
Piccolo system user guide



8 Fourth Street
PO Box 1500
Hood River, OR 97031
(541) 387-2120 phone
(541) 387-2030 fax
CCT@gorge.net

Bill Vaglianti
Ross Hoag

February 26, 2004
Version 1.1.9

Table of Contents

1	Introduction.....	4
1.1	Features	4
1.2	CPU	5
1.3	Sensors	5
1.4	Servos.....	5
1.5	Datalink.....	5
1.6	Payload interface.....	6
1.7	Expansion.....	6
1.8	PiccoloPlus.....	6
2	Avionics	6
2.1	Front panel connections	6
2.2	Power options.....	9
2.3	Servo options	9
2.4	Installation location and orientation.....	9
2.5	Air data system	9
2.6	GPS antenna.....	9
2.7	Communications antenna.....	10
3	Ground Station	10
3.1	Power	10
3.2	Connections.....	10
3.3	Pilot in the loop.....	10
4	Operator Interface	11
4.1	Installation and running	11
4.2	Running.....	11
4.3	Ground station screen	12
4.4	Avionics Window	13
4.5	Telemetry page.....	14
4.6	AP commands	16
4.7	Map	18
4.8	Preflight.....	21
4.9	Autopilot limits	22

4.10	Sensors	23
4.11	Surfaces.....	25
4.12	Autopilot gains.....	29
4.13	Payload page	30
5	Development environment.....	33
5.1	Software environment.....	33
5.2	Design environment.....	33
5.3	Hardware in loop simulation.....	33

1 Introduction

Piccolo is a complete integrated avionics system for small unmanned aircraft. The system includes avionics hardware and software, ground-station hardware and software, and a development and simulation environment. This document is the end-user's manual for the *Piccolo* system, and provides a basic description, interface guidelines, and system setup.

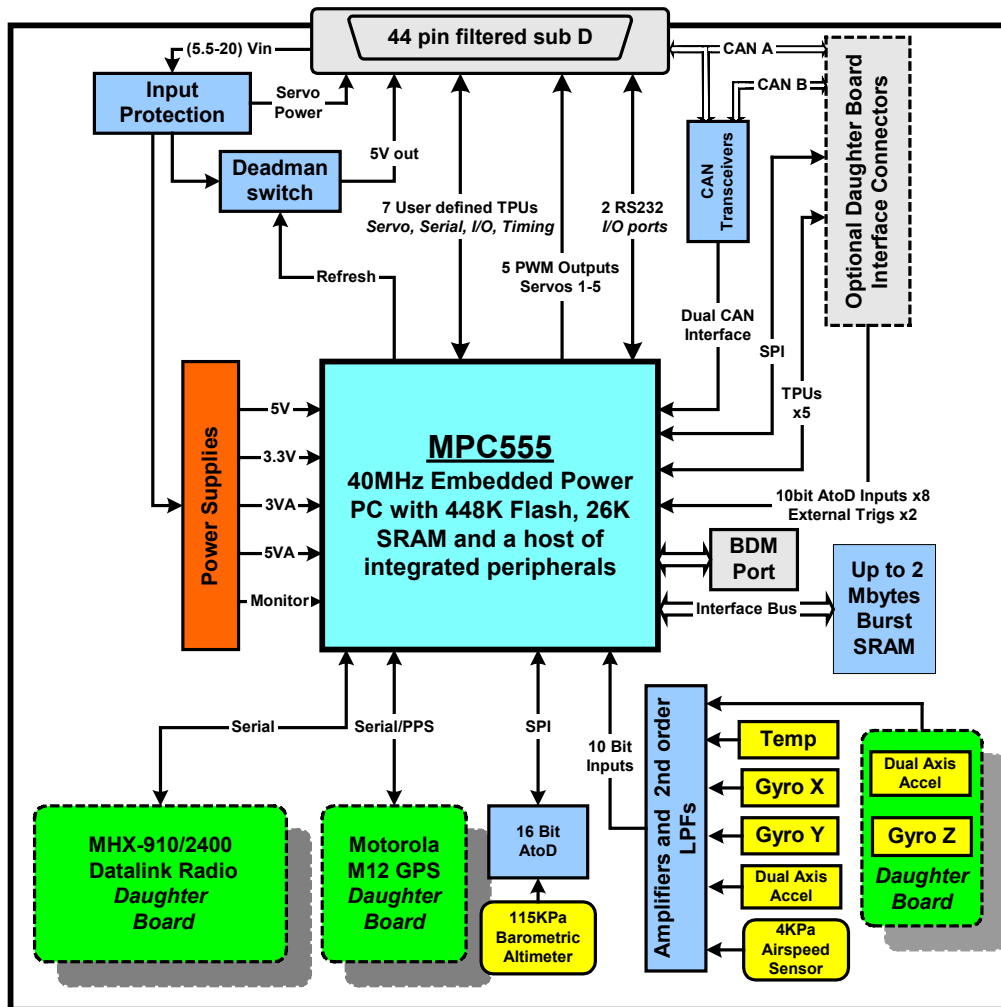


Figure 1. Piccolo block diagram

1.1 Features

- Exceptional integration and processing power from the MPC555 main processor.
- Small size: 4.8" x 2.4" x 1.5"
- Weight 212 grams (7.5oz) with radio, GPS and enclosure. 90 grams (3.1oz) without radio and enclosure.
- 3-axis rate and acceleration measurements.
- Deadman's switch for safety.
- Expandable using the external interface or daughter board connector.

- Complete hardware integration including sensors, datalink, and servo drivers.
- Wide range of software support, including complete end-user programmability.
- Reference software design for ground station and simulation tools.
- Fully autonomous or pilot in the loop operation.
- Attitude independent installation.

1.2 CPU

The heart of *Piccolo* is the MPC555 microcontroller. This is a new breed of automotive controller based on the PowerPC architecture. The MPC555 marries an enormous array of interfaces to a powerful RISC engine that delivers 40Mhz PowerPC operation, including hardware floating point.

1.3 Sensors

Piccolo features three Tokin CG-16D rate gyros and two two-axis ADXL202 accelerometers, which combined with its CPU horsepower gives *Piccolo* the ability to run sophisticated filters; and allows installation in any attitude. The *PiccoloPlus* uses three ADXRS300 gyros and two two-axis ADXL210e accelerometers.

Piccolo includes a dual ported mpxv50045 4kPa dynamic pressure sensor, an absolute ported mpx4115a barometric pressure sensor, and a board temperature sensor. Together these sensors provide the ability to measure true air speed and altitude.

The Motorola M12 GPS provides *Piccolo* with its basic groundspeed and position. The M12 is differential capable, and *Piccolo* will supply it with any DGPS corrections received over the datalink. The *Piccolo* ground station is capable of generating these corrections using its own M12.

Piccolo can generate an estimate of winds aloft each time the aircraft turns by fitting true airspeed against GPS groundspeed. The wind estimate is useful since it enables an intelligent selection of heading when the wind speed exceeds the flight speed of the aircraft.

1.4 Servos

Piccolo includes 10 servo outputs. These outputs are geared for model aircraft servos, which require pulse-width modulated signals, and 4-6 volt power. Alternatively some servo systems use a CAN (Controller Area Network) bus to provide the servo signals. *Piccolo* includes two independent CAN buses. In either case the servo power supply is isolated from the main logic power supplies so that transient servo loading doesn't disable core system functions.

1.5 Datalink

Piccolo includes a sophisticated datalink that is built on the MHX 910/2400 radio modem from Microhard Systems Inc. The datalink has up to 40Kbaud of throughput and is used for command and control, autopilot telemetry, payload data transfer functions, differential GPS corrections uplink, and pilot in the loop modes. The datalink architecture allows multiple aircraft to be controlled by a single operator from a single ground station.

1.6 Payload interface

An unmanned aircraft is only as useful as its payload. Accordingly *Piccolo* provides several means of connecting to payloads, including CAN, asynchronous serial, and discrete I/O. Data received from the payload(s) can be downlinked over the main datalink.

1.7 Expansion

In addition to the interfaces available on the external connector *Piccolo* includes two daughter board connectors for user-defined applications. These connectors provide analog, SPI, timing, serial, and CAN methods of attaching hardware.

1.8 PiccoloPlus

PiccoloPlus is the next generation of the *Piccolo* system. The *PiccoloPlus* was introduced in early 2004 as a form-fit-function replacement for the *Piccolo* avionics. It utilizes new inertial sensors to improve the performance and robustness of the system. For the purposes of this manual the *Piccolo* and the *PiccoloPlus* are equivalent units.

2 Avionics

Avionics refers to the airborne component of the *Piccolo* system. It is the most complicated part of the system and requires the most care in setup. Most of the setup is actually done through the operator interface software, and is described in that section. This section deals with the physical installation of the avionics.



Figure 2 - Piccolo Front Panel

2.1 Front panel connections

All connections to the *Piccolo* are through the front panel as shown in Figure 2 and include the filtered 44-pin vehicle interface connector, GPS and UHF antenna SMA connectors as well as the Pitot and Static pressure port nipples which are compatible with 3/32" ID tubing. Table 1

outlines the functionality of each pin on the external interface connector. A complete flight harnesses reference schematic with power and servo leads is available for review on our [downloads](#) page.

Table 1 - Vehicle Interface Connector Pin Out

PIN	NAME	TYPE	LEVEL	FUNCTION	
15	GND			Gnd for servo 1	1
30	SERVO_Vout		*	Power for servo 1	
44	PWMSM[0]	O	5V	Servo 1 signal output - Left Aileron	
14	GND			Gnd for servo 2	2
29	SERVO_Vout		*	Power for servo 2	
43	PWMSM[1]	O	5V	Servo 2 signal output - Left Elevator	
13	GND			Gnd for servo 3	3
28	SERVO_Vout		*	Power for servo 3	
42	PWMSM[2]	O	5V	Servo 3 signal output - Left Throttle	
12	GND			Gnd for servo 4	4
27	SERVO_Vout		*	Power for servo 4	
41	PWMSM[3]	O	5V	Servo 4 signal output - Left Rudder	
11	GND			Gnd for servo 5	5
26	SERVO_Vout		*	Power for servo 5	
40	PWMSM[4]	O	5V	Servo 5 signal output - Left Flap	
39	TPU_A[0]	I/O	5V	Servo 6 signal output - Right Aileron	SPARE I/O
38	TPU_A[1]	I/O	5V	Servo 7 signal output - Right Elevator	
37	TPU_A[2]	I/O	5V	Servo 8 signal output - Right Throttle	
36	TPU_A[3]	I/O	5V	Servo 9 signal output - Right Rudder	
35	TPU_A[4]	I/O	5V	Servo 10 signal output - Right Flap	
25	USER_Vout	O	*		PWR/GND
24	USER_Vout	O	*		
23	USER_Vout	O	*	Servo power input 4.8-6V DC (4 or 5-cell pack)	
6	SWITCHED_USER_Vout	O	*	Switched Output Voltage (Driven by Deadman's Switch)	
10	GND			Ground	
9	GND			Ground	CAN
8	GND			Ground	
7	CAN_GND			CAN Ground	
22	CAN_HI_A	I/O	CAN	CAN A Serial High	SERIAL COMMS
21	CAN_LO_A	I/O	CAN	CAN A Serial Low	
34	TXD_RS232	O	232	User RS232 Output - TPU_B[0]	
33	RXD_RS232	I	232	User RS232 Input - TPU_B[1]	
5	TPU_B[2]	I/O	5V	User Configurable I/O	
20	TPU_B[3]	I/O	5V	User Configurable I/O	
4	GND				
19	*PROGRAM/USER	I	5V	Program/User Mode Control Input - MPIO32B5	
18	*HRESET	I	5V	Hardware Reset - active low	
3	GND			Ground	
32	SCI_2_TX_232	O	RS232	Program Port/Ground Station Interface/User Serial	
31	SCI_2_RX_232	I	RS232	Program Port/Ground Station Interface/User Serial	
16	NC1			NC	
1	VIN			Main DC Input - 5.5-20 Vin	
2	GND				
17	NC2			NC	

* - These pins are ganged together onboard the Piccolo. Typically one is used as the servo power input with the rest available as outputs to power servos or payloads.

2.2 Power options

Piccolo accepts 8 to 20V DC (3.6 Watts at 12 Volts nominal) input power. This power is regulated down to internal voltages required for the main board. A second input, 4.8 to 7V DC, is used to power the servos (either a 4 or 5-cell battery pack can be used). It can be applied to any one of the unused servo power or user_vout pins (we typically use pin 23).

2.3 Servo options

Piccolo has ten PWM (Pulse Width Modulation) outputs available for servos. There are two basic modes of operation, 5-channel mode, and 10-channel mode. Most systems will use 10-channel mode. In either mode individual servo control signals can be reassigned to other payload control functions. Details of the different servo options can be had from section 4.11.

2.4 Installation location and orientation

There is no limitation on the installation location of *Piccolo*. However in general you should choose a location that will minimize vibration (see the “[Vehicle Integration Guidelines](#)” document for further installation details). Piccolo’s basic autopilot system does not use the accelerometers. However if you plan to use the accelerometer in your application you will want to mount the avionics close to the center of gravity in order to reduce the acceleration that results from centripetal motion and changes in angular rate.

The orientation of the avionics in its final installation is important. The avionics has a natural coordinate system, which is marked on the front panel. The coordinate system uses English aircraft conventions, with the X-axis pointing through the nose, the Y-axis out the right wing tip, and the Z-axis pointing down. You do not have to install the avionics in its natural coordinate system; however the Euler angle rotations between the avionics coordinate system and aircraft coordinate system must be known.

2.5 Air data system

Piccolo has pressure port inputs for total (or pitot) pressure and static pressure. A 4KPa differential pressure sensor measures the difference between the total pressure and static pressure to determine the dynamic pressure. A 115KPa absolute pressure sensor measures the static pressure. An accurate air data system is critical to the performance of Piccolo. Both the dynamic and static pressures are used in the primary control loops. In addition the dynamic pressure is used to calculate the true air speed, which is used for wind measurement.

The total and static pressure ports should be plumbed to pressure probes (see “[Vehicle Integration Guidelines](#)” document for further details).

2.6 GPS antenna

The GPS antenna should be mounted on the aircraft where it has a clear view of the sky. In particular consider how the antenna’s view of the sky will change with aircraft pitch and roll motion. Ideally the antenna should be mounted on a large metal groundplane, however satisfactory performance can be had with a small (a few inches across) groundplane made of foil or metal tape on the aircraft.

2.7 Communications antenna

A 900MHz or 2.4GHz ISM band $\frac{1}{4}$ wave or similar antenna will need to be installed on the host airframe. You should try to maximize the separation between the GPS and UHF antennas to minimize potential interference. Again refer to the “[Vehicle Integration Guidelines](#)” document for examples and further details.

3 Ground Station

Piccolo’s ground station is based upon the same hardware that makes up the avionics package. It manages the communication link to one or more avionics systems, interfaces to the pilot in the loop console, and provides a command and control stream to the operator interface PC.

3.1 Power

The ground station is powered through the included power supply that provides DC power. A battery is included in the ground station for backup operation in case the main input power is lost. When main power is connected the battery will be charging (Yellow LED). The ground station will not be on until the external switch is turned on (Green LED).

3.2 Connections

The Ground Station connects to the operator interface PC through a standard 9-pin serial cable. The ground station GPS Antenna is connected to the rear panel SMB connector and the ground station UHF antenna is hooked up to the BNC connector. The pilot console connects through the included 6-pin circular DIN pilot console cable. Please refer to the *Quick Setup Guide* for photos and a complete descriptions of each component.

3.3 Pilot in the loop

Pilot in the loop commands are input to the ground station through a Futaba compatible buddy-box 6-pin DIN connector. The ground station will attempt to decode 6 pulses in the serial train. They are: aileron, elevator, throttle, rudder, autopilot on/off (gear), and flaps. The fifth channel, usually used for retract landing gear, is used to select autopilot on/off status. A pulse longer than 1.5ms on this channel selects autopilot off. A short pulse selects autopilot on.

The power switch on the Futaba pilot console should be left off; it will receive its power from the ground station. In addition the RF module of the pilot console can be removed since it serves no purpose in this application.

4 Operator Interface

The operator interface (OI) is a software system that runs on a Windows PC and provides a command and control interface for Piccolo operators. Most of the system features are accessible from this interface, and so we will use this part of the user's guide to describe many of the basic capabilities of the system.

4.1 Installation and running

The operator interface must be correctly installed before it can be used. A windows installer¹ file, "Operator Interface.msi" is provided for this. Double click on the file to launch the installer. The onscreen instructions are simple to follow and amount to choosing where you'd like the operator interface to be installed.

4.2 Running

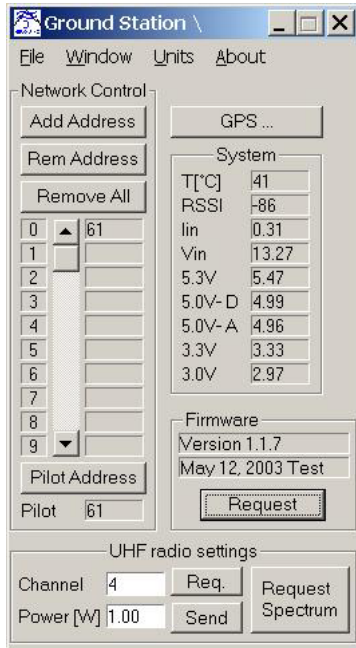
Once installed the operator interface executable can be run like any windows application. In its default mode it expects to find the ground station connected to serial port 1 of the PC. However the mode of operation can be affected with command line switches

Table 2. Command line switches

Switch	Meaning
-P=	Controls the serial port that the OI will use to connect to the ground station. For example -P=2 will cause the OI to look for the ground station on COM2.
-B=	Changes the baud rate used for the serial connection. The nominal connection to the ground station is done at 56K (57600). However specialty applications may require a different baud rate.
-LISTEN=	Enables the OI to act as a server. In this mode the operator interface will listen for incoming TCP connections and send and receive Piccolo packets over the connection. The port number used for the listening mode is passed on the command line, for example -LISTEN=2000 will cause the OI to listen for clients on port 2000.
-SERVER=	Enables the OI to connect to a Piccolo data server (which could be another OI in server mode). The data passed on the switch controls the computer name and port number that the OI will try to connect to. For example -SERVER=MyServerName:2000 will cause the OI to initiate a TCP connection to a computer call MyServerName, on port 2000. This switch cannot be used at the same time as -LISTEN or -P.
-R=	Enables the OI to replay a binary log file. The data passed on the switch is the name of the binary log file (usually *.tel). This switch cannot be used with any of the other switches except -LISTEN.

¹ The windows installer is a relatively new component of Windows. Windows 2000 and XP include the windows installer; however older versions of Windows must have the installer installed (ironic, I know). If your version of Windows does not recognize the .msi extension it is because you need the windows installer. Use the WinInstall9X.exe or the WinInstallNT.exe files to install the installer.

4.3 Ground station screen



The operator interface communicates to the ground station over a serial link (default to COM1). When the OI starts the first page which is visible is the ground station screen.

4.3.1 Window Menu

The window menu has options for minimizing or restoring any avionics windows in the system, for selecting any avionics window, and for cascading or tiling the avionics windows. As of version 1.1.4 the window menu also gives the option for an advanced versus simple view. The simple view hides the primary system setup windows on the avionics window.

4.3.2 Units Menu

The units menu allows the user to select the units that telemetry are displayed in. Angles can be displayed in decimal degrees, degrees minutes seconds, or in radians. Anything which is not an angle can be displayed in either Metric or English units.

Figure 3. Operator interface ground station window

4.3.3 GPS

The GPS button invokes a dialog that gives a display of the current ground station GPS data, and the differential base station setup.

4.3.4 Network Control

The ground station manages simultaneous communications to multiple avionics by using a polling scheme in which it periodically requests communications from each avionics in a list. The list is composed of *Piccolo addresses*, which are the serial number of each avionics.

The Add Address button can be used to send a message to the ground station telling it to add the requested serial number to its polling list. The Pilot Address button works like the Add Address button, except it tags the address to receive manual pilot commands from the pilot console. Only one pilot address can exist in the system. The Remove Address button is used to remove addresses from the list, excepting the pilot address. Remove All will remove all addresses from the list, again excepting the pilot address. The ground station stores the list of serial numbers in non-volatile memory so any changes made to the list are not lost when the ground station is powered down.

Note that each address in the list will occupy a certain amount of communications bandwidth, whether the avionics in question actually exists or not. Hence the list should not contain addresses that you do not wish to talk to. In practice there is only enough bandwidth for about 10 avionics at one time.

4.3.5 System

The system group contains diagnostic data from the ground station. T[°C] is the temperature of the main board in degrees Celsius. The boards have been tested from 0 to 80°C. RSSI is the receive signal strength indicator from the MHX radio. It is in units of dBm and ranges from -71

(strong signal) to -115 (no signal). When communicating with multiple avionics the `RSSI` field will be a composite of the different signal strengths. `Iin` refers to the input current to the ground station, and is displayed in Amps. `Vin` is the input voltage, which should always be near 12 Volts for the ground station. The remaining system data are only visible in advanced mode. `5.3V` is the voltage after the first stage regulator, and should nominally be between 5.3 and 5.5 Volts. `5.0V-D` and `5.0V-A` are the digital and analog five volt rails respectively. `3.3V` is used to run the MPC555 and memory. `3.0V` is used to run the M12 GPS.

4.3.6 UHF radio settings

The UHF radio settings allows the user to control the power output and the hopping pattern of the Microhard radio in the ground station. The radio is spread spectrum, which means that it doesn't use a single frequency, but instead hops along a pattern of frequencies. In order to operate multiple ground stations in close proximity they must be using different hopping patterns to avoid interference. Pushing the `Req` button will request the current hopping pattern and power output of the radio. Pushing the `Send` button will update the hopping pattern and power output of the radio.

The request spectrum button is used to invoke a special feature of the radio which allows it to operate as a spectrum analyzer. The radio will sample all of its channels and return a text based result which allows the user to see the background noise level on every channel. Note that when any of these radio features are invoked the radio will be taken offline, suspending communications for a short while.

4.4 Avionics Window

When data are received from a Piccolo avionics a new window is created which is specific to that avionics. The window is built upon a tab dialog box, with each tab displaying a different part of the system. The top of the tab dialog displays the address of the avionics, the number of milliseconds that the avionics has been on, and the date and time from the GPS.

Just under the title bar is a series of lights that are used as alarms. They represent the five most important parameters regarding the health of the vehicle. The alarms are given in order of decreasing priority from left to right. For each parameter the light will be red if the data indicates that parameter is out of range:

- `ALT` is the altitude alarm. The altitude alarm is based upon the altitude limits specified for the avionics (see section 4.9).
- `IAS` is the indicated airspeed alarm. Similar to the altitude alarm the `IAS` alarm is based upon the airspeed limits for the avionics (see section 4.9).
- `SYS` is the system diagnostic alarm. It is based upon the main input voltage, and the board temperature.
- `GPS` is the GPS alarm. It is based upon the health of the GPS receiver as indicated by the number of satellites and `PDOP` (precision dilution of precision) value.
- `COM` is the communications alarm. It is based upon the communications timeout specified for the avionics (see section 4.9).

An audio output can be enabled for the alarm system. If one or more of the parameters are alarming, and if the audio output is enabled, then a warning will be given every two seconds for the alarm with the highest priority.

4.5 Telemetry page

The telemetry page displays data from all the sensors of the system. This display in Figure 4 is shown in advanced mode.

4.5.1 GPS data

The GPS data gives the information received from the GPS onboard the avionics. The GPS data are updated once per second. The data in Figure 4 indicate that the vehicle is covering ground at 25 meters per second over San Francisco. The data displayed here are coming from the piccolo simulator; which why the GPS indicates that no satellites are visible. Notice the PDOP value. This is the precision dilution of precision number, which is based purely on the geometry of the visible satellite constellation. Typical values should be less than 5. There are usually always at least six satellites visible, and sometimes as many as 10. If you are seeing high PDOP values or few satellites that indicates the GPS is not performing well, typically due to problems with the antenna configuration.

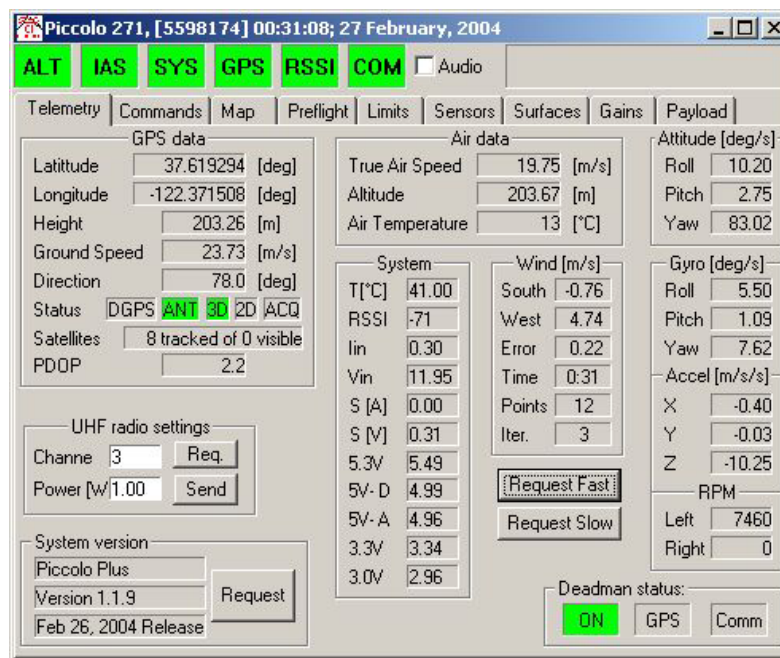


Figure 4. Telemetry page

4.5.2 Air data

The air data gives the airspeed and barometric altitude information. The temperature indicated is based on the internal board temperature with an offset to account for warming in the box.

4.5.3 Attitude

The attitude display gives the estimated Euler attitude angles for the vehicle. The angles are estimated based on integration of the gyro rate data combined with a correction step based upon

the GPS and air data information. The pitch data given is not actually the pitch angle of the aircraft, but is instead the climb gradient (which is the pitch angle minus the angle of attack).

4.5.4 System

The system group contains diagnostic data from the avionics. This data is identical to that from the ground station. $T[^\circ\text{C}]$ is the temperature of the main board in degrees Celsius. The boards have been tested from 0 to 80°C for *Piccolo*, and -40 to 85°C for *PiccoloPlus*. $RSSI$ is the receive signal strength indicator from the MHX radio. It is in units of dBm and ranges from -71 (strong signal) to -115 (no signal). I_{in} refers to the input current, and is displayed in Amps. V_{in} is the input voltage, which should always be between 8 and 20 volts. $S[V]$ is the servo voltage, and $S[A]$ is the servo current. The servo voltage and current are only available on *PiccoloPlus*. The remaining system data are only visible in advanced mode. $5.3V$ is the voltage after the first stage regulator, and should nominally be between 5.3 and 5.5 Volts. $5.0V-D$ and $5.0V-A$ are the digital and analog five volt rails respectively. $3.3V$ is used to run the MPC555 and memory. $3.0V$ is used to run the M12 GPS.

4.5.5 Winds

The winds group displays the latest data from the wind-finding by maneuver calculation. Winds are named for the direction they come from, hence South wind is the component of the wind from the South going to the North. The time of the wind calculation is also provided, as is the number of points used in the calculation and the iterations required to get the solution to converge. The wind data are an integral part of the sensor fusion algorithms on the vehicle as well as the navigation system. In addition they can be useful when planning longer range flights.

4.5.6 Gyro and Accelerometers

The gyro and accelerometer data follow standard US aircraft sign conventions; i.e. the X-axis points through the nose, the Y-axis through the right wing, and the Z-axis down. Hence when flying at a one-g loading the Z accelerometer will indicate -9.8 m/s/s.

4.5.7 RPM

External IO lines 5 and 20 (TPU_B2 and B3) are provided for measuring RPM. The RPM is measured by sampling the time between falling edges on the signals. TPU_B2 is used to measure left engine RPM and B3 is used to measure right engine RPM.

The request fast and request slow buttons alter the rate at which the telemetry data are sent. The system defaults to slow mode in which it sends data once per second. In fast mode the data are sent 20 times a second (or as fast as available bandwidth will allow). Fast mode is provided for diagnostic purposes and is not used for routine flying.

4.5.8 Deadman Status

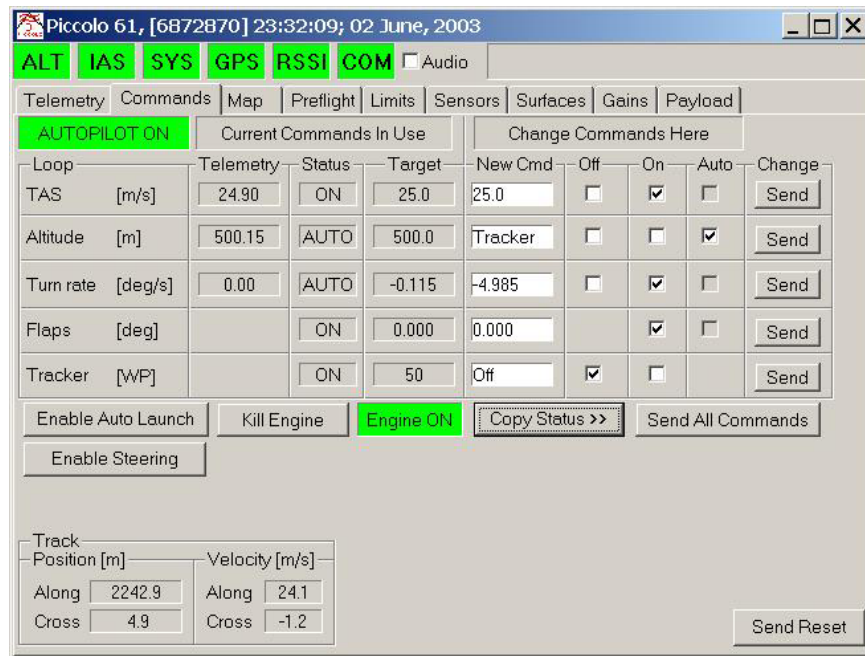
The deadman is an i/o line on the external connector which can be used to control a flight termination system (ignition, fuel cutoff, parachute, etc). The deadman output requires a continual hardware refresh in order to remain on. Hence if the system hardware or software fails the deadman output will drop. It can also be explicitly turned off by GPS failure or by communications failure. In Figure 4 the deadman is on, and the GPS and communications are working normally. If the deadman were off, or the GPS or communications were failed the fields would be red.

4.5.9 UHF radio settings

The UHF radio settings allows the user to control the power output and the hopping pattern of the Microhard radio in the avionics. The radio is frequency hopping spread spectrum, which means that it doesn't use a single frequency, but instead hops along a pattern of frequencies. It is possible to work with multiple ground stations by configuring each ground station to use a different hopping pattern and then changing the hopping pattern of the avionics to match the desired ground station. Pushing the `Req` button will request the current hopping pattern and power output of the radio. Pushing the `Send` button will update the hopping pattern and power output of the radio. When changing hopping patterns the avionics will automatically revert to the previous setting if communications are not established with a ground station on the new pattern. Note that while requesting or changing the radio settings the radio will be taken offline and communications will be suspended for a few seconds. As a result you should not request or change the radio settings when flying under manual control.

4.6 AP commands

The autopilot commands page displays the current autopilot command status, and allows you to change the commands. In the upper left hand corner the green box indicates that the autopilot is on. This is because the global autopilot switch on the pilot console is on; or the console is not plugged into the ground station; or this avionics is not selected for manual control. If manual control were selected from the pilot console the `AUTOPILOT ON` box would be red, and would read `MANUAL`.



Loop	Telemetry	Status	Target
TAS [m/s]	24.90	ON	25.0
Altitude [m]	500.15	AUTO	500.0
Turn rate [deg/s]	0.00	AUTO	-0.115
Flaps [deg]		ON	0.000
Tracker [WP]		ON	50

New Cmd	Off	On	Auto
25.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Tracker	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
-4.985	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
0.000		<input checked="" type="checkbox"/>	<input type="checkbox"/>
Off	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Enable Auto Launch Kill Engine **Engine ON** Copy Status >> Send All Commands
 Enable Steering

Track
 Position [m]
 Along 2242.9 Cross 4.9
 Velocity [m/s]
 Along 24.1 Cross -1.2

Send Reset

Figure 5. Autopilot commands page

4.6.1 Current autopilot status

The left half of the autopilot command page gives the current status. There are five autopilot loops, and hence five rows in the table. The first column of the table gives the telemetry data that is appropriate for that loop. The second column of the table gives the status of

each individual loop (AUTO, ON, or OFF). The third column gives the commanded value for each loop.

The AUTO states means that the loop is enabled and the command for that loop is automatically generated. This applies only to the turn rate and altitude commands when the tracker is on. ON or OFF indicate the loop is enabled or disabled respectively. The actual loop status is a logical OR of the global autopilot status and the loop status. Hence it is possible to have a loop on, but still have the autopilot off.

4.6.2 Changing autopilot commands

The right half of the autopilot command page provides a location for changing the current commands, either individually or all at once. The desired commands are entered in these fields are then are sent by pushing the `Send Autopilot Command` button. For cases where you only want to change one command press the send button next to the loop in question. The `Copy Status` button will copy the information from the status display into the command fields.

4.6.3 Automatic launch

The `Enable Auto Launch` button puts the avionics into its automatic launch mode. This mode is used for catapult launches. When entering this mode with the autopilot on, the autopilot will suspend execution of the main control loops, leaving their outputs at the trim position. The only exception is the altitude control loop which will continue to function. The autopilot will stay in this mode until the airspeed signal reaches 75% of its minimum command value. When this happens the autopilot will pause for 0.5 seconds, and then return to normal operation.

4.6.4 Engine Kill

The `Kill Engine` button gives the user the ability to override the altitude control loop and force the throttle output to zero. The engine kill feature can be driven by the user, through the button on the autopilot page, or driven by the deadman status logic, see section 4.9 for details.

4.6.5 Manual steering

Manual steering provides the ability to use the pilot console to steer the vehicle while still under control of the autopilot. If the vehicle is selected for manual pilot control then the data from pilot console are sent to the vehicle even if it is in autopilot mode. When manual steering is selected the tracker is suspended and the aileron stick on the pilot console becomes the source of the turn rate command.

4.6.6 Track data

Below the autopilot status display the tracker input variables are displayed. They are: the along track distance, which is the distance to go before reaching the waypoint; the cross track distance, which is the perpendicular distance from the track to the vehicle (positive right); and the along and cross track velocities.

4.7 Map

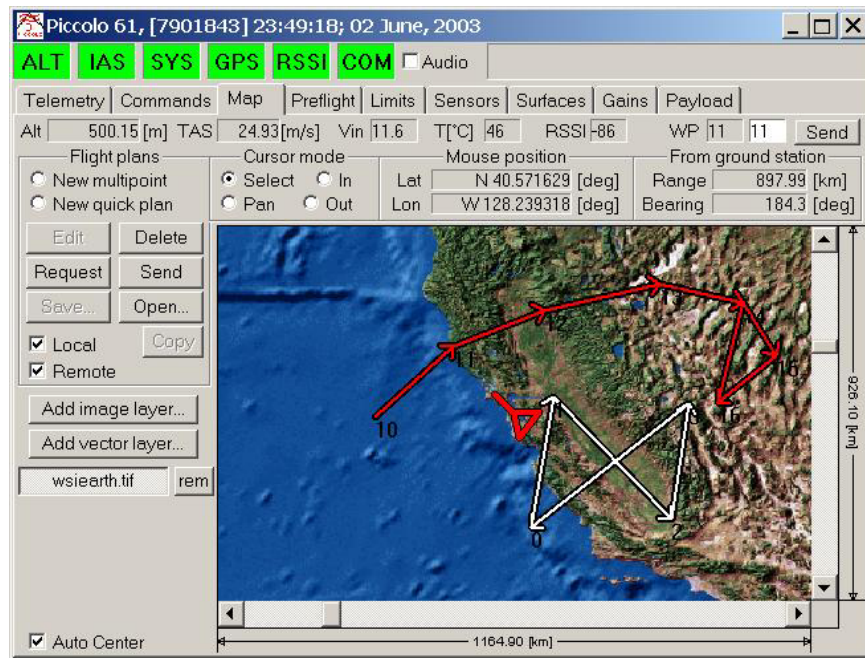


Figure 6. Map page

The map page displays the current location of the vehicle, and provides an interface for creating flight plans. The map is built on ESRI's MapObjects, which means that it is capable of displaying geo-referenced raster files, as well as vector shape files. To add a layer to the map select the button `Add image layer` or `Add vector layer`. Each layer added is displayed below the add buttons. In Figure 6 there is one layer, which is named `wsiearth.tif`. The layer can be turned off by pressing the `wsiearth.tif` button, and removed by pressing the `rem` button.

Whenever the mouse pointer is over the map the location of the pointer is displayed at the top of the map. The map can be zoomed in by selecting the `In` radio button and using the mouse to click and drag a rectangle on the map. The rectangle will become the extents of the map. Zooming out is accomplished by selecting the `Out` radio button and clicking on the map. Each click expands the map area by 50%. To move the map select the `Pan` button, and click and the drag the map with the mouse. Or use the scroll bars to pan the map.

At the top of the map page the most important data from the telemetry stream are displayed: the altitude, true air speed, input voltage, board temperature, and receive signal strength from the radio. In addition the waypoint (shown as WP in the upper right corner) that the vehicle is currently tracking is displayed. The waypoint command can be changed by entering a new waypoint and clicking `Send`.

The map also displays flight plans for the vehicle. There are two types of flight plans displayed: remote plans and local plans. The display of either type of flight plan can be turned on or off using the `Show Local` or `Show Remote` check boxes. Remote plans are those that are stored onboard the avionics, i.e. remote from the operator interface, and they are drawn in red. Local plans are those that are stored on the operator interface, i.e. they have not yet been sent to the avionics, and they are drawn in white.

To see the flight plans that are onboard the avionics press the `Request` button. This will trigger the avionics to send all of its flight plans.

4.7.1 All about flight plans

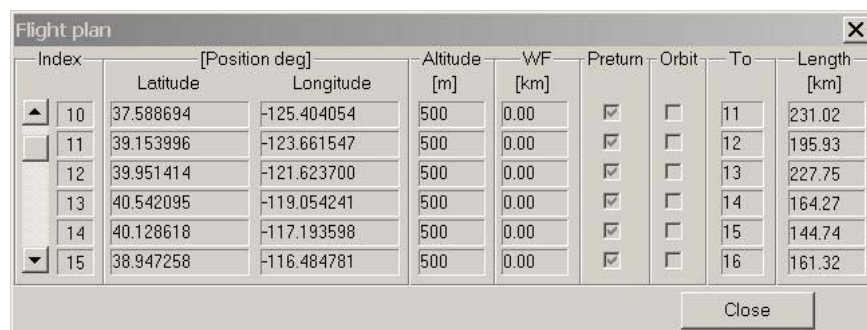
Flight plans are made of linked lists of waypoints. Each waypoint encodes latitude, longitude, altitude, and the index of the next waypoint to be used. The avionics has storage for 100 waypoints. These waypoints can be used to make one flight plan that is 100 waypoints long, or 50 flight plans with two waypoints each, or any combination between. The key concept is that all flight plans must close. This means that at some point the next waypoint must point to a waypoint that was previously in the list. In Figure 6 the flight plan is made of waypoint 10, which goes to 11, which goes to 12, to 13, to 14, to 15, to 16, and back to 14. Note that although all flight plans must close, they do not have to be a loop. In the sequence given above the aircraft will end up flying the triangle between waypoints 14, 15 and 16.

Flight plans also encode altitude. Each time the navigator switches waypoints it changes the altitude command to the value in the waypoint it is going to. To edit a flight plan, including the altitude, push the `Select` radio button, highlight the flight plan and then click `Edit`. The dialog in Figure 7 dialog will appear.

In addition to the basic position the flight plans include a feature called the wind-finding interval (WF in Figure 7). This interval describes the time between performing s-turns along the flight plan. The s-turns allow the avionics to measure the wind. You can disable the wind-finding by setting the interval to zero or less than zero.

Each waypoint also has a feature called pre-turn, which can be enabled or disabled according to the check box in Figure 7. Pre-turn allows the avionics to estimate when to turn so that it doesn't overshoot the segment that follows the waypoint. When you create a flight plan in the operator interface pre-turn will be on by default; however if you need to guarantee the vehicle flies over a waypoint you can disable the pre-turn for that point.

Each waypoint also has a feature called orbit, which can be enabled or disabled according to the check box in Figure 7. Orbit allows the avionics to circle the waypoint rather than flying to it. When orbit is enabled the avionics will not switch waypoints, but instead will fly to the target waypoint and orbit it in a left hand turn. The orbit radius is equal to the tracker convergence parameter, see section 4.12.



Index	[Position deg]		Altitude	WF	Preturn	Orbit	To	Length
	Latitude	Longitude	[m]	[km]				[km]
10	37.588694	-125.404054	500	0.00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	11	231.02
11	39.153996	-123.661547	500	0.00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	12	195.93
12	39.951414	-121.623700	500	0.00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	13	227.75
13	40.542095	-119.054241	500	0.00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	14	164.27
14	40.128618	-117.193598	500	0.00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	15	144.74
15	38.947258	-116.484781	500	0.00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	16	161.32

Close

Figure 7. Flight plan edit dialog

Note that Figure 7 shows the dialog for a remote plan, which cannot be edited. If the plan were local the edit boxes would be white (i.e. editable) and any changes made would be reflected on the map.

4.7.2 *Making new flight plans*

To make a new flight plan select the `New multipoint` radio button. Then using the mouse click on the map once for each waypoint. To end the flight plan double click. The flight plan will automatically close. To control which point it closes to, double click on top of the desired point. When you are finished the OI will prompt you for the flight plan altitude. After entering the altitude the new flight plan is drawn as a local plan on the map. There is only ever one local plan, so the previous local plan will be discarded. The local plan can be edited to include any altitude information or fine tuning of the position data.

Alternatively you can make a simple two waypoint flight plan by clicking `New quick plan`. Using this option a flight plan is built from the current location of the vehicle to the wherever you click on the map.

4.7.3 *Sending flight plans*

To send a local flight plan to the avionics press `Send`. The map will prompt for the starting waypoint index. The index you choose depends on what plans are already onboard the avionics. After sending the flight plan the avionics will echo the plan back to the map, and it will be displayed in red. The local plan, displayed in white, will remain.

4.7.4 *Dragging waypoints*

As of software version 1.1.9 and later you can click and drag waypoints. To do this choose the `Select` radio button. Then click and drag on a waypoint. When you release the mouse button the waypoint location will be updated. If the flight plan was local the change occurs immediately. If the flight plan was remote the change is sent to the avionics and the screen will be redrawn when the avionics echoes the new waypoint position.

4.7.5 *Geo-referencing your own map file*

If you have an image of the area you wish to fly and you want to use this image for your map you will need to geo-reference it. The map will accept geo-referenced TIFF image files; so your first step is to get your image in TIFF format. The geo-referencing is done through a separate file, called a world file, that has the same name as the image file, but with the extension “`tfw`”. The world file contains six numbers, one on each line, and has the following format:

```
degrees longitude per x pixel
0
0
degrees latitude per y pixel (negative)
Longitude of the upper left corner (positive East)
Latitude of the upper left corner (positive North)
```

The geo-referencing assumes that the image is linear, therefore it must be a rectilinear projection, and it must be oriented so that North and South are in the vertical direction. Note that the y axis scale factor should be negative; this is a result of computer graphics coordinates systems which increase downwards. An example world file, which is for Wasco airport in Oregon, follows:

```
0.00005140576
0.000000000
0.000000000
-.00005140576
```

-120.68207806
45.591328

4.8 Preflight

The preflight page gives a simple preflight checklist. Each item of the checklist is represented by a button that can be pushed in to indicate that step has been done. For a detailed description of the preflight checklist items see the document “[Preflight Checklist for the Piccolo System](#)”. In addition the actual checklist item the preflight page include some of the user interface elements required to perform the checklist items, including the altimeter setting, sensor zeroing, and GPS health.

4.8.1 Altimeter setting

The preflight page also has the altimeter request and send button. The altimeter setting is the atmospheric pressure at seal level used to calculate altitude. Push the request button to learn the current setting; and push the send button to alter the setting to whatever is in the edit box

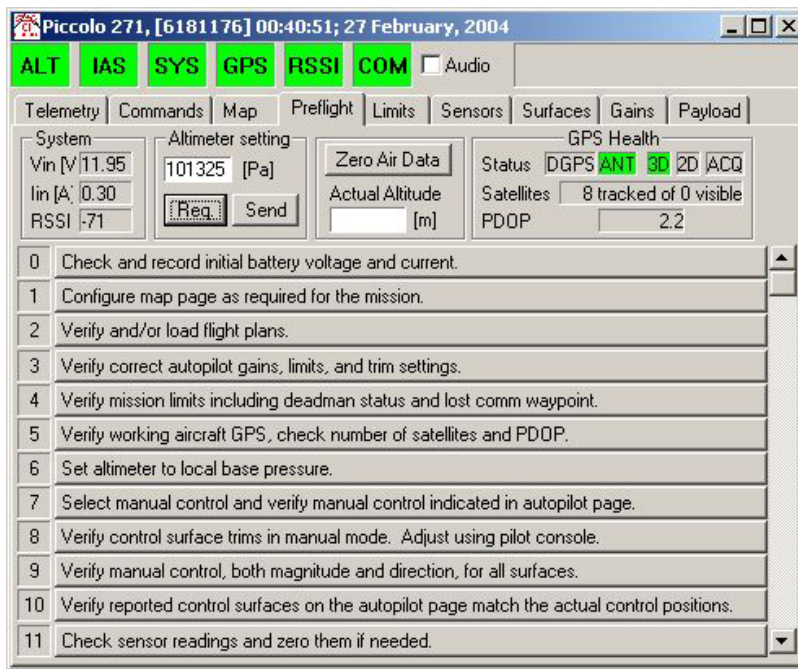


Figure 8. Preflight checklist.

The Zero Air Data button is used to zero the dynamic and barometric pressure sensors.

4.8.2 Zero Air Data

The air data sensors (dynamic and barometric pressure) can be zeroed to take out any residual error before launch. To do this press the Zero Air Data button. The sensor readings at the time the button is pushed will be stored as the sensor offset and applied to subsequent readings. If the Actual Altitude field contains a value when the sensors are zeroed then the barometric pressure zero is also adjusted to match the actual altitude. It is best to zero the barometric pressure sensor after setting the altimeter setting.

4.9 Autopilot limits

The limits page is used to view and alter the autopilot and mission limits.

4.9.1 Command limits

The airspeed, altitude, and turn rate commands are limited by max and min dynamic pressure, max and min altitude, and max and min bank angle. The Piccolo autopilot is mostly de-coupled between longitudinal and lateral control, hence extending the bank angle limits beyond 30° should only be done with caution.

4.9.2 Output limits

In addition to the command limits this page also shows the output limits. The five outputs from the autopilot will not exceed the values given here. To see the current limits press the `Request Limits` button. To change the limits press the `Send Limits` button.

4.9.3 Mission limits

The limits page also shows the mission limits. The pilot timeout gives the amount of time the system will remain in manual control mode without new manual control inputs. If this timeout elapses the system will automatically switch to autopilot mode. If communications fail entirely for the time defined in `Comm timeout` the avionics will take action by flying the flight plan indicated by the `Lost comm waypoint`. It is important to note that the lost communications waypoint identifies a whole flight plan, not just a single waypoint. When the communications timeout elapses the avionics examines the lost comm flight plan and makes an intelligent choice of the waypoint to use. The choice is based on finding the waypoint which is closest to the avionics AND which is closer to the termination point of the flight plan than the avionics is.

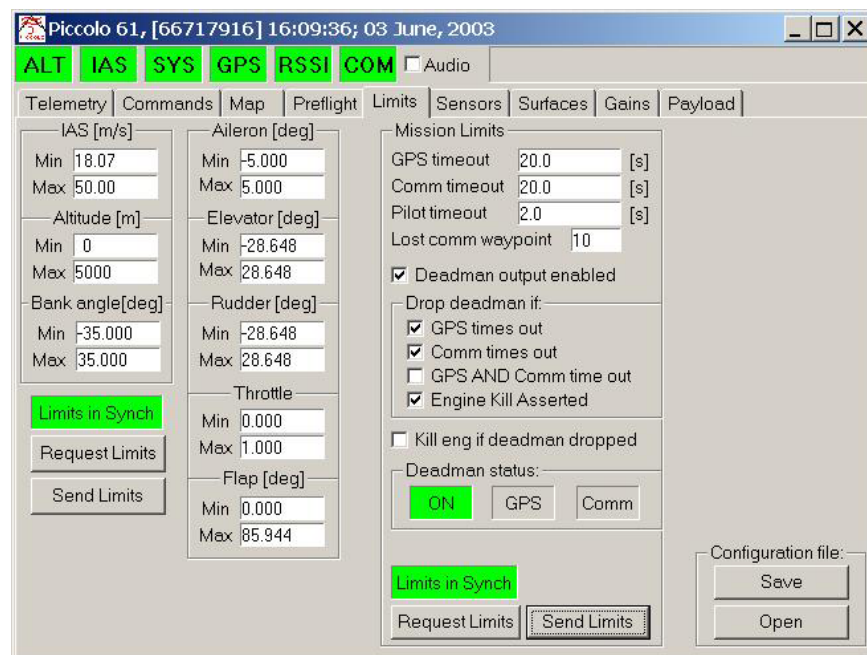


Figure 9. Autopilot limits page

The avionics also supports a “deadman’s” output. This is an output switch which requires a continuous hardware refresh in order to stay active. Hence if the system fails for any reason the switch will turn off. For safety the deadman can be connected to the engine ignition, or some

flight termination system. The deadman output can also be controlled as a function of the mission limits. The deadman has a global enable which is controlled by the user, and flags to drop the deadman output in the event of:

- Lost communications (controlled by timeout)
- Lost GPS (controlled by timeout)
- Lost GPS AND lost communications.
- Engine kill is asserted by the user (see section 4.6).

The deadman logic can also be used to assert the engine kill state if the deadman is dropped. Hence for vehicles which do not have a convenient flight termination system (like an ignition) the engine can be driven to zero throttle if the communications or GPS fails.

4.10 Sensors

The sensors page gives the current sensor readings and calibration information for each sensor. It is normally only used during the setup process. On the left of the page each sensor is listed as a radio button. Select the radio button corresponding to the sensor calibration you wish to view. Push the Request All button to trigger the avionics to download all of the sensor information.

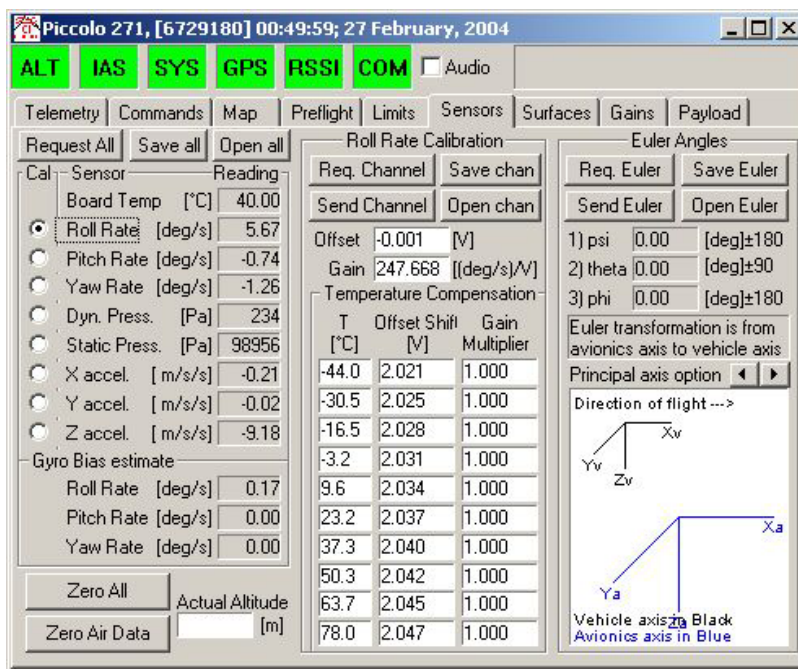


Figure 10. Sensor calibrations

4.10.1 Sensor calibrations

The sensors are calibrated using a linear relationship which is based on an offset and gain number. The calibrated output is the raw signal minus the offset, times the gain. The offset and gain number is displayed top center in the sensor calibration page.

In order to extend the temperature range over which the avionics can function the sensors are temperature compensated. This compensation takes the form of an offset shift and a gain multiplier that depends on the temperature of the board. The table of offset shift and gain

multiplier is displayed below the main offset and gain numbers. Each *Piccolo* unit is temperature compensated at the factory.

4.10.2 Zeroing sensors

The sensors can be zeroed to take out any residual error before launch. To do this press the `Zero Sensors` button. The sensor readings at the time the button is pushed will be stored as the sensor offset and applied to subsequent readings. If the `Actual Altitude` field contains a value when the sensors are zeroed then the altitude zero is also adjusted to match the actual altitude.

During the zeroing process the system assumes that it is level and not accelerating. Hence the dynamic pressure sensor, gyros, and x and y accelerometers should be reading zero; while the z accelerometer reads -9.8 m/s^2 . Since the system automatically estimates gyro biases, most of the time you only want to zero the dynamic and barometric pressure sensors. In that case the `Zero Air Data` button can be used.

4.10.3 Gyro bias estimate

Even with temperature compensation the gyros have more drift than is acceptable. Hence the avionics continuously estimates the bias error in the gyros using data from the GPS. The bias estimates are displayed in the gyro bias estimate box.

4.10.4 Gyros in *Piccolo*

The gyros are the most crucial sensors for basic stability of the vehicle. They are also the most sensitive to disturbances. In particular the most common problem with the gyros results from rapid temperature changes. These errors are most visible in the gyro bias numbers. The gyros have about $\pm 100^\circ/\text{s}$ at 10 bits of dynamic range. Hence if the bias value approaches these limits the gyros will stop functioning as the sampling hardware becomes saturated. The gyro calibration scheme used in the *Piccolo* provides for a hardware shift of the gyros. Hence the actual offset number displayed on the calibration pages represents an offset in hardware rather than software. The *Piccolo* software will automatically shift the gyro hardware bias if it detects that the gyro errors are becoming too large for correct operation of the system.

4.10.5 Gyros in *PiccoloPlus*

The gyros used in *PiccoloPlus* are much more stable and robust than those used in *Piccolo*; as well as having more dynamic range ($\pm 100^\circ/\text{s}$ at 16 bits). Hence the hardware shifting done in *Piccolo* has been eliminated in *PiccoloPlus*.

4.10.6 Installation orientation

Since the avionics has 3-axis gyros and accelerometers it can be installed in any orientation and the readings rotated to match the installation angles. The installation orientation is given by the three Euler angles Yaw (psi), Pitch (theta), and Roll (phi). Currently the operator interface only supports choosing Euler angles that are along principal axes of the vehicle. There are 24 different possible principal axes orientations, which can be selected with the `Principal axis options` spin buttons. The Euler angles are used to describe the rotation from the avionics coordinate frame to the vehicle coordinate frame. The avionics coordinate frame is marked on the front panel of the box. To change the angles press the `Send Euler Angles` button. Finally, note that the calibration data are given in avionics axis, but the readings are given in vehicle axis

4.11 Surfaces

The surfaces page is used to control how the five autopilot outputs are mixed and matched to generate up to ten servo outputs; and also to provide the calibration information to go from surface angle to servo pulse width.

Configuration

☒ Enable surface setup Request all data Send all data

Configuration

Send Config.

☒ 10 channel
☐ 5 channel

Mixing

Aileron Differential: 0

Flap to Elevator [%]: 0

Aileron to Rudder [%]: 10

Flaperon [%]: 0

☐ Elevon

Ruddervator

☐ On
☐ Upright
☒ Inverted

Surface File

Save Open

Surface selection

☒ Aileron 1
☐ Elevator 1
☐ Throttle 1
☐ Rudder 1
☐ Flap 1
☐ Aileron 2
☐ Elevator 2
☐ Throttle 2
☐ Rudder 2
☐ Flap 2

AILERON 1 calibration

Order Table

Send Table

	Angle [deg]	Pulse [us]
0	-25.73	1103
1	-20.00	1194
2	-14.32	1286
3	-8.59	1378
4	-2.86	1470
5	2.86	1561
6	8.59	1653
7	14.32	1745
8	20.00	1837
9	25.73	1929

Arm Surface Test

Test Angle: [] [deg]

Test Pulse: [] [us]

Figure 11. Surface calibration page

4.11.1 Actuator calibrations

In a real airplane the surface position, direction of travel, and amount of travel depend on the actuator to surface linkage design. In order to account for variations in the linkage the autopilot and/or mixer outputs are translated to actuator signals via calibration data. The calibration data account for sign convention, surface neutral, travel limits, and any non-linearities in the surface motion.

Each channel has it's own calibration data, an example of which can be seen Figure 11. The data relate the actuator pulse width in micro-seconds to the surface motion in degrees. In the case of the throttle the surface position is interpreted as percent of full scale, and is always between 0.0 and 1.0. The surface position data are usually given in ascending order, and the pulse widths are in ascending or descending order. The sign convention on the surface deflection has the following rules: flaps, ailerons, and elevators are all positive down. Rudders are positive right. For ruddervators the sign convention follows the elevator rule, i.e. positive down.

The autopilot outputs 5 signals that are then adjusted using the output mode (5 or 10 channel) and mixing rules. The resulting outputs are then converted to pulse widths via linear interpolated lookups through each channel's calibration data. If the requested deflection is outside of the table, then the pulse width is limited to the extremes in the table. Although the pulse width column in Figure 11 is regularly spaced, this is not required. Note the extreme pulse width values of 1103 and 1929. These correspond to $\pm 100\%$ pulse width as measured from a Futaba pilot console.

The surface calibrations and mixing rules cannot be altered unless the `Enable surface setup` check box is selected. The `Request all data` button will trigger the avionics to send all of its surface setup information. On the left of the page the mixing options are given. To change the mixing options alter the settings accordingly and then press the `Send Config` button. The mixing rules shown in Figure 11 are for a conventional aircraft that does not have ailerons, and uses rudder as its primary roll rate control (note the 100% mixing from aileron to rudder)

The surfaces page only shows the pulse width to angle calibration for one surface at a time. The surface displayed is chosen with the `Surface selection` radio buttons. To change the surface calibration table enter in the new table data; then press `Order Table`, which will put the table in order of increasing angles, then press `Send Table`.

In order to determine the calibration numbers it is helpful to be able to explicitly set the pulse width being sent to any given channel. This can be done with the surface test feature in the bottom center of the page. To use the feature the autopilot must be on. Then pressure the `Arm Surface Test` button. The surface test feature will remain armed for 30 seconds. Then enter in a desired pulse width and press the `Test Pulse` button. The requested pulse width will be sent out the selected channel until the arming of the surface test times out. While the pulse is being sent you can measure the actual surface deflection. By doing this for each desired pulse width the surface calibration table can be built up.

4.11.2 5 channel mode

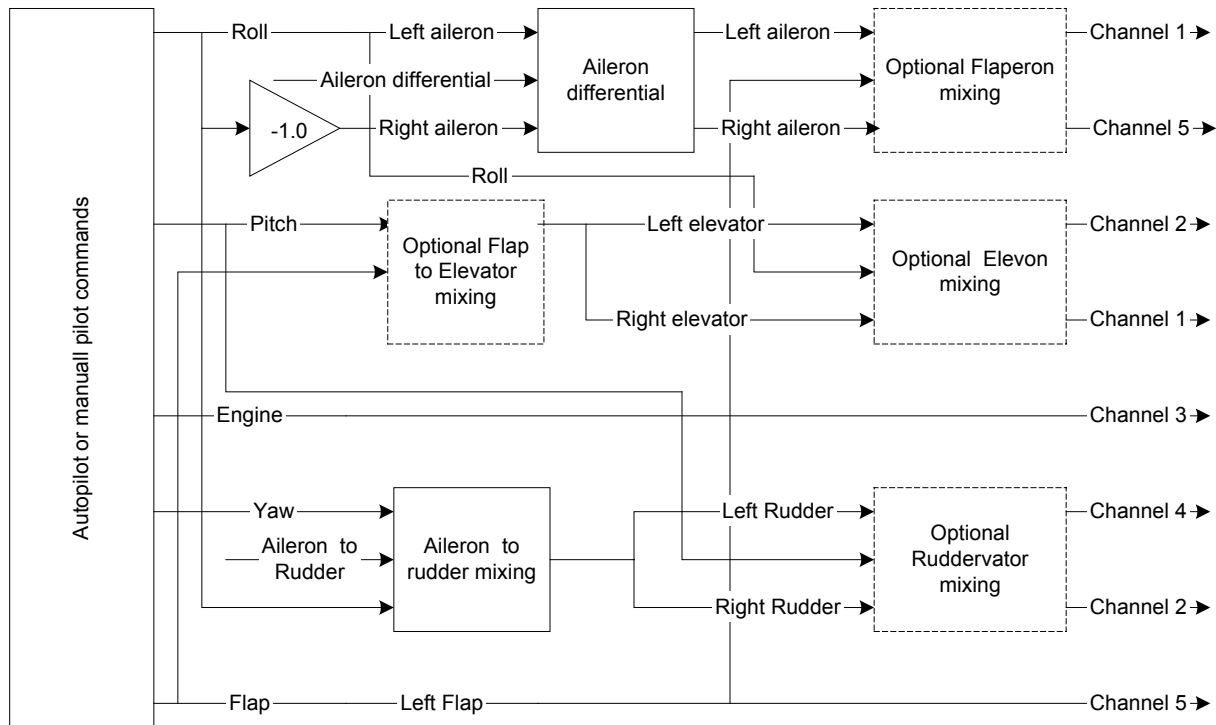


Figure 12. 5 channel servo output diagram

In this mode only 5 output channels are used. The channel mapping depends on the type of mixing option that is to be supported, see Figure 12. Note that in 5 channel mode flaperon and elevon mixing cannot be used simultaneously, elevon and ruddervator mixing cannot be used simultaneously, and aileron differential is only available in flaperon mode.

4.11.3 10 channel mode

In this mode all 10 output channels are used. The channel mapping depends on the type of mixing option that is to be supported, see Figure 13.

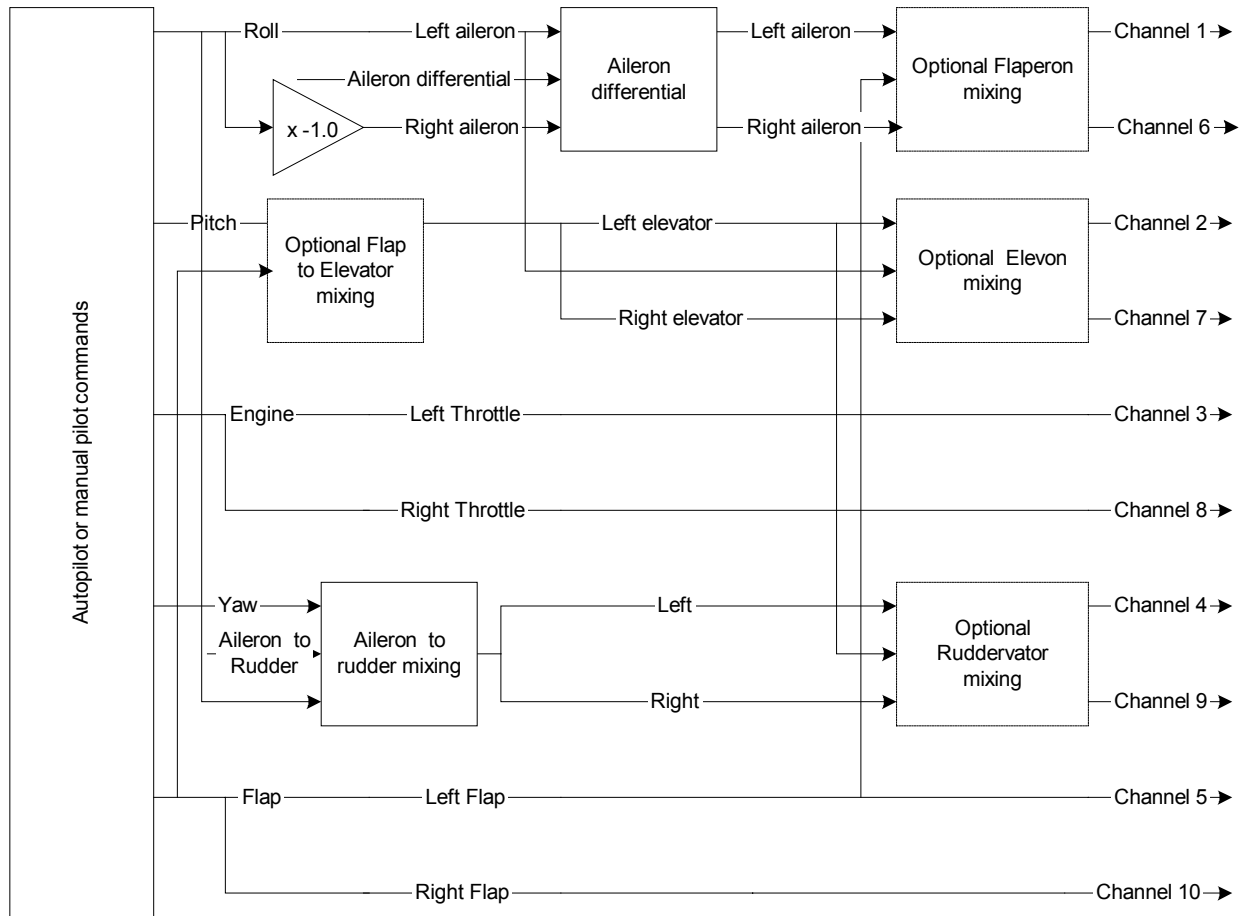


Figure 13. 10 channel servo output diagram

4.11.4 Aileron differential

Aileron differential is used to control the ratio of up aileron motion to down aileron motion. By adjusting the aileron differential the designer can guard against tip stall induced by downward aileron motion, or remove asymmetries caused by the up aileron moving into more sluggish air than the down aileron.

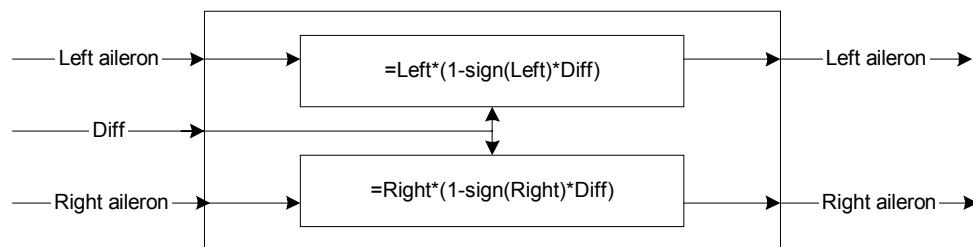


Figure 14. Aileron differential

The differential parameter in Figure 14 controls how the differential is applied. When differential is zero the up and down aileron motions are equal; when it is 1.0 the ailerons only move upwards; and when it is -1.0 the ailerons only move downwards.

4.11.5 Aileron to rudder mixing

In this mode the aileron output from the autopilot is summed with the rudder output to form a new rudder output. The aileron output is unchanged. This feature is useful for aircraft that have strong adverse yaw characteristics that must be countered; or for aircraft which have their primary roll control via the rudder.

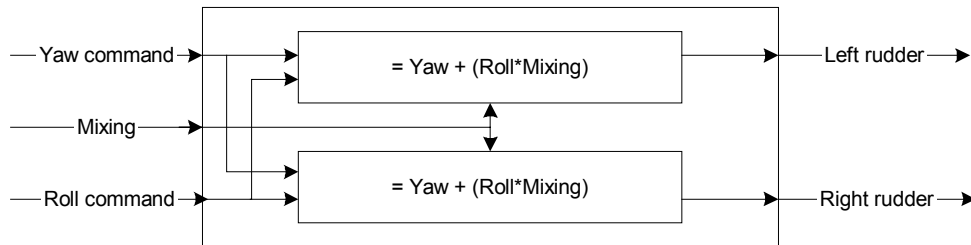


Figure 15. Aileron to rudder mixing

4.11.6 Flap to elevator mixing

The flap to elevator mixing is used to compensate for the change in elevator trim as the flaps are moved. Most vehicles will use a positive amount of flap to elevator mixing(i.e. positive flap deflection causes positive elevator deflection).

4.11.7 Flaperon mixing

In this mode the ailerons are used to provide aileron and flap functions. This is done by biasing the aileron outputs with the flap output, according to Figure 16. Note that the flaperon value can be positive or negative. A flaperon of 100% will cause the ailerons to move down as far as the flaps move down. A flaperon of -100% will cause the ailerons to move up as far as the flaps move down.

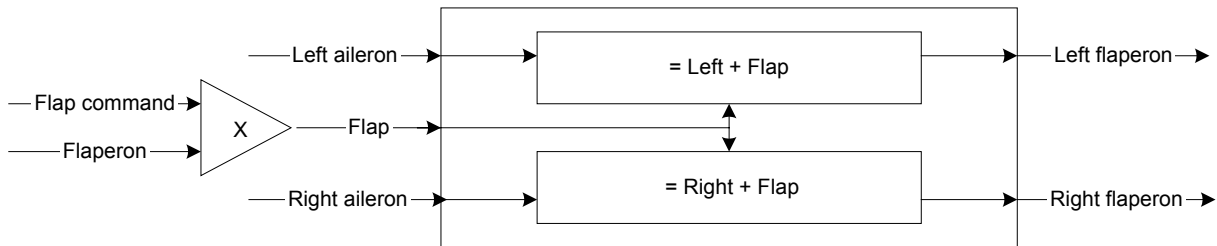


Figure 16. Flaperon mixing

4.11.8 Elevon mixing

In this mode two elevator like surfaces are used to provide aileron and elevator functions. This is done by biasing the elevator outputs with the aileron output according to Figure 17.

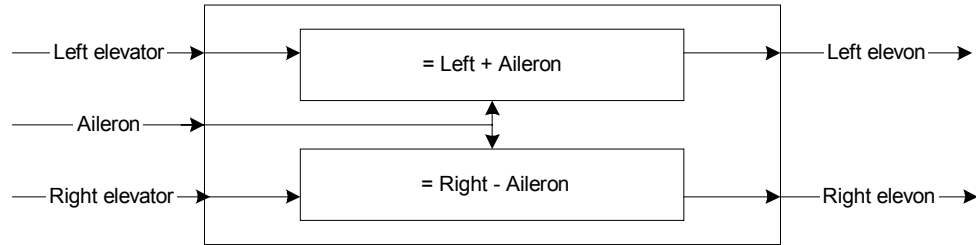


Figure 17. Elevon mixing

4.11.9 Ruddervator mixing

This mode is used for V-tail operation, in which the rudder and elevator outputs are mixed together. Ruddervator mixing depends on the type of V-tail, either upright (V) or inverted (Λ). When defining rudder motion for a V-tail, the sign convention follows the same one used in the elevators.

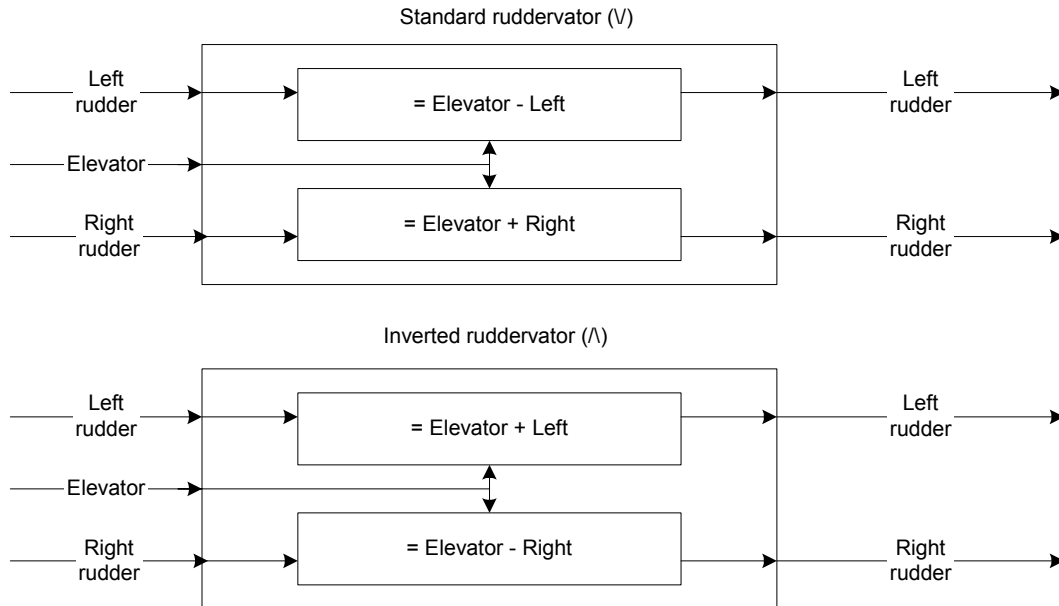


Figure 18. Ruddervator mixing

4.12 Autopilot gains

The gains page is used to view and alter the autopilot gains. There are gains for seven loops.

Table 3. Autopilot loops

Loop	Inputs	Outputs	Notes
Dynamic Pressure	Dynamic Pressure	Elevator	Maintains a commanded dynamic pressure.
Altitude	Static pressure	Throttle	Maintains a commanded altitude.
Turn	Yaw rate	Aileron, Rudder	Maintains a commanded turn rate.
Tracker	GPS	Turn rate Command	Drives the turn rate loop to achieve desired track targets.
Roll	Roll angle	Aileron	Alternative turn rate control and roll

			angle disturbance rejection.
Pitch	Pitch angle	Elevator	Damps out pitch oscillations.
Yaw	Yaw rate	Rudder	Damps out yaw oscillations.
Turn compensator	Commanded turn rate, Pitch rate	Elevator	Helps maintain the TAS command when entering or exiting a turn.

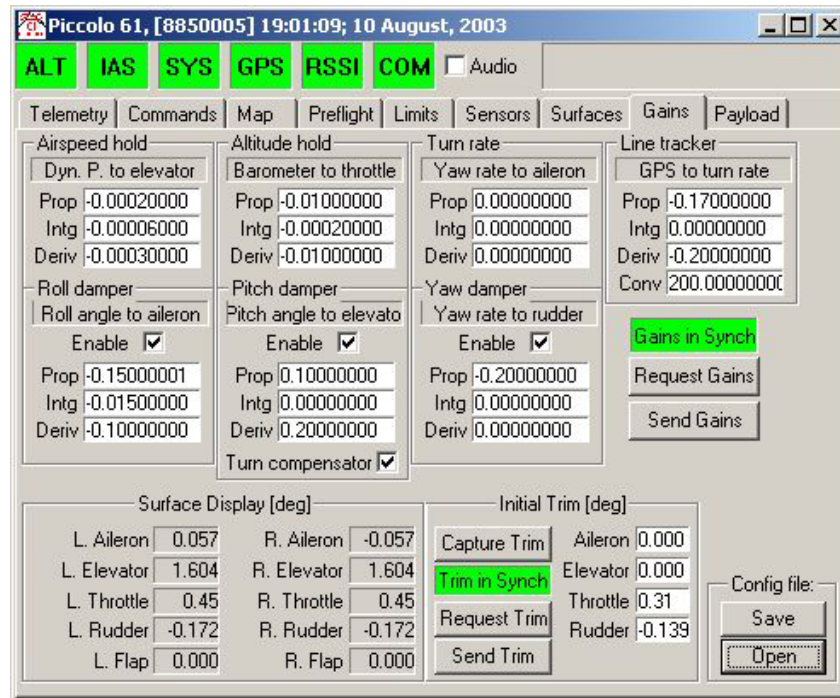


Figure 19. Autopilot gains page

To see the current gains press the `Request Gains` button. When the avionics responds the `Gains in Synch` field will turn green. To change the gains alter one of the fields and push the `Send Gains` button. Adjustments to the gains should be done first on the simulator before fine-tuning in flight

The gains page also gives a display of the current control surface positions, both the left and right. The labels on the control surface display will update according to the type of mixing selected in the surfaces page (i.e. elevons, ruddervator, etc.). To the right of the surface display is the initial trim value. Whenever a loop is turned from off to on, its integrator gets initialized. The turn rate, airspeed, altitude, and yaw damper loops get initialized to the value displayed in the Initial trim display. The other loop integrators get initialized to 0. To request the current trim setting push `Request Trim`. You can enter trim values and push the `Send Trim` button to change them. You can also tell the avionics to use its current loop commands as the trim value by pushing the `Capture Trim` button.

4.13 Payload page

The payload page gives the user the ability to control the I/O lines on the external connector. Ordinarily the manipulation of these lines is done through third-party applications which can provide the user with an application specific interface. However the operator interface provides

a generic payload control display which can be used for testing purposes. There are twelve I/O lines and two serial ports which can be manipulated. Each of the I/O lines has a primary function and several alternate functions. See Table 4 for a description.

Table 4. Signals available on the main Piccolo connector for payload expansion

Pin	Name	Type	Primary Function	Alternate functions
44	PWMSM[0]	O	Left aileron pulse	Discrete output; pulse/period output
43	PWMSM[1]	O	Left elevator pulse	Discrete output; pulse/period output
42	PWMSM[2]	O	Left throttle pulse	Discrete output; pulse/period output
41	PWMSM[3]	O	Left rudder pulse	Discrete output; pulse/period output
40	PWMSM[4]	O	Left flap pulse	Discrete output; pulse/period output
39	TPU_A[0]	I/O	Right aileron pulse	Discrete I/O; pulse/period I/O
38	TPU_A[1]	I/O	Right elevator pulse	Discrete I/O; pulse/period I/O
37	TPU_A[2]	I/O	Right throttle pulse	Discrete I/O; pulse/period I/O
36	TPU_A[3]	I/O	Right rudder pulse	Discrete I/O; pulse/period I/O
35	TPU_A[4]	I/O	Right flap pulse	Discrete I/O; pulse/period I/O
5	TPU_B[2]	I/O	Left RPM	Discrete I/O; pulse/period I/O
20	TPU_B[3]	I/O	Right RPM	Discrete I/O; pulse/period I/O
31-32	Program Port	Serial	External comm. at 57,600	External comm. at user baud rate; Program at 56K
33-34	User Port	Serial	Payload data at 57,600	Payload data at user baud rates less than 56K

Table 5. Modes of operation for the payload signal lines.

Modes	#	Description
Default	0	Signal operates as its primary function
Discrete Out	1	Operates as a discrete output.
Discrete In	2	Operates as a discrete input, only valid for the TPUs.
Pulse/Period Out	3	Generates a repeating square wave with user specifiable period and high time.
Inverted Pulse/Period Out	4	Generates a repeating square wave with user specifiable period and low time.
Pulse In	5	Measures the time between the rising and falling edge the signal, only valid for the TPUs.
Inverted Pulse In	6	Measures the time between the falling and rising edge of the signal, only valid for the TPUs.
Period In	7	Measures the time between rising edges, only valid for the TPUs.

Each signal line has two 32-bit parameters that are used to describe its functionality. The meaning of each parameter depends on the mode of operation.

Table 6. Parameters for different operational modes of the payload lines.

Modes	Meaning of first parameter	Meaning of second parameter
Default	N/A	N/A
Discrete Out	Zero for low output, nonzero for high output.	N/A
Discrete In	N/A	N/A
Pulse/Period Out	Period of the output signal in nanoseconds.	Pulse width of the output signal in nanoseconds.
Inverted Pulse/Period Out	Period of the output signal in nanoseconds.	Pulse width of the output signal in nanoseconds.
Pulse In	Number of input periods over which to accumulate the signal.	N/A
Inverted Pulse In	Number of input periods over which to accumulate the signal.	N/A
Period In	Number of input periods over which to accumulate the signal.	N/A

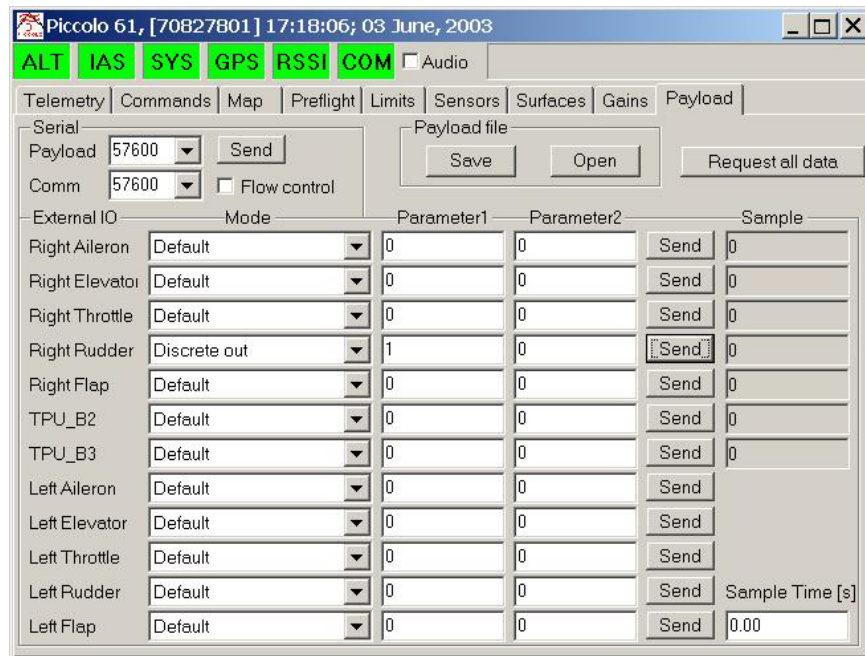


Figure 20. Payload control page

The payload control page also gives the ability to set the baud rate for the two external serial ports. One of the ports, called the payload port, is used to send a bi-directional stream of data between the operator interface and the avionics. The other port supports the same communications protocol that runs over the main UHF link, and allows external communications systems to be integrated. The external communications port can run with or without the flow control scheme implemented for the UHF data link. See the document “Piccolo Communications” for details.

5 Development environment

The development environment is the set of tools that allows the vehicle designer to configure the avionics and software to accommodate the specific application. The tools that make up the development environment include:

- Avionics hardware, with built-in debugging and communication resources.
- Software compiler/debugger and driver source code.
- Control systems design and analysis tools.
- Auto-code generation tools for control systems and state machines.
- Aircraft simulation model.
- Hardware in loop simulation tools.

While there are many small avionics systems available, few provide all the tools listed above. *Piccolo* is novel in part because the development environment was designed in from the beginning.

5.1 Software environment

Piccolo includes built-in software that will be suitable for many applications, however for those applications that require fine control over the software, the software source code for the entire *Piccolo* system can be licensed.

5.2 Design environment

In recent years new design tools for control systems have come to maturity. The popular *Matlab* environment with its controls simulation package *Simulink*, has been augmented by the addition of autocode tools *Real-Time-Workshop (RTW)* and *Stateflow*. Using these tools control system designers can graphically model physical plants and design control laws and state machines. The resulting designs can then be automatically coded in C using *RTW*, compiled using the software environment, and downloaded to the target system. *Piccolo* has more than enough CPU muscle to run complicated autcoded algorithms. Indeed the basic control software that comes with *Piccolo* was designed in exactly this fashion, and Cloud Cap has made the design available to developers.

5.3 Hardware in loop simulation

Hardware-in-loop (HIL) simulation is a cornerstone of unmanned aircraft development. Well designed simulators allow the aircraft control laws and mission functionality to be tested without risking hardware in flight test. Although HIL simulation can not replace flight testing, it measurably reduces the likelihood of failure by detecting bugs and deficiencies in the lab.

To facilitate this vital (and typically difficult) function the *Piccolo* developers kit includes a hardware in the loop simulator that can be used to test the performance of a *Piccolo* implementation. The HIL simulator is based upon the external CAN interface. When in simulation mode *Piccolo* sends control surface signals and receives sensor inputs from the CAN interface. A PC with a CAN interface card running the HIL simulation software completes the necessary hardware. See the documents “[Hardware in the Loop simulator for the Piccolo Avionics](#)” and the “[Piccolo Quick Setup Guide](#)” for more details on how to use the simulator.

