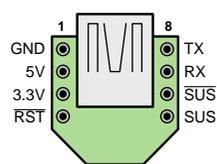
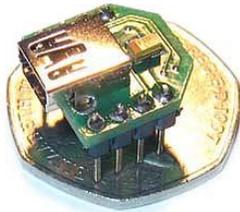


**Overview:** The USBUART device is a complete USB to UART bridge providing the perfect solution for updating RS-232 designs to USB using a minimum of PCB space. Eight pins on a standard 0.1" pitch provide key signals to get you up and running, including 5V and 3.3V power, ground, transmit and receive. The device is completely self-contained, and at a mere 15mm by 15mm may well be the smallest of its kind. The mini-B USB connector sticks over the edge of the PCB enough to protrude out of the side of an enclosure, making the device perfect for use on small runs of boxed prototypes. Royalty-free virtual COM port (VCP) device drivers are provided to allow a USBUART based product to appear as a COM port to PC applications. Baud rates of up to 921600 bits/sec are possible and on-chip EEPROM may be used to customise many product specific variables.

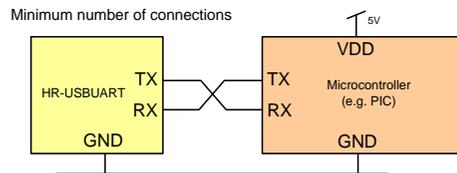
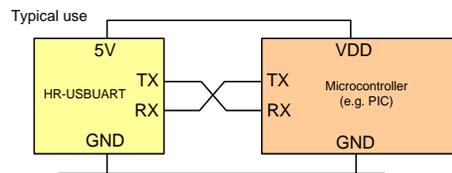


**Pinout:** The USBUART allows asynchronous UART operation without the clutter of extra pins for seldom used handshaking lines. Many microcontrollers have on-chip UARTs which would require a minimum of three pins to connect to the USBUART: TX, RX and GND. The provision of the 5V USB supply on pin 2 allows the USBUART device to power a simple microcontroller system and connect to a USB port with only 4 pins (pins 1,2,7,8). The pinout of the device is shown to the left and the pin

functionality is as follows:

| No. | Name | Description  |
|-----|------|--|
| 1   | GND  | Ground   |
| 2   | 5V   | 5V supply from USB bus   |
| 3   | 3.3V | 3.3V Regulator output (100mA max)  |
| 4   | !RST | Device Reset. Initiate a system reset by driving this pin low for at least 15 $\mu$ s. |
| 5   | SUS  | This pin is driven high when the device enters the USB suspend state.                  |
| 6   | !SUS | This pin is driven low when the device enters the USB suspend state.                   |
| 7   | RX   | Asynchronous data input (UART Receive)   |
| 8   | TX   | Asynchronous data output (UART Transmit)   |

**Practical usage:** The USBUART is designed to fit perfectly into the end of a prototyping breadboard as shown in the photo to the right. The following schematics show how the USBUART device could be used to connect to a microcontroller based system, either in the case where the microcontroller is powered from another source or when it is powered from the USB bus:



**USB controller:** The USBUART contains a USB 2.0 compliant function controller which manages all data transfers between the USB and the UART as well as command requests generated by the USB host controller and commands for controlling the function of the UART. The USB Suspend and Resume signals are supported for power management of both the device as well as external circuitry. The USBUART will enter Suspend mode when Suspend signalling is detected on the bus. On entering

Suspend mode, the SUSPEND and !SUSPEND signals are asserted. SUSPEND and !SUSPEND are also asserted after a USBUART reset until device configuration during USB Enumeration is complete. The device exits the Suspend mode when any of the following occur: (1) Resume signalling is detected or generated, (2) a USB Reset signal is detected, or (3) a device reset occurs. On exit of Suspend mode, the SUSPEND and !SUSPEND signals are de-asserted.

**UART:** The UART interface consists of the TX (transmit) and RX (receive) data signals. Handshaking is not supported. The UART supports a variety of data formats and baud rates. The data format and baud rate programmed into the UART is set during COM port configuration on the PC. The data formats and baud rates available are listed below.

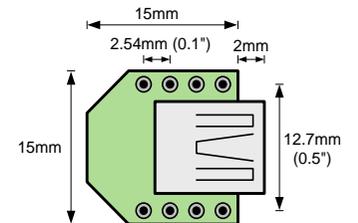
|              |  |
|--------------|--|
| Data bits    | 8  |
| Stop bits    | 1  |
| Parity type  | None, Even, Odd  |
| Flow control | None   |
| Baud rates   | 300, 600, 1200, 1800, 2400, 4800, 7200, 9600, 14400, 19200, 28800, 38400, 56000, 57600, 115200, 128000, 230400, 460800, 921600 |

**Internal EEPROM:** Internal EEPROM may be used to customize the USB Vendor ID, Product ID, Product Description String, Power Descriptor, Device Release Number and Device Serial Number as desired for OEM applications. Customization of the USB configuration data is optional. A unique serial number is given to each USBUART device so that it is possible for multiple devices to be connected to the same PC. Internal EEPROM can be programmed using the chip manufacturer tools.

**Virtual COM port device drivers:** The Virtual COM Port (VCP) device drivers available on the Active Robots website allow the USBUART to appear to the PC's application software as an additional COM port (in addition to any existing hardware COM ports). Application software running on the PC accesses the USBUART as it would access a standard hardware COM port. However, actual data transfer between the PC and the USBUART is performed over the USB. Therefore, existing COM port applications may be used to transfer data via the USB to the USBUART without modifying the application. The drivers for Windows based PCs can be found on the Active Robots website <http://www.active-robots.com> on the USBUART product page. Make sure that you have installed the drivers before plugging in the USBUART for the first time.

**3.3V regulator:** The USBUART includes an on-chip 5-to-3 V voltage regulator which can be used to power external 3V devices. The maximum output current for this regulator is 100mA.

**Physical package:** The USBUART is designed to take up very little space on a prototyping breadboard or PCB. Since only GND, 5V, TX and RX are required for a typical USBUART based application, only the top two pins on either side of the device (1,2,7,8) are required. This means that if necessary, PCB space can be saved by removing pins 3-6. The USB mini-B connector is designed to overhang the edge of the PCB by a couple of mm, allowing it to protrude out of the side of an enclosure if required. The pins are on a 0.1" pitch to allow simple use on a standard prototyping breadboard. Approximate dimensions are shown here, a mechanical drawing is available on our website.



This is not a technical drawing, all dimensions are approximate

**More information:** The USBUART device is based around the CP2101 chip from Silicon Labs. For more detailed technical information and information on drivers for other platforms please consult Silicon Labs.

**Contact:** If you have any problems or questions about this device then don't hesitate to contact us on our website at <http://www.active-robots.com>.