TRILL

touch sensing
for makers

From the makers of
bela
bela.io

1 Overview

Trill is a family of touch sensors designed by the team at Bela. Trill is designed as a convenient way to integrate capacitive touch sensing into interactive projects.

Trill is compatible with any system that supports I2C communication. Visit `https://github.com/BelaPlatform/Trill` or `https://github.com/BelaPlatform/Trill-Arduino` to download libraries and examples for Bela, Arduino, Teensy, as well as Linux systems such as Raspberry Pi.

To find the complete Trill documentation as well as a Get Started Guide for multiple platforms, go to `https://bela.io/trill`

2 Trill Sensor Types

There are six Trill sensor types: Bar, Square, Craft, Hex, Ring and Flex. Each Trill type offers different physical and sensing affordances:

Trill Type	Sensing Mode	Multi-touch?	Dimensions (W x H)
Bar	1-axis slider	Yes	112mm x 22mm
Square	2-axis pad	No*	74mm x 74mm
Craft**	30-channel breakout	Yes	54mm x 25mm
Hex	2-axis pad	No*	54mm x 54mm (edge to edge)
Ring	1-axis slider	Yes	56mm (outer), 36mm (inner)
Flex***	1-axis slider (included)	Yes	36.3x22.5mm (base), 36mm (inner)

* Pseudo-multi-touch is possible on these sensors but due to the matrix arrangement of pads it is not possible to reliably track the position of individual touches when there is more than one present.
** Trill Craft is a 30-channel breakout board for creating custom touch interfaces out of any conductive material.
*** Trill Flex comes with a single-axis, multi-touch sensor printed on flex PCB. Since the sensor is detachable you can create your own sensors for use with Trill Flex. See the tutorial at [1]

[1] Trill: designing a custom flex sensor `https://learn.bela.io/flex-design`

3 Technical specification

The Trill family of sensors is based around the Cypress Semiconductor CY8C20636A-24LTXI IC, whose datasheet is available at [2].

	Unit	Value	Condition
Operating voltage (Vcc)	V	1.71V to 5.5V*	
Operating current	mA	4	3.3V, 5V
I2C bus speed	kHz	400(max)	

* These are the values reported in the CY8C20XX6A/S datasheet. Trill devices have been tested with Vcc = 3.3V and Vcc = 5V.

4 Pinout

On each Trill sensor there are 6 signals present: VCC, GND, SDA, SCL, RST and EVT.

4.1 About the signals

VCC is where you give power to the sensors. GND is the ground connection of the sensors. SDA is the data line of the I2C bus while SCL is the clock line of the I2C bus. RST is the reset pin: when a voltage is applied to this pin the device will reset. EVT is the event pin which is pulsed every time a scan is completed and each time a touch is detected.

4.2 Trill Bar, Square, Hex, Ring, Flex

The following pinout apply to Trill Bar, Square, Hex and Ring. Each of these sensors includes a Grove connector that attaches to the sensor and ends in pins that can be plugged into a breadboard

Pin	Signal	Color*
1	GND	Black
2	VCC	Red
3	SDA (Data)	White
4	SCL (Clock)	Yellow

* The color listed is that of wire on the provided Grove cable.

4.2.1 Additional connections

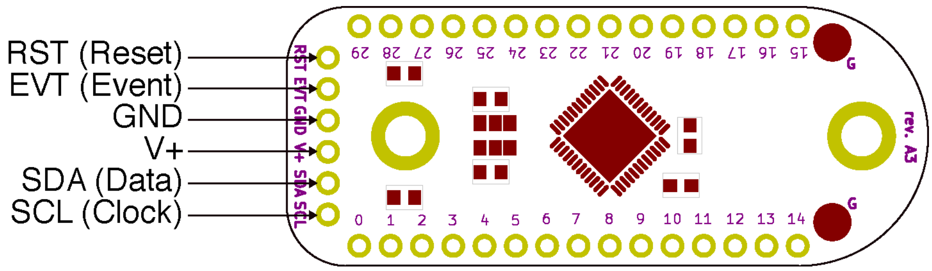
The Event (EVT) and Reset (RST) signals are available as unpopulated, labelled solder pads.

Trill Ring additionally has two pads on the reverse side which can be used as capacitive buttons. These pads behave in the same way as the pins on Trill Craft: just solder a wire to the pads and connect them to any conductive material.

[2]CY8C20XX6A/S datasheet, Cypress Semiconductor company, Document number: 001-54459
https://www.cypress.com/file/138731/download

4.3 Trill Craft

Trill Craft has a total of 30 capacitive sensing pads, one for each channel of sensing (15 pads on each side). Trill Craft also has two SMD pads on the front side labelled "G" which are connected to GND. The power, I2C, RST and EVT signals are available as through-hole pads along the shorter straight edge:



5 I2C protocol

The I2C protocol is used to communicate with the Trill devices using the SDA and SCL lines. These are connected directly to the SDA and SCL lines of the CY8C20636A-24LTXI on the device and no pull-up resistors are provided. More details about the protocol, including timing information, can be found in [3].

5.1 Address selection

The I2C address of a Trill device is determined by the default base address (which is different for each device type) and by the state of the three address pins. These can be either left floating or connected to GND or to Vcc.
The addresses obtainable on each device type are listed below. For details on how to set the address, see [4].

Trill Type	Default	Additional addresses
Bar	0x20	0x21 0x22 0x23 0x24 0x25 0x26 0x27 0x28
Square	0x28	0x29 0x2A 0x2B 0x2C 0x2D 0x2E 0x2F 0x30
Craft	0x30	0x31 0x32 0x33 0x34 0x35 0x36 0x37 0x38
Ring	0x38	0x39 0x3A 0x3B 0x3C 0x3D 0x3E 0x3F 0x40
Hex	0x40	0x41 0x42 0x43 0x44 0x45 0x46 0x47 0x48
Flex	0x48	0x49 0x4A 0x4B 0x4C 0x4D 0x4E 0x4F 0x50

5.2 Communication

In the text below we reference several constants defined in the Trill firmware, available at [5].
At each transaction between the host and the Trill device, the host can directly write or read portions of the Cypress IC's RAM. When writing to the device, the first byte written in a transaction is the memory offset at which the following bytes in the same transaction are written, or the memory offset at which future read transactions will start from.

[3] See footnote 2
[4] Trill: changing sensor address
https://learn.bela.io/products/trill/all-about-i2c/#changing-your-trill-sensors-address
[5] Trill Firmware https://github.com/BelaPlatform/Trill/blob/master/firmware/common/trill-main.h

Memory offsets between `0` and `(COMM_DATA_OFFSET_BYTES - 1)` (inclusive) constitute the command region of the memory. Here, the host can write commands to the device, which will process them and make any response available in the same memory region. The time it takes for the device to parse the message and provide a response depends on the current state of the device. The host should wait at least 10 ms between write operations to ensure each command is parsed appropriately and should wait at least 10 ms between a write and a subsequent read.
Memory offsets between `COMM_DATA_OFFSET_BYTES` and `(COMM_BUFFER_SIZE_BYTES - COMM_DATA_OFFSET_BYTES - 1)` (inclusive) constitute the data region of the memory.

5.2.1 Data format

The format of the data stored in the data region depends on the device type and the mode the device is in. A description of device modes is available at [6].

Trill type	Mode	Data length	Vertical touches	Horizontal touches	Pads
Bar	Centroid	20	5	0	0
Square	Centroid	32	4	4	0
Ring	Centroid	24	5	0	2
Hex	Centroid	32	4	4	0
All	Raw/Diff/Baseline	60	0	0	30

Each pair of bytes represents a 16-bit unsigned integer word in Big-endian format.
Each pad reading is described by 2 bytes (1 word), representing the amount of capacitive activation on the corresponding pad.
In centroid mode each touch is described by 4 bytes (2 words). The first word represent the touch location and the second word represent the touch size. Devices with two axes of sensing (Square, Hex) have vertical touches first, followed by horizontal touches. The readings of the two pads on Ring are stored after the touch values. The special value `0xFFFF` is used for touch numbers higher than the number of detected touches.

5.3 Typical operation

A typical interaction of a host with a Trill device involves writing commands to the command region to set up the device scanning properties, possibly reading back results from the command region to ascertain the device type and its firmware revision number. After this initial setup, the device is prepared for reading data by writing in a dedicated transaction the `COMM_DATA_OFFSET_BYTES` byte, so that successive reads will read from the data memory region. From there on, repeated reads of the appropriate length will return the content of the data region.
Examples of libraries communicating with Trill can be found in [7].

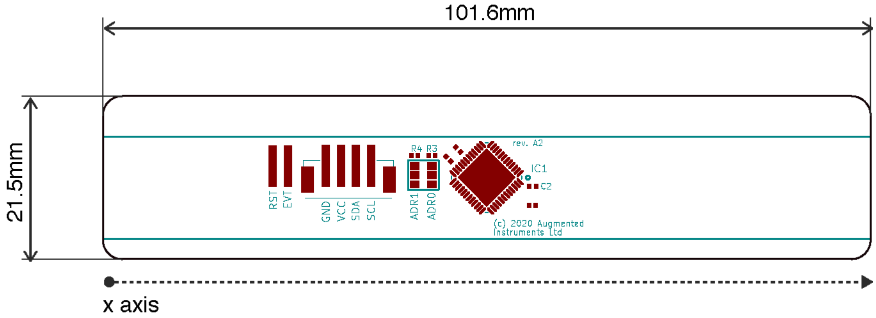
[6] Integrating Trill into your projects
`https://learn.bela.io/products/trill/integrating-trill-into-your-projects/`

[7] Trill Linux `https://github.com/BelaPlatform/Trill-Linux`, Trill Arduino `https://github.com/BelaPlatform/Trill-Arduino`

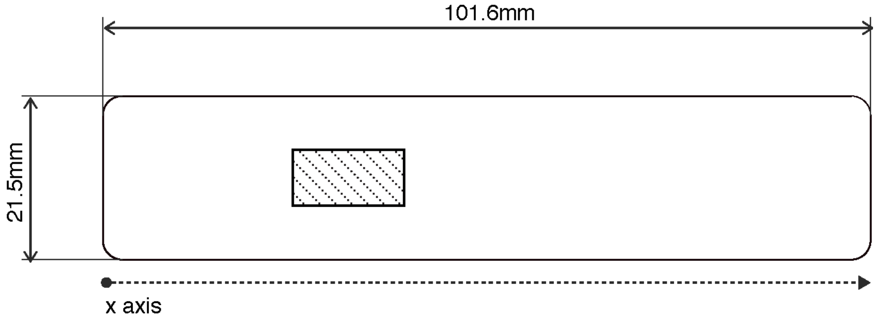
6 Dimensions

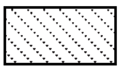
TRILL BAR

REVERSE VIEW



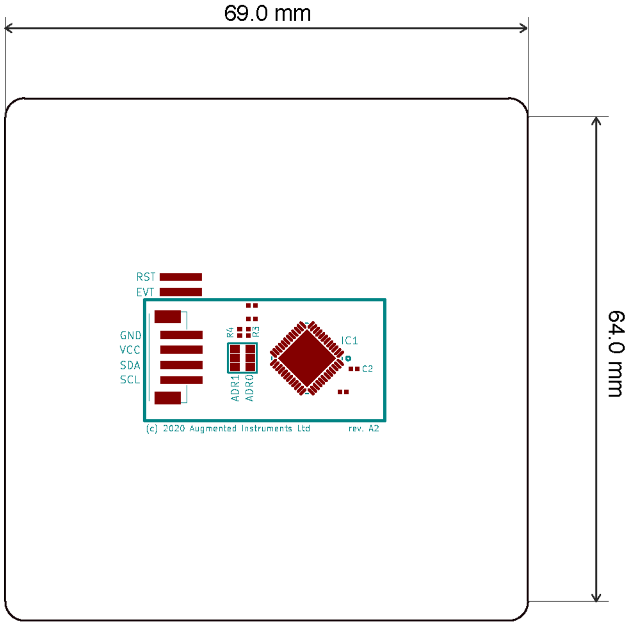
FRONT VIEW



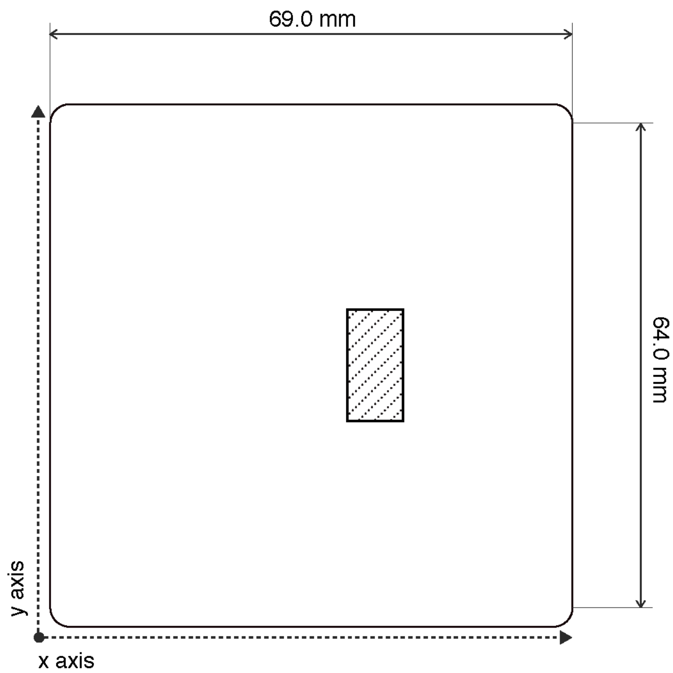
 = Location of Grove connector on reverse side

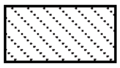
TRILL SQUARE

REVERSE VIEW



FRONT VIEW

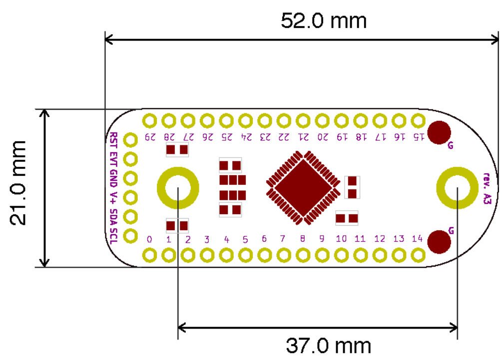




= Location of Grove connector on reverse side

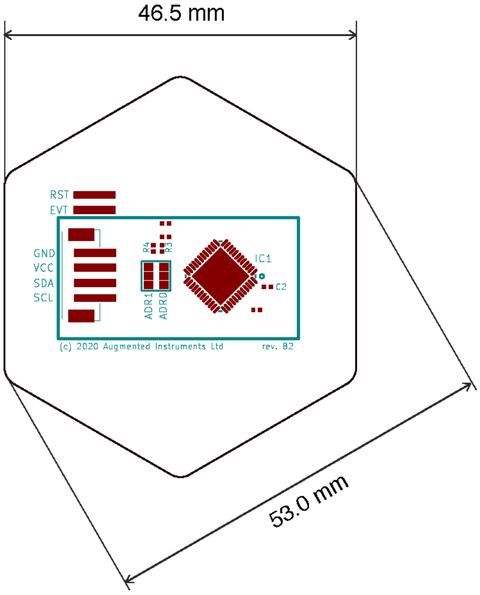
TRILL CRAFT

FRONT VIEW

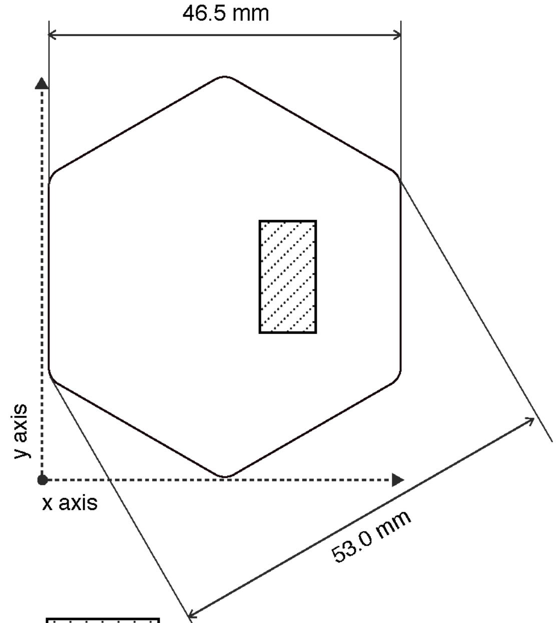


TRILL HEX

REVERSE VIEW



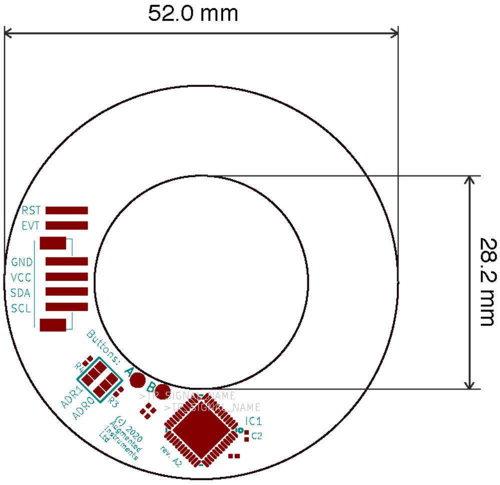
FRONT VIEW



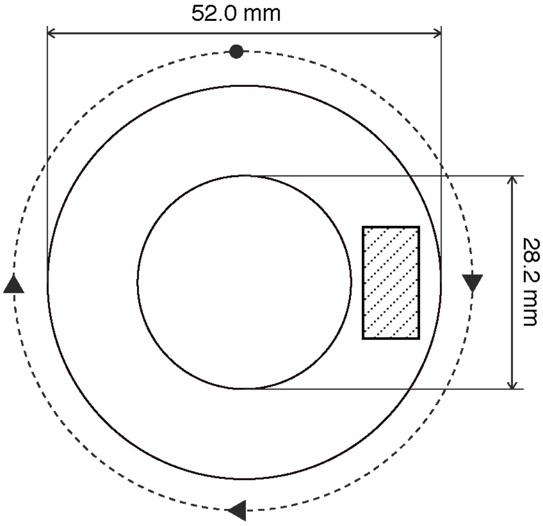
= Location of Grove connector on reverse side

TRILL RING

REVERSE VIEW



FRONT VIEW

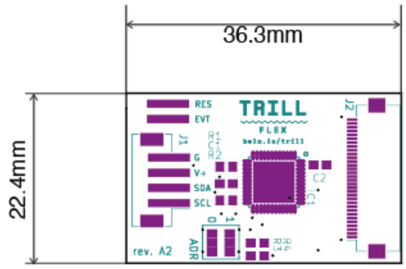


= Location of Grove connector on reverse side

Note: On Trill Ring the sensor reading is 0 at the point of origin, and values increase from 0.0 to 1.0 in the direction indicated.

TRILL FLEX

BASE BOARD



FLEX SENSOR (DETACHABLE)