel is not obstructed.

Refer to additional product safety information.

# Depth camera supported operating modes

Azure Kinect DK integrates a Microsoft designed 1-Megapixel Time-of-Flight (ToF) depth camera using the image sensor presented at ISSCC 2018. The depth camera supports the modes indicated below:

| Mode | Resolution | FoI | FPS | Operating range* | Exposure time |
|---|---|---|---|---|---|
| NFOV unbinned | 640x576 | 75°x65° | 0, 5, 15, 30 | 0.5 - 3.86 m | 12.8 ms |
| NFOV 2x2 binned (SW) | 320x288 | 75°x65° | 0, 5, 15, 30 | 0.5 - 5.46 m | 12.8 ms |
| WFOV 2x2 binned | 512x512 | 120°x120° | 0, 5, 15, 30 | 0.25 - 2.88 m | 12.8 ms |
| WFOV unbinned | 1024x1024 | 120°x120° | 0, 5, 15 | 0.25 - 2.21 m | 20.3 ms |

| Mode | Resolution | FoI | FPS | Operating range* | Exposure time |
|---|---|---|---|---|---|
| Passive IR | 1024x1024 | N/A | 0, 5, 15, 30 | N/A | 1.6 ms |

*15% to 95% reflectivity at 850nm, 2.2 μW/cm$^2$/nm, random error std. dev. ≤ 17 mm, typical systematic error < 11 mm + 0.1% of distance without multi-path interference. Depth provided outside of indicated range depending on object reflectivity.

## Color camera supported operating modes

Azure Kinect DK includes an OV12A10 12MP CMOS sensor rolling shutter sensor. The native operating modes are listed below:

| RGB Camera Resolution (HxV) | Aspect Ratio | Format Options | Frame Rates (FPS) | Nominal FOV (HxV) (post-processed) |
|---|---|---|---|---|
| 3840x2160 | 16:9 | MJPEG | 0, 5, 15, 30 | 90°x59° |
| 2560x1440 | 16:9 | MJPEG | 0, 5, 15, 30 | 90°x59° |
| 1920x1080 | 16:9 | MJPEG | 0, 5, 15, 30 | 90°x59° |
| 1280x720 | 16:9 | MJPEG/YUY2/NV12 | 0, 5, 15, 30 | 90°x59° |
| 4096x3072 | 4:3 | MJPEG | 0, 5, 15 | 90°x74.3° |
| 2048x1536 | 4:3 | MJPEG | 0, 5, 15, 30 | 90°x74.3° |

The RGB camera is USB Video class-compatible and can be used without the Sensor SDK. The RGB camera color space: BT.601 full range [0..255].

ⓘ Note

The Sensor SDK can provide color images in the BGRA pixel format. This is not a native mode supported by the device and causes additional CPU load when used. The host CPU is used to convert from MJPEG images received from the device.

## RGB camera exposure time values

Below is the mapping for the acceptable RGB camera manual exposure values:

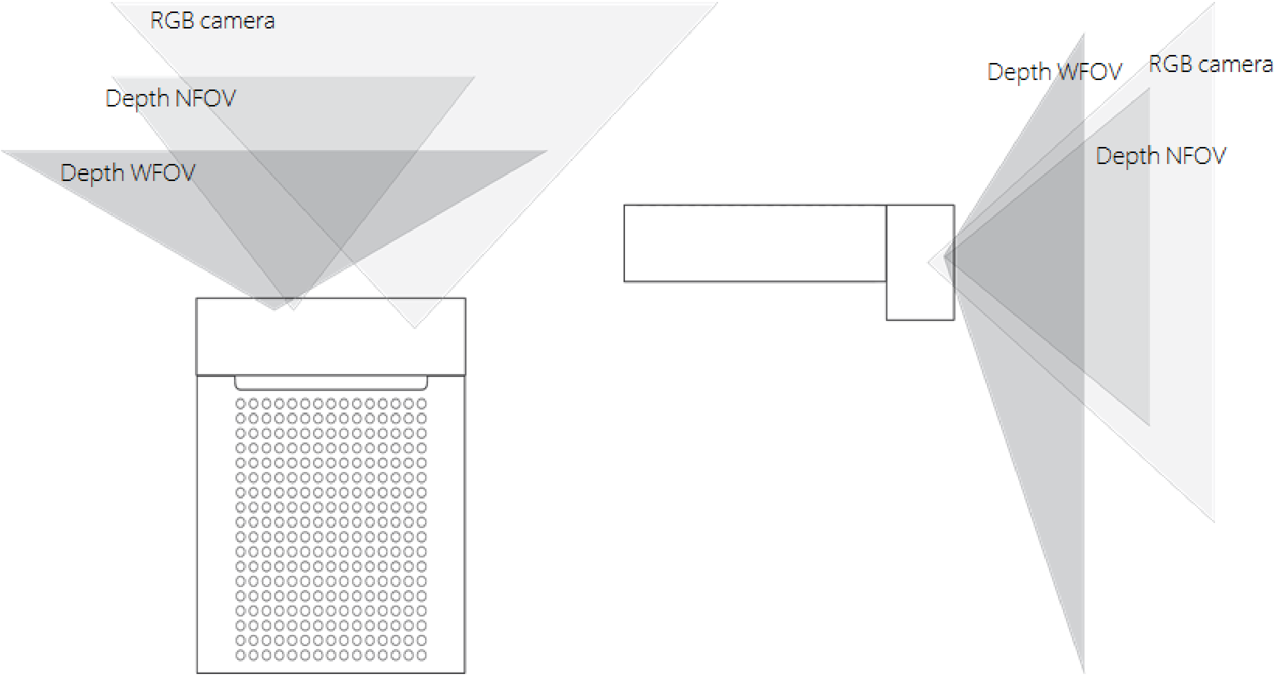
| exp | 2^exp | 50Hz | 60Hz |
|---|---|---|---|
| -11 | 488 | 500 | 500 |
| -10 | 977 | 1250 | 1250 |
| -9 | 1953 | 2500 | 2500 |
| -8 | 3906 | 10000 | 8330 |
| -7 | 7813 | 20000 | 16670 |
| -6 | 15625 | 30000 | 33330 |
| -5 | 31250 | 40000 | 41670 |
| -4 | 62500 | 50000 | 50000 |
| -3 | 125000 | 60000 | 66670 |
| -2 | 250000 | 80000 | 83330 |
| -1 | 500000 | 100000 | 100000 |
| 0 | 1000000 | 120000 | 116670 |
| 1 | 2000000 | 130000 | 133330 |

## Depth sensor raw timing

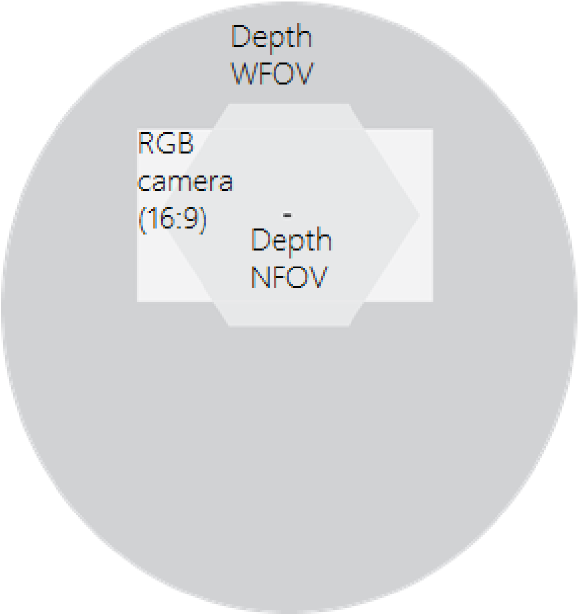
| Depth Mode | IR Pulses | Pulse Width | Idle Periods | Idle Time | Exposure Time |
|---|---|---|---|---|---|
| NFOV Unbinned<br>NFOV 2xx Binned<br>WFOV 2x2 Binned | 9 | 125 us | 8 | 1450 us | 12.8 ms |
| WFOV Unbinned | 9 | 125 us | 8 | 2390 us | 20.3 ms |

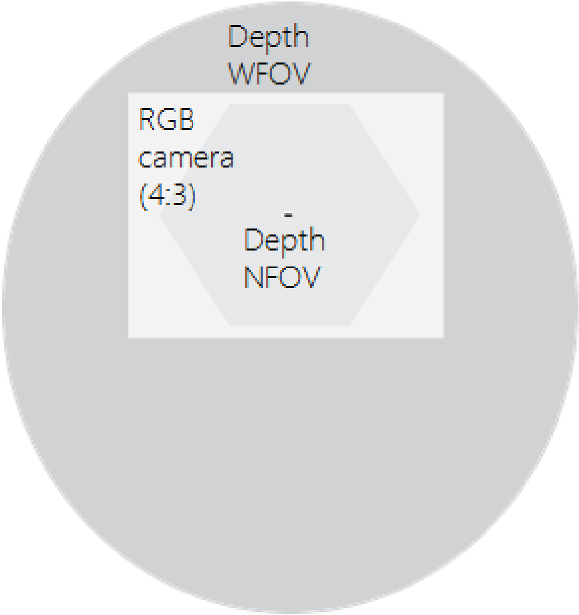
# Camera field of view

The next image shows the depth and RGB camera field-of-view, or the angles that the sensors "see". This diagram shows the RGB camera in a 4:3 mode.



This image demonstrates the camera's field-of-view as seen from the front at a distance of 2000 mm.





ⓘ **Note**

When depth is in NFOV mode, the RGB camera has better pixel overlap in 4:3 than 16:9 resolutions.

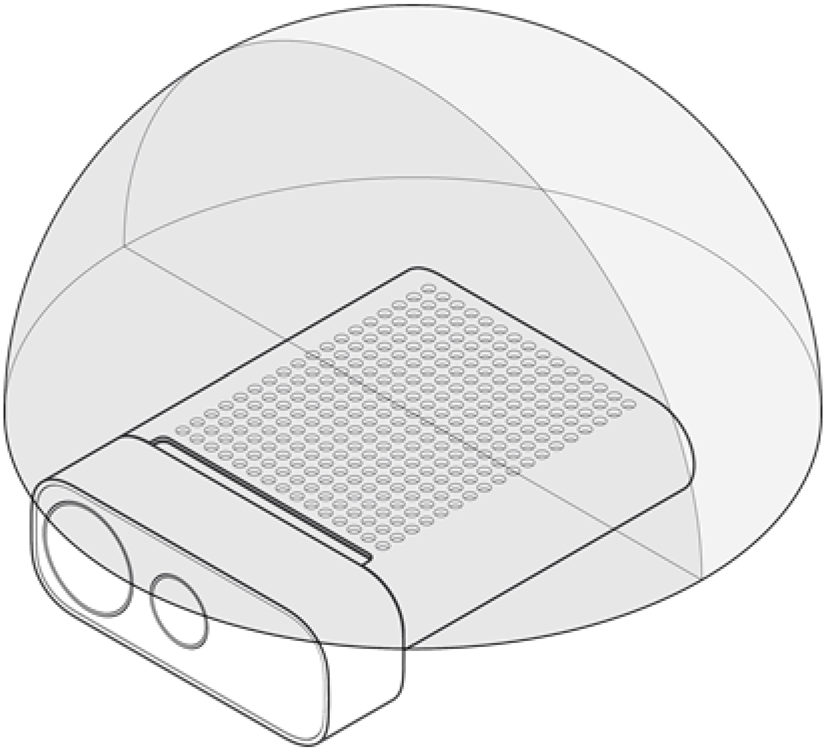
# Motion sensor (IMU)

The embedded Inertial Measurement Unit (IMU) is an LSM6DSMUS and includes both an accelerometer and a gyroscope. The accelerometer and gyroscope are simultaneously sampled at 1.6 kHz. The samples are reported to the host at a 208 Hz.

# Microphone array

Azure Kinect DK embeds a high-quality, seven microphone circular array that identifies as a standard USB audio class 2.0 device. All 7 channels can be accessed. The performance specifications are:

- Sensitivity: -22 dBFS (94 dB SPL, 1 kHz)
- Signal to noise ratio > 65 dB
- Acoustic overload point: 116 dB



# USB

Azure Kinect DK is a USB3 composite device that exposes the following hardware endpoints to the operating system:

Vendor ID is 0x045E (Microsoft), Product ID table below:

| USB Interface | PNP IP | Notes |
|---|---|---|
| USB3.1 Gen1 Hub | 0x097A | The main hub |
| USB2.0 Hub | 0x097B | HS USB |
| Depth camera | 0x097C | USB3.0 |
| Color camera | 0x097D | USB3.0 |
| Microphones | 0x097E | HS USB |

# Indicators

The device has a camera streaming indicator on the front of the device that can be disabled programmatically using the Sensor SDK.

The status LED behind the device indicates device state:

| When the light is | It means |
|---|---|
| Solid white | Device is on and working properly. |
| Flashing white | Device is on but doesn’t have a USB 3.0 data connection. |
| Flashing amber | Device doesn't have enough power to operate. |
| Amber flashing white | Firmware update or recovery in progress |

## Power device

The device can be powered in two ways:

1. Using the in-box power supply. Data is connected by a separate USB Type-C to Type-A cable.
2. Using a Type-C to Type-C cable for both power and data.

A Type-C to Type-C cable isn't included with the Azure Kinect DK.

ⓘ **Note**

- The in-box power supply cable is a USB Type-A to single post barrel connector. Use the provided wall-power supply with this cable. The device is capable of drawing more power than two standard USB Type-A ports can provide.
- USB cables do matter and we recommended to use high-quality cables and verify functionality before deploying the unit remotely.

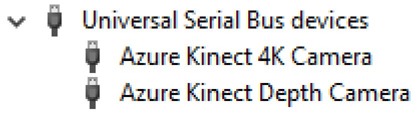
**Tip**

To select a good Type-C to Type-C cable:

- The USB certified cable must support both power and data.
- A passive cable should be less than 1.5m in length. If longer, use an active cable.
- The cable needs to support no less than >1.5A. Otherwise you need to connect an external power supply.

Verify cable:

- Connect device via the cable to the host PC.
- Validate that all devices enumerate correctly in Windows device manager. Depth and RGB camera should appear as shown in the example below.



Universal Serial Bus devices
Azure Kinect 4K Camera
Azure Kinect Depth Camera

- Validate that cable can stream reliably on all sensors in the Azure Kinect Viewer, with the following settings:
  - Depth camera: NFOV unbinned
  - RGB Camera: 2160p
  - Microphones and IMU enabled

# Power consumption

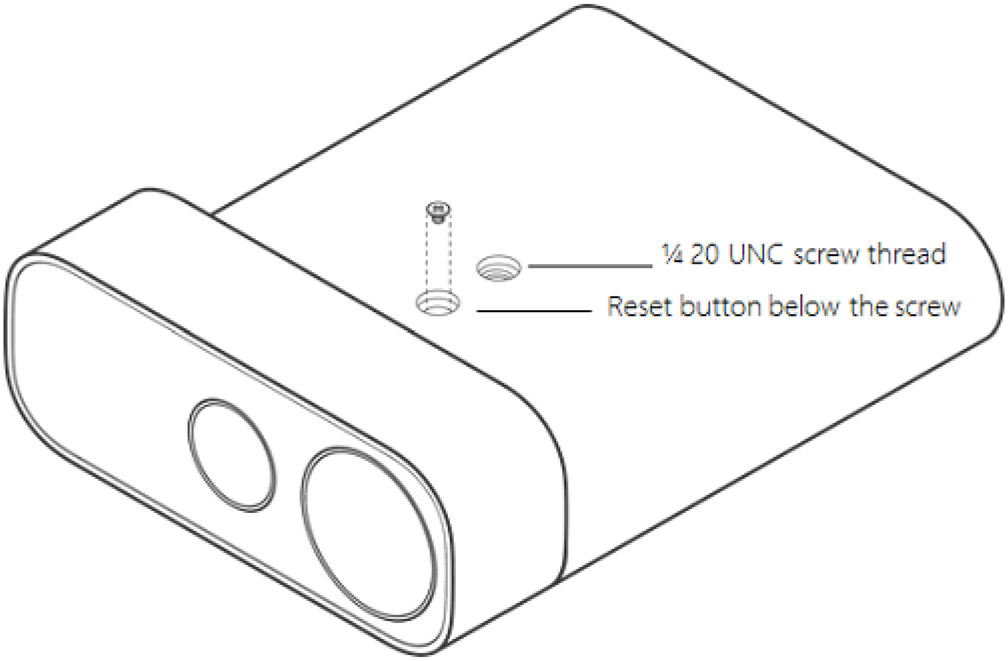
Azure Kinect DK consumes up to 5.9 W; specific power consumption is use-case dependent.

# Calibration

Azure Kinect DK is calibrated at the factory. The calibration parameters for visual and inertial sensors may be queried programmatically through the Sensor SDK.

# Device recovery

Device firmware can be reset to original firmware using button underneath the lock pin.



To recover the device, see instructions here.

## Next steps

- Use Azure Kinect Sensor SDK
- Set up hardware

### Is this page helpful?

Yes No