

SENSIMETRICS



Spönder
USER MANUAL

The logo for Spönder features the word "Spönder" in a dark blue, rounded sans-serif font. The letter "ö" is replaced by a light blue Wi-Fi symbol consisting of three curved lines of increasing size. Below the main text, the words "USER MANUAL" are written in a smaller, dark blue, all-caps sans-serif font.



Sponder is an application that turns an Android tablet into a customizable response terminal. The tablet is customized via instructions sent from a master computer that specify the configuration and operation of a displayed array of text buttons. The tablet sends back to the computer the user's button selections. Communication is wireless (Bluetooth) and instructions and replies are simple text strings. Instructions can be sent from the master computer using any application that can read and write to a serial port, such as Matlab or a simple terminal emulator.

Sponder communicates with the master computer using the Serial Port Profile (SPP) provided by Bluetooth. When pairing the computer and tablet through Bluetooth, a virtual serial port on the computer is assigned to handle communication with the device. Multiple tablets can be paired with a computer, with each device assigned a separate serial port. A Bluetooth controller on the master computer can maintain *simultaneous* connections with up to *seven* devices through separate virtual serial ports.

The tablet's screen is divided into three areas: (1) the text area at the top of the screen, which can be used to display arbitrary text such as instructions or questions; (2) the configurable button area below the text area; and (3) the OK button to the right of the text area to finalize a selection of buttons.

System Requirements

Tablet

Android 4.1 or higher.

Bluetooth with support for the Serial Port Profile (SPP). NOTE: Not all tablet manufacturers and models support this profile.

Recommended tablet: Nexus 7

Master computer

PC: Windows 7 or Windows 8.

Apple: Mac OS 10.8.5. Earlier OS versions may work but have not been tested.

Bluetooth with support for the Serial Port Profile (SPP). Note that many computers that have built-in Bluetooth do not support SPP. A USB Bluetooth adapter (dongle) can be used to add that capability. The AZiO BT-D-V201 is one such dongle that supports the SPP; it has been tested successfully with seven simultaneous connections.

In addition, the built-in driver on your computer may not support the SPP. The Toshiba Bluetooth stack is a driver that does provide that support. The Toshiba drive is supplied with the AZiO dongle but it is also available as a free download from the web.

Recommended USB adapter: AZiO BT-D-V201

Recommended driver: Toshiba Bluetooth Stack

Downloading & Installing Sponder on the Tablet

On the tablet, go to Settings and enable Wi-Fi, entering the password for your local Wi-Fi network if necessary.

Then start the Play Store app on the tablet, which will take you to the Google Play Store on the web (<https://play.google.com/store>). Search in the store for Sponder. After purchasing, the app will download. If installation does not begin automatically, drag down the top left corner of the screen (showing a download symbol) and select the downloaded file. Select 'Yes' to install the application.

After installation, the Sponder application will be found on the Apps screen (accessed through the  button on the tablet).

Pairing the Tablet with the Master Computer

In order to connect via Bluetooth, two devices must first be paired in order to establish a trusted association and to agree on use of the Serial Port Profile. Pairing is only required once to enable subsequent connections.

When pairing the tablet with the master computer, the Sponder application must be running in the foreground to advertise the availability of a serial connection. Otherwise, the devices may pair but it would not be possible to communicate with Sponder on the tablet.

Pairing must be initiated on the master computer by searching for available Bluetooth devices and selecting the desired tablet running Sponder.

See Appendix A for detailed instructions on the pairing procedure.

Connecting

Sponder must be running visibly on the tablet to accept an incoming connection. The application stops scanning for incoming connection requests when it is not visible in order to conserve battery power.

Opening the virtual serial port assigned to the tablet *on the master computer* opens a connection between the devices. This can be done using any application that can open serial ports such as Matlab or a terminal emulator.

Writing ASCII characters to the master computer's serial port transmits them to the Sponder application on the tablet.

Closing the serial port closes the connection.

The connection needs to be re-opened after it is closed or is lost some other way, such as by separating the devices until they are out of range or by turning off Bluetooth on either device.

Once an active Bluetooth connection is established, the Sponder application retains the connection and continues to receive and process commands even when it is not visible. However, when the application is not visible, the operating system may terminate it to free memory at any time, which will close the Bluetooth connection.

The process of making the connection is described in detail in Appendix A.

Communication Protocol

Communication between the computer and tablet is done with messages:

The operation of the tablet is specified via *commands* sent by the computer to the tablet.

The tablet replies with a message to acknowledge whether it has understood and executed a command or not.

The tablet sends a message when the user completes a response selection.

Commands

Commands from the master computer must be sent as an ASCII text string beginning with a command code (described below), followed by a list of parameters and/or text to be displayed, and ending with the form-feed character \f.

Sponder concatenates messages sent from the master computer but does not interpret them as instructions until the form-feed character is encountered. Instructions that do not end with \f do not elicit an acknowledgement.

Mode Commands

Selection mode – indicated by instruction code ‘S’ -- determines whether a) one selection per column of buttons is required (corresponding to independent sets of radio buttons), or b) one selected button in the entire array is sufficient, or c) any number of button selections is allowed.

OK Button mode -- instruction code ‘K’ -- determines whether the OK button is used to confirm a selection and transmit the results, or whether results are transmitted immediately upon completion of a selection.

Display mode – instruction code ‘D’ – determines whether all buttons become invisible or remain visible but disabled after a selection is completed.

Action and Parameter Commands

The ‘B’ command is used to create and configure individual buttons in an array. An optional flag used with the ‘B’ command designates a given button as being the correct response, and visual feedback is provided once the user finalizes a selection.

The ‘b’ command operates on all buttons as a group, enabling/disabling them, making them visible or not, and erasing or resetting them.

The ‘F’ command specifies the duration of correct-answer feedback or disables feedback.

The ‘O’ command sets the orientation of the display.

The ‘T’ command is used to display text in the header area.

The ‘t’ command specifies the font used in the header area.

See the Commands section below for detailed descriptions.

Acknowledgement of Commands

The tablet acknowledges every command ending with the form feed character \f that it receives with a message sent back to the computer. Acknowledgements begin with 1 if the command was understood and executed, or 0 (zero) otherwise, followed by an error message.

All Acknowledgements have this form:

'1' if o.k.
'0 *errorMessage*' otherwise

Acknowledgements end with the newline character \n. This facilitates reading response messages with applications such as Matlab, which can use the fgetl() function to scan to the end of the line.

The master program should await and read the acknowledgement from the tablet and then clear its input buffer. This ensures the tablet is ready to receive further instructions.

Example: If the instruction 'T:Good Morning' is executed successfully, the acknowledgement message will be: '1'.

Example: If the instruction 'B 3 8:Yes' is sent to the tablet, the acknowledgement message will be:
0 Error: button column index must be in the range 1 to 7.

User Response

After a user completes response selection, a *user response* is sent from the tablet to the computer.

The format of the user response is as follows:

'R *N*: *list*'

The first character in a user response is always the letter R.

N is the number of responses.

list is a tab-delimited list of the *N* text strings from the selected response buttons.

Example: 'R 2: text2 text5' -- the user made two responses, on the buttons labeled 'text2' and 'text5'.

The fact that responses are reported in terms of button labels means, of course, that buttons must have unique labels.

Responses in the list are ordered according to the location of buttons in the array, counting down the first column, then the second, etc. Thus, the order of responses in the list does not reflect the order in which the responses were made by the user.

Examples of all commands and responses are given in Appendix B in the Matlab language.

It is advisable to begin a session by clearing the input buffer, then sending commands to erase all buttons from a previous session and specifying operating modes before creating a new button array and collecting responses. In many scenarios the operating modes and button states will remain fixed in most respects throughout a session (a sequence of items posed to a user for responses). But they can be changed quickly enough that it is possible to completely re-configure the display and operation of the tablet from one item to the next posed to the user.

Commands

Messages between computer and tablet are sent as strings of characters that adhere to precise formatting rules. In the following, square brackets indicate that a given set of parameters can be left out. However, the order of parameters for a command must follow the specifications indicated here or results may be unpredictable.

B *row column [nc] [R G B] [fontSize]:[text]*

Create a button displaying a text phrase. A set of several such commands specifies the configuration of the button array. Buttons are made visible with the b:v command (see below).

Allowed values are:

<i>row</i>	1-10	The array is limited to ten rows.
<i>columns</i>	1-7	The array is limited to seven columns.
		Buttons resize themselves to fill the available space. Thus, unspecified buttons do not leave an empty space: if the first and third button of the first column are specified, they will size themselves to each take up half of the vertical extent of the column. If there are no other columns, these two buttons will stretch to fill up the full horizontal extent of the screen.
		Make a button non-active with no text to create empty space.
<i>n</i>		Make this button non-active. A non-active button is visible but cannot be pressed, and it is not affected by b:e commands.
<i>c</i>		Designate this button as a correct response.
		If no button has the 'c' parameter, no feedback is given after a participant has completed the selection. Feedback is enabled if one or more button commands contain the 'c' parameter. With feedback enabled, any button not designated with a 'c' is treated as an incorrect response. It is the experimenter's responsibility to designate the one correct button in each column when column mode is set. Feedback duration is set with the 'F' command (see below).
		Note that the 'n' and 'c' commands are mutually exclusive, so only one of these parameters can be used for a given button.
<i>R G B</i>	0-255	Specify the color tint of the button font using a red, green, blue triplet. Values must be positive integers. Default color is white.
		Note that other colors are fixed: the tablet background is dark gray; button background is light gray; the button color when pressed is light blue; and feedback is given by changing the button font color to green or red, for a correct or incorrect response, respectively.
<i>fontSize</i>	0-255	Font size for the text on the button. Values must be positive integers.
		By default, the application will automatically select a font size that accommodates a word of 6-7 characters on all buttons for which a font size was not specified. This font size will vary according to the size of the various buttons that are visible, and will change if the array configuration is changed.
<i>text</i>	ASCII	All text following the colon is displayed on the button, including spaces.

Text should not include newline characters (\n), tabs (\t), or other escape sequences.

Note that parameters that are not specified in the 'B' command are set to their default value, overwriting previous settings. This is different from the text field, for which the 't' command sets a persistent state (font size, color).

Example: 'B 1 1:Hello world'

Displays the two words on the top left button of the array.

Example: 'B 2 3 d 0 0 255 18:Maybe'

Specifies the button in the second row of the third column. The button is disabled and is given a blue tint with the text rendered in font size 18.

Example: 'B 4 3 d 255 0 0:No'

Specifies the button in the fourth row of the third column; the button is disabled and is given a red tint.

Example: 'B 4 1 c:Yes'

If selected, this button will be shown with green text once the selection is finalized to indicate it is the correct response. Other selected buttons will be shown with red text.

Example: 'B 4 3 16:No'

Specifies the button in the fourth row of the third column using a font size of 16.

b:flag

Command to control all buttons as a group.

Allowed values for *flag* are:

- v Make buttons visible.
- i Make buttons invisible (and so disable them automatically).
- e Enable all buttons. (NOTE: Buttons that were specified as 'non-active' with the 'B' command are not affected).
- d Disable all buttons (but remain visible).
- r Reset all buttons (so they are no longer selected), enable them, and make them visible. Buttons designated as a correct response remain as such. Use 'F:r' to remove the "correct" designation from all buttons, so that no feedback is given. Since the command 'b:r' both resets buttons and makes them visible, it can be used to present buttons at the beginning of a new trial irrespective of the selected display mode.
- x Delete all buttons. (The 'B' command can be used to specify new buttons).

Example: 'b:v'

D:flag

Controls how buttons are displayed and controlled following a response.

Allowed values for *flag* are:

- i Buttons are made invisible, and are disabled and reset to their unpressed state. For trials with changing options, such as questionnaires, the invisible buttons can be deleted with 'b:x' and a new set of buttons can be created.
- d Buttons are disabled and reset to their unpressed state but remain visible. Suited to cases where the same set of buttons is used for every trial.

The command 'b:r' (reset) both resets buttons and makes them visible, so it can be used to present buttons at the beginning of a new trial irrespective of the selected display mode.

Example: 'D:i'

F:[duration] [r]

Feedback parameters.

duration 1-10000 Feedback duration in milliseconds.

This parameter is relevant only if at least one button was designated as the correct response (see the 'c' flag of the 'B' command).

r Reset feedback by removing the "correct" designation from all buttons.

Example: 'F:1000'

Correct and incorrect responses are highlighted in green and red, respectively, for 1 second.

Example: 'F:r'

Reset feedback. Duration remains unchanged for future use.

Example: 'F:2000 r'

Reset so no feedback is given. Feedback will be provided for 2 seconds when buttons are once again designated as being correct with the 'B:' command.

K:flag

OK button is visible or invisible.

Allowed values for *flag* are:

- y The OK button is visible and must be pressed to complete the selection and transmit the result. The OK button is disabled until the user has completed a valid selection (one button from every column in column mode, or any button in either of the other two selection modes).
- n The OK button is invisible and not used. Responses are transmitted as soon as a button is selected in single-button mode ('S:s'), or when buttons have been selected in all columns in column mode ('S:c'). The OK button must be used when any number of selections are allowed ('S:a'). A 'K:n' instruction returns an error message if issued when the unconstrained selection mode 'S:a' has already been set.

Example: 'K:y'

O:orientation

Orientation of the display.

This parameter must be set before any button is pressed on the tablet or the command will return an error.

Allowed values for *orientation* are:

p Portrait.

l Landscape. The flag is a lowercase letter “ell”.

Example: ‘O:p’

S:flag

Types of selections allowed (single selection, column mode, unconstrained selections).

This instruction must be given before any button is pressed on the tablet or an error will result.

Allowed values for *flag* are:

- s Only a single button from the entire array can be selected, i.e., the entire array behaves like a set of mutually exclusive radio buttons.
- c One button per column of buttons can and must be selected, i.e., each column is a set of mutually exclusive radio buttons, each requiring a response.
- a Any number of buttons can be selected from the array. The OK button is automatically made visible when this mode is selected, and cannot be made invisible with the ‘K’ command (see below). The OK button is enabled after the first selection is made.

Example: ‘S:a’

T:text

Text for header area.

Use ASCII characters including \n for line breaks.

Example: ‘T:Do you drink coffee in the morning? \n\nPlease respond truthfully.’

‘T:’ to clear the text field. However, font size and color persist.

t:[R G B] [fontSize]

Text formatting for header.

Parameters: Fonts should be positive integers. R, G, B values must be integers in the range 0-255. The command must include either the font size or the RGB triplet, or both.

Example: ‘t:255 0 0 24’

Sets text color to red and font size 24.

Example: ‘t:16’

Sets font size but leaves text color unchanged.

Example: ‘t:0 64 0’

Sets text color to dark green while font size is unchanged.

Frequently Asked Questions – Troubleshooting

An error message on the master computer indicates that the Bluetooth serial port is not available or in use. In Matlab, the message may read: “A Bluetooth serial failure has occurred. Failed to open an RFCOMM serial channel. Check if authentication needs to be enabled in your device.”

Ensure the serial connection was closed upon ending the previous session. Older versions of Matlab sometimes fail to do this.

Open the control panel on the computer to access Bluetooth settings. Delete the serial port that has been assigned the “SponderSecureComm” channel. Create a new serial port and give it the previous name if desired.

The application fails to accept a Bluetooth connection, even though it did so many times before, and shuts down.

When a connection is accidentally lost and reestablished, the Android operating system may fail to free up resources. After about 20 such cycles, the application may close due to a lack of available Bluetooth sockets. To reset Bluetooth, drag down the panel at the top right of the tablet’s screen and turn Bluetooth off and on again.

The master computer cannot establish a Bluetooth connection with the tablet’s application even though the devices paired successfully.

Ensure the Sponder application is visible and searching for Bluetooth connections when initiating the pairing request from the master computer. If the Sponder application was not visible during pairing, the master computer will not communicate using the required serial connection.

Select ‘unpair’ in the Bluetooth control panels on both the master computer and the Android device, delete the serial port that was previously assigned to the “SponderSecureComm” channel, toggle Bluetooth off and on again on both devices, then repeat the full pairing procedure while the Sponder application is running in the foreground on the tablet.

Application Notes

The tablet will not go to sleep while Sponder is visible.

Appendix A – Pairing the tablet with a master computer.

Begin with the procedure for the Android tablet described here, then follow the procedure appropriate for your computer operating system as described below.

Start both the tablet and the master computer.

On the tablet

From the Apps screen (accessed through the  button on the tablet), go to Settings->Bluetooth. Turn on Bluetooth.

Tap the overflow menu symbol (similar to ) at the top right of the Bluetooth screen and select *Rename tablet*.

Enter a suitable name (e.g., Tablet-1) and press *Done*.

Again tap the overflow menu symbol and select *Visibility timeout* to set a suitable interval (e.g., 5 minutes).

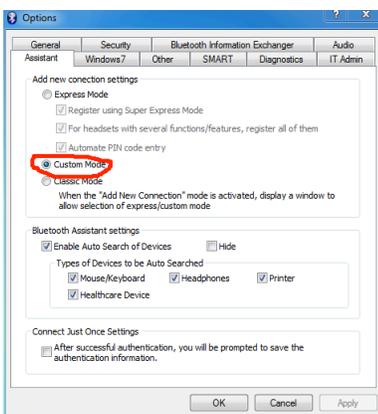
Once back on the main Bluetooth screen, tap on the tablet's new name to make the tablet visible to nearby Bluetooth devices for pairing.

Navigate back to the Apps screen and select the Sponder application, leaving it visible so it advertises availability of the serial port connection.

On Windows 7 or 8 computers

Make sure that a Bluetooth adapter is installed on your computer, either as part of your system or as a USB dongle. Also, be sure to install the appropriate driver.

Go to Control Panel -> Devices and Printers and you should see the Bluetooth Adapter as one of your system's devices. Right-click on it and select Options. This should bring up this window.

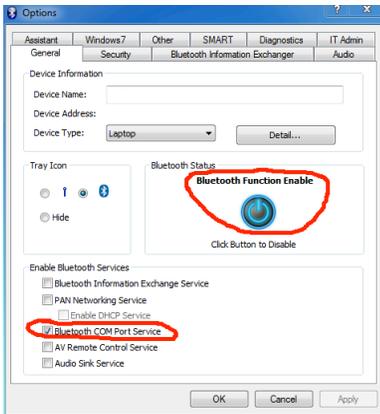


Select the **Assistant** tab and, under Add new connection settings, select **Custom Mode**.

Select the **General** tab: under Enable Bluetooth Services, select **Bluetooth COM Port Service**.

Click the **Bluetooth Function Enable** button to turn Bluetooth on.

Press OK to close the window.

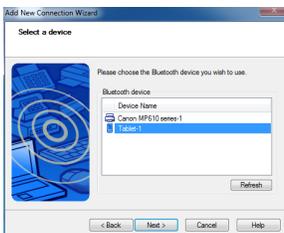


Back in Control Panel -> Devices and Printers. right-click the Bluetooth Adapter and select Add New Connection. This should bring up the Add New Connection Wizard:

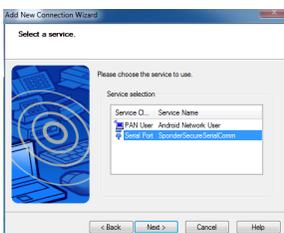


In the Add New Connection Wizard, select **Custom Mode**.

Ensure the tablet is discoverable and the Sponder application is active in the foreground. Press Next in the Add New Connection Wizard. Select the tablet to be connected and press Next.



Select **Serial Port SponderSecureSerialComm** and press Next.



Select **Use default COM port** and note the name of the port, e.g. COM40. Select Next on subsequent windows and Finish.

In the Bluetooth Settings window, right-click the icon bearing the name of the tablet and select Connect.



Agree to pairing with the tablet, and accept the pairing both on the tablet and the PC.

Close the Bluetooth Settings window if desired.

Use the COM port name in the application that will send commands to the tablet.

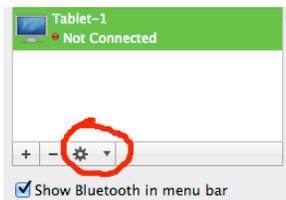
Mac OS X

Go to System Preferences -> Bluetooth.

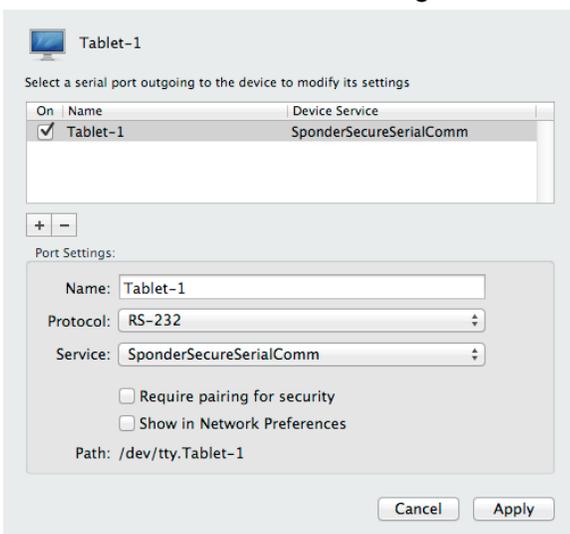
Select the + symbol near the bottom left of the window.

Select the Android device when it appears on the list and accept the pairing in the pop-up window on the tablet. Press Quit on the Mac to return to the Bluetooth preferences window.

Select the gear drop-down menu near the bottom left of the window and select Edit Serial Ports ...



Edit the Name under Port Settings as desired, e.g., Tablet-1.



Note the Protocol is RS-232 and the Service is SponderSecureSerialComm. Also make note of the Path at the bottom of the window.

Click Apply and close the Bluetooth preferences window, then enter the Path name in the application used to open the serial port on the Mac in order to connect to the Android device.

Appendix B – Example using Matlab.

Ensure the Sponder application is visible on the tablet.

On the master computer, start Matlab.

Enter the following commands in Matlab, or re-package them into scripts or functions, as you like:

```
clear all
pause on

s=serial('/dev/tty.Tablet-1','BaudRate', 115200)
% Creates a serial port object s in Matlab.
% 'COMportName' may be something
% like '/dev/tty.Tablet-1' in Mac OS
% or 'COM5' in Windows.
% The maximum baud rate supported on a system may be
% slower than the example given here.

fopen(s); % If successful, a small window will
% appear briefly on the tablet
% announcing the new connection.

while (s.BytesAvailable ~= 0) % Clear out the input buffer
    fgetl(s) % for a clean start.
end

fprintf(s, '%s\n', 'b:x'); % Erase buttons from previous
session.
message = fgetl(s) % Wait for confirmation.
if (message(1) ~= '1') % Check whether command was recognized.
    error(message);
end

% Configure the application.
%-----
fprintf(s, '%s\n', 'S:s'); % Allow only a single selection.
% Buttons act as mutually exclusive
% radio buttons.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\n', 'K:y'); % Use OK button to confirm
selection.
message = fgetl(s)
if (message(1) ~= '1')
```

```

    error(message);
end

fprintf(s, '%s\f', 'D:d');           % Disable buttons after a
response                             % is made, but do not hide them.

message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 'O:l');           % Landscape orientation.
                                       % (Lowercase letter "ell".)

message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 'F:500 r');       % Duration of feedback 500 msec
and
% reset so no button is designated correct, disabling feedback.
% Not applicable in this example,
% but included to demonstrate the
% complete set of configuration
% commands.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

% Specify text and buttons.
%-----
fprintf(s, '%s\f', 'T:Hello World'); % Text that will appear in the
upper
% display area on the tablet.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 't:255 255 0 24'); % Set text font and color.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 'B 1 1:Yes');     % Specify first button.
message = fgetl(s)

```

```

if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 'B 2 1:No');      % Specify second button in first
column.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 'B 3 1:Maybe'); % Specify third button in first
column.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fprintf(s, '%s\f', 'b:r');          % Reset and show the three buttons.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

while (s.BytesAvailable == 0)      % Wait for response from tablet.
    pause(0.01);                    % This can also be done via a callback.
end

% Select a word from the array on the tablet, then press OK.
%-----
fgetl(s)                            % Reads the selected word in the
% format 'R 1: word', indicating that
% this is a response record containing
% the result of one selected button.

% Prepare new trial here.

pause(1.0);                          % For demonstration only.

fprintf(s, '%s\f', 'b:r');          % Present buttons for new trial.
message = fgetl(s)
if (message(1) ~= '1')
    error(message);
end

fclose(s);                            % Ends the session.
% Older versions of Matlab may
% not exit gracefully.

```

Communicating with Multiple Devices

Pair each tablet with the master computer, creating new serial (COM) ports with different names, then open a serial port object in Matlab using the different identifiers for each device. For example:

```
s1=serial('COMportName1', 'BaudRate', 115200)
fopen(s1)
s2=serial('COMportName2', 'BaudRate', 115200)
fopen(s2)
etc.
```

```
fprintf(s1, '%s\r', 'T:Hello 1');
fprintf(s2, '%s\r', 'T:Hello 2');
```

```
% when finished
fclose(s1);
fclose(s2);
```