

VarioHID

TECHNICAL INFORMATION

The VisioBraille VarioHID protocol –
basics, key assignments and
adjustments in NVDA and JAWS

2026-09-22

Content

1. Disclaimer.....	2
2. HID braille basics	3
3. VarioHID, VarioClassic and BRLTTY	3
4. Keys and assignments	4
5. Adjustments in NVDA	6
6. Adjustments in JAWS	7
7. VarioHID on macOS	7
Contact and further information	7

Note on this translation

This document was translated from the German original into English partly automatically, using AI-based translation, and was subsequently reviewed. Despite that review, translation errors cannot be entirely ruled out. In case of doubt, the German version is authoritative.

If you notice an error, please let us know at visiobrasille@lucas-jena.de.

This document is intended for experienced users, dealers and developers who want to use the VarioHID protocol of the Vario 340, the VarioHID 40 and the Vario 4 series. This package contains predefined key assignments for JAWS and NVDA, and it explains everything you need to know about HID braille. Please read this document carefully and pay attention to the files described in it. On ARM-based systems, HID braille may in fact be the only solution that works.

1. Disclaimer

Despite the greatest possible care, LUCAS components GmbH accepts no liability or warranty for fitness for a particular purpose, or for damage caused to your screen reader or your system. Changes should only be made by technically experienced specialists. The reader's consent is expressly assumed.

Prerequisite

For proper operation, at least firmware version 3.0 must be installed on the Vario 340 and the VarioHID 40, and version 3.6 on the Vario 4 series.

2. HID braille basics

HID stands for Human Interface Device. This standard has been used since the introduction of the USB interface for input devices such as keyboards, mice and game controllers, which work natively at the highest system level. Over time the HID standard has been extended further and further, and it now also includes a protocol for braille displays. Such displays can not only be connected without a driver; thanks to a standardised protocol they also work just as easily as conventional input devices.

Until now, braille displays have used specific protocols and interfaces, many of them a virtual COM port (VCP), often with FTDI chips that convert the serial signal for USB. This required interface drivers, and the screen reader had to know the protocols of the braille displays it supported. Whenever a manufacturer changed its design, the protocol in the screen readers had to be adapted as well. Although many braille displays use HID as their transmission standard today, most of them still have their own protocols, and screen readers have been carrying these along for years.

HID braille was developed as a standard above all on the initiative of the large technology companies, which now build screen readers into their devices. It uses the HID interface, but with a standardised protocol, so that braille displays can simply be connected and work, just like mice and keyboards. Screen readers can therefore supply any braille display with data.

HID braille is thus an extension of the familiar HID standard by a generic protocol for braille displays. For further information, see the file `usb-if_hid_press_release_final.pdf` in this folder.

3. VarioHID, VarioClassic and BRLTTY

At VisioBraille we have set ourselves the goal not only of keeping what has

proven itself, but also of helping to shape the future. HID braille will establish itself in the medium term, yet older technologies will still be in use for a long time to come. Linux is also increasingly relevant, so we want to achieve the maximum compatibility for our braille displays.

This is implemented with three operating modes that are quick and easy to switch between. **VarioClassic** uses HID as the interface, but keeps the well-established protocol, so that the braille displays remain compatible with older JAWS versions and with macOS. **BRLTTY** is an additional alternative mode on the Vario 340 and the VarioHID 40, which allows the braille displays to be used with Linux and Android when BRLTTY is installed.

To use the braille displays with **VarioHID**, the latest firmware must be installed: version 3.0 on the Vario 340 and version 3.6 on the Vario 4 series. Older versions do support HID braille, but not VarioHID.

NVDA has supported HID braille natively via USB and Bluetooth since 2021, JAWS and Fusion since version 2026. The mode can also be used with other screen readers such as VoiceOver and the updated Windows Narrator; simply connecting the braille display is enough.

4. Keys and assignments

The HID braille specification defines the possible keys and divides them into groups. Manufacturers can choose from this pool according to the type of key, and there are also variables for individual key names. In addition, HID braille offers actions: command sequences that can be permanently assigned to keys. The braille display can, for example, pass on the action that the screen reader should switch from computer braille to contracted braille and back. The screen reader then does not have to know the key; the braille display tells it what to do. Depending on their design, manufacturers can therefore transmit either keys or command sequences.

Our challenge was to adapt the display keys to HID braille with two goals in mind. First, the keys should work sensibly even without a specific assignment; second, the proven functions should remain available to the user as usual. The available assignment types are joystick, rocker, D-pad, pan left and pan right for scrolling, and freely assignable keys. The latter would require an assignment table in any case.

It seemed plausible to us that D2 and D5 should be assigned to pan left and

pan right. But what to do with the remaining display keys? After some experimentation it was clear that we would assign them as a D-pad. Digital pads are familiar from game controllers; they are moved in four directions, and pressing them is even provided for in the HID braille standard. A D-pad is always used for directional input, and even when a specific key combination is loaded, nobody would be likely to press a D-pad in all directions at once on another product. At the same time, a modern Vario braille display works in its basic functions even without key assignments. The result is maximum compatibility without interfering with other products.

Screen readers usually query the name of the connected braille display and can load a specific key assignment on that basis. So that all braille displays from VisioBraille work with the same assignment pattern, we decided on a generic name by which all current and future braille displays identify themselves to the screen reader: **VarioHID**.

Vispero will implement the key assignment in future JAWS versions, and other manufacturers will do the same. The screen reader then only has to hold the up to 57 additional key combinations for the Vario to work as before. At least up to NVDA 2026 there are no plans to query the name of the braille display; there you have to do it yourself, but you then get the familiar convenience with HID braille as well. This is a good example of why the VarioClassic mode makes sense: with it, the Vario braille displays work with NVDA as usual. For developers and in a professional context there is currently no need for HID braille. With JAWS/Fusion 2026 and later, however, you save yourself the dedicated JAWS extension, and a current Vario braille display works straight away.

Unlike the display keys, VarioHID brings differences that have to be taken into account: not only the new names, but also the evaluation in reverse order.

Current key assignment

Key	Official designation	Internal designation
D1	D-pad up	DPadUp
D2	Pan left	PanLeft
D3	D-pad down	DPadDown
D4	D-pad left	DPadLeft
D5	Pan right	PanRight
D6	D-pad right	DPadRight

If you set up a key assignment yourself in a screen reader, the new key names from the “Internal designation” column are shown instead of D1 to D6. In the keyboard help you can determine whether VarioHID or VarioClassic is in use by pressing the various key patterns. The Vario 340 and the VarioHID 40 show the currently selected mode on the right-hand side of the display. You switch to VarioHID by holding down display key D4 while switching the device on. On the Vario 4 series you use the menu; in firmware 3.6 you can choose between VarioHID and VarioClassic.

If you use JAWS and NVDA in parallel, you should settle on VarioHID and equip both screen readers with the new assignments. If you switch between macOS and Windows, we recommend using VarioClassic with the additional extension, because on macOS the D-pad is currently only assigned to basic functions.

Note for developers

The order in which keys are evaluated differs between the protocols. VarioClassic passes the D keys as D1+D2+D3+D4+D5+D6; with VarioHID it is the other way round. The highest key always comes first, in this case DPadRight+PanRight+DPadLeft+DPadDown+PanLeft+DPadUp.

When you adjust an assignment list by hand, this order has to be observed, otherwise the assignment is not recognised.

5. Adjustments in NVDA

For the key assignment we took our lead from the SuperVario, including the second routing level. With D1+routing you can have the formatting information under a character announced. The assignment is stored in the file “gestures.ini”.

In principle all you have to do is copy this file into the NVDA user directory (%AppData%\nvda), or paste its contents into an existing file of the same name. If the file does not exist, the additional assignment is taken into account the next time NVDA is started. If key combinations have already been defined and the file already exists, copy the contents into the existing file. If an assignment for another braille display already exists, it is not

overwritten. At least up to version 2026.1, NVDA does not check whether an assignment already exists.

6. Adjustments in JAWS

In JAWS version 2026 you have to make sure that “HID braille” is enabled in the braille display settings. Only then is an HID braille display detected automatically, although with the default assignment. In future JAWS versions VarioHID will be integrated at the factory.

The file “JAWS.txt” contains the additional key assignments, which you have to insert at the end of the file “default.jkm”. You will find it under %AppData%\Freedom Scientific\JAWS\2026\Settings\enu.

Alternatively you can replace the existing file with the enclosed “default.jkm” from JAWS 2026, which already contains the changes. This also enables the second and third routing levels in combination with D1 for formatting information and D3 for selecting text.

7. VarioHID on macOS

Apple has announced that no new braille drivers will be adapted or integrated. Anyone developing a braille display for macOS today has to support HID braille. Accordingly, the Vario braille displays work with both protocols; with VarioClassic the D keys are used with the familiar assignment.

With VarioHID, D1, D3, D4 and D6 can currently only be used for the arrow keys, but you can define your own key assignments in the VoiceOver settings. Because VarioClassic works at present and has no disadvantages, we currently see no need for action. Nevertheless, we have informed Apple of the changed key assignments and asked for them to be integrated into VoiceOver. Experience suggests that this is a process whose end is not foreseeable at present (May 2026).

Contact and further information

LUCAS components GmbH

Address: Max-Grossmann-Straße 1, 07745 Jena, Germany

Phone: +49 3641 6686 133

Email: visiobracille@lucas-jena.de

Website: www.visiobracille.de