

External control via serial interface

Besides transmitting data in any of the supported simulator software formats, it is also possible to “control” the display by sending serial commands in human readable text format. Any serial communications software/ terminal could be used to send commands. Perhaps you would like to build a script or your own software to display data. In this case you may find this document useful.

In order for this interface to apply, select either “External” or any of the “X-Plane” or “FS2020” options under the main Configuration menu → Other → Sim Mode.

After plugging into a USB port, Horizon Clock is automatically identified as a serial communications device under all common operating systems like Windows, Linux or Mac. (Driver files are usually not required because normally the installation is automatic, but if you need to install drivers just download them from the download section of our site). Any application can access the device’s port as a common serial connection. Communication must be setup with these parameters: Speed: 500000bps, 8bit data, 1 stop bit, no parity, no flow control. Data update via the serial interface is slower than any of the other simulators’ interfaces and will not achieve high frame update rates, however it will maintain full user functionality and continue to display time, temperature and humidity data.

Each command starts with “&&” (that is two & characters without the quotes) followed by two letters. For some commands the device responds with “RD” (Ready) when the command is accepted and for others it echoes back the two command letters to confirm reception. After the command executes successfully it responds with “OK”. On error some commands respond with “AB” (abort). See the table below for all supported commands. Note that all commands timeout very fast (usually less than a second), so they need to be sent as complete packets of data compared to human-typed data (as in a terminal, in which case you could try to copy and paste complete command strings instead of typing).

Command	Title	Description/ Device Response/ Details
&&AG	Get device version number	Response is “HCxxxxxxx” where x is version id characters.

Command	Title	Description/ Device Response/ Details
&&AS	Set clock time and date	Responds with "RD" if the device is ready to accept the new time and date data. After RD is received, send the new time and date in the format "hhmmssYYYYMMDD" where h:hours, m:minutes, s:seconds, Y:year, M:month, D:day. Time is expressed in 24-hour format.
&&AM	Get most recent measurements (temp. and rel. humidity)	Response is in format "sTT.T HH.H" where s: sign character (either + or -), T: temperature digits and H: rel. humidity digits
&&FL	Stop auto flight mode	Stops updating virtual flight data (as if the autopilot was turned off). Echoes back "FL" when command is received and responds with "OK" when successful. Stopping the autopilot is required before updating flight data (like speed, altitude etc). The autopilot will re-engage after about 60 seconds if no commands are received.
&&FE	Resume auto flight mode	Resumes updating flight data. Also removes any text previously set by command &&FT. Echoes back command letters "FE" when command is received. Responds with "OK" if command is accepted.
&&FT	Display text on bottom of AHRS screen	Echoes back command letters "FT" when command is received. Next waits for exactly 22 ANSI characters. For smaller length strings, add padding with spaces. Only capital letters, numbers and these symbols: ! " # \$ % & ` () * + , - . / : ; < = > ? @ [\] ^ _ are allowed. Non supported symbols will not be displayed. The device responds with "OK" after data is accepted. This text is cleared when there is user interaction or command &&FE is sent.

Command	Title	Description/ Device Response/ Details
&&FS	Update speed	Next 3 characters (bytes) after command are speed digits. Send the special characters "III" to display a magenta X-sign over speed instead (speed not available or sensor malfunction). Echoes back command letters and responds with "OK" on success or "AB" (abort) on error.
&&FA	Update altitude	Next 5 characters (bytes) after command are altitude digits. Besides digits, the minus sign is also accepted for negative values and "IIIII" displays a magenta X-sign (altitude not available or sensor malfunction). Echoes back command letters and responds with "OK" on success or "AB" (abort) on error. Example communications (sets altitude at 5100 feet): Input (towards device): &&FA05100 Output (from device): FAOK
&&FV	Update vertical speed	Next character (byte) must be either "0" (climb) or "-" (descent). Next 2 characters are digit numbers "00" to "30" (maximum). Echoes back command letters and responds with "OK" on success or "AB" (abort) on error.
&&FB	Update slip ball position	Next character must be either "0" (right movement) or "-" (left movement). Next 3 characters are "000" (no movement) to "127" (maximum position). Echoes back command letters. Responds with "OK" on success or "AB" (abort) on error.
&&FR	Update roll angle	Next character must be either "0" or "-" (right/left roll). Next 3 characters are an integer value ranging "000" to "720" as the roll angle in 0.25 degree steps (720 equals $720 \times 0.25 = 180$ degrees). Echoes back command letters. Responds with "OK" on success or "AB" (abort) on error.

Command	Title	Description/ Device Response/ Details
&&FP	Update pitch angle	Next character must be either "0" or "-"(upwards/ downwards pitch). Next 3 characters are an integer value ranging "000" to "360" as the pitch angle in 0.25 degree steps (360 equals $360 \times 0.25 = 90$ degrees). Echoes back command letters. Responds with "OK" on success or "AB" (abort) on error.
&&FH	Update heading	Next 3 characters are heading digits from "000" to "359". Value "000" is displayed as 360. Echoes back command letters. Responds with "OK" on success or "AB" (abort) on error.