This document’s aim is to help engineers to select the most suitable USB class for a project.

There are many more USB download packages available from Renesas.

See <http://am.renesas.com/support/software/index.jsp>

*USB Full Speed is assumed in this document, and though target devices are supposedly of the RX Family, transfer types and principles discussed are universal.*

## CDC

The CDC Abstract class emulates a serial port; commonly known as a COM port interface from a PC host application perspective . Since COM port communication is a very well-known engineering method to communicate with between electronic equipment (legacy RS232) it is very straightforward for engineers to use.

Renesas has host and peripheral (“device class”) CDC software packages for all USB devices. The PCDC packages emulate a COM port towards a PC. In particluar, the “CDC Flashloader” download (R01AN1869) also has PC host GUI example code showing how to communicate over a COM port using Qt.

## HID

A HID peripheral sends data in one direction, namely “IN” as seen from host. Data is not normally sent in the OUT direction. This can very well be done though, using the control endpoint zero. But observe that a peripheral cannot request specific bus access for control transfers and USB system control may restrict bandwidth for control transfers.

A HID peripheral uses a Report Descriptor to define the IN data. This enables the host application to know how the data is structured. Therefore, if the report format is changed, the host driver need not be updated. The application must instead look at the Report Descriptor to know how to parse the data.

Renesas has both host and peripheral HID download packages for most USB devices.

## *CDC vs. HID transfer differences*

The CDC class uses the bulk transfer type for both IN and OUT data . The HID class uses the Interrupt type for IN data and the control endpoint for any OUT data - though this is not commonly used as we saw above.

The max amount of data that can be sent per frame and per endpoint for both interrupt and bulk transfer types is 64 bytes. This is specified by data member *wMaxPacketSize* of the endpoint descriptor.

For HID, the *bInterval* data member determines how often HID data will be sent to host. It is set in the range 1-64 ms. CDC data will be sent each 1 ms frame as long as there is data waiting in the pipe. That is, data may be sent each ms frame, but with lesser priority. The consequence of this is that jitter will increase within the frame compared to HID.

For more on transfer characteristic differences, see table below.

## MSC

From a communication standpoint, the Mass Storage Class is equivalent to CDC in that it uses one Bulk endpoint each for IN and OUT data. The differece from the host being that is sees a file system instead of using simple commands to send data. Therefore, only reading and writing to whole files is expected.

Renesas has both host and peripheral MSC download packages for most USB devices.

From an application perspective on the peripheral side, this class can provide for local storage of data via a file system library siting on top of the local storage. This can in fact be the same media as the host sees. Each time the peripheral wants to writed data, it mounts the filesystem, write data into a file, then close the file system before releasing the storage to the host ; so a syncing mechanism is needed. Example code which includes a local file system library is available with the Renesas package “*USB PMSC with Local File System and Flash Storage*” for the RX63 (*R01AN1929EU0100)*.

## LibUSB

LibUSB is not a class, but is a very flexible way to create your own solution. This is because 1) you can use *any* transfer type to communicate, and 2) you are not constrained to follow a class specification. LibUSB is a host API set of read write etc commands.

The Renesas download packages “LibUSB - Create a Solution without the Class Struggle” contains a presentation explaining LibUSB in detail. This download is available for both RX63N and RX111, and contains both PC user code using Python and peripheral RX example code, together with training material.



Figure 1. LibUSB explained in detail.

# USB Class Comparison Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Class type** | **HID**  Interrupt IN transfers | **CDC** Bulk IN/OUT transfers  - also applies to **MSC** | **LibUSB** |
| **Typical usage** | Peripheral data input. | Serial COM-port emulation - for which there is most often plenty of legacy code. | When a custom solution can be used, that is, a specific class is not mandatory. (No need to follow about a class spec.) |
| **Speed  (Data thru-put)** | Full-speed HIDs can request bandwidth up to 64 bytes/ms (frame). Each endpoint has one packet of max 64 bytes per frame. Only one endpoint is used in the Renesas code by default. As each frame is one ms we could potentially pump 64\*8\*1000 = 512000 bps over the bus. With two endpoints that would double, but code needs non-trivial modifications. With the maximum 4 interrupt endpoints (PIPE 6-9), we get 4\*512000 = 2 Mbps. Also observe that you on the receiving end need to reassemble the data from all endpoints into the correct sequence. This will cause additional handling complexity at the user level. | Like HID, CDC can request bandwidth up to 64 bytes/ms (frame). As each frame is one ms, that means we could potentially pump 64\*8\*1000 = 512000 bps over the bus. With two endpoints that would double. Though two endpoints are possible, only one is used in the Renesas code by default - the code would need non-trivial modifications. With the maximum 5 bulk endpoints (PIPE 1-5) we get 5\*512000 = 2.5 Mbps. Also, just as with HID, the receiving end needs to handle the multiple endpoints and reassemble the data in the correct sequence, meaning increased handling complexity from the user level. | Any transfer type can be used, though Renesas has no example code for Isochronous transfer.  Same analysis as for HID and CDC applies except we can now use also OUT interrupt endpoints. The default LibUSB download code uses two bulk and two interrupt endpoints, so more endpoints than the previous two combined by default. The host sample code in *LibUSB - A Complete RX USB Function and PC Host Solution* uses both types in the accompanying demo. |
| **Band-width** | bInterval (1-64 ms) specifies how often the data is sent.  HID uses the interrupt transfer type and bus bandwidth is guaranteed. This class is therefore good for devices which need a guaranteed amount of bandwidth with Host. | No guaranteed bandwidth if delayed over occasionally busy hub. | Any transfer type can be used except Isochronous, where same analysis as for HID and CDC applies. |
| **PC host** | The host side drivers are built in to Windows/Linux. Customized host driver is not needed.  The report format can change without changing host driver as the Report Descriptor is sent to host at enumeration.  An INF file stating VID and PID should be provided to end user. | No modified driver necessary, only modification of the Windows INF file to include the proper PID/VID and appropriate string descriptors. | Windows INF file to include the proper PID/VID and appropriate string descriptors. The *libusb0.dll* driver must be in the INF folder of install media and installed to user PC. |
| **Appli- cation layer** | This class is used for mouse, keyboard and other human interface devices.  Report Descriptor needs to be specified and read by host to know how the data is to be interpreted. This does add flexibility since a new driver need not be installed on the host, but does add a layer of complexity for the host unless the standard mouse or keyboard Report Descriptor is not used. | This class can be used for general serial communication. (Compare RS-232 serial).  CDC will be presented as a virtual serial port to the host (Windows/Linux) and serial application program need to be developed.  PC user code is available using Qt. See PCDC Flashloader download package. | PC user code is available using Python code and Python GUI. |

= *Carl Stenquist, April, 2016.*