

Eyedrivomatic:

Software Setup and Getting Started.

Introduction

Eyedrivomatic is a system which allows users of existing eyegaze systems to drive their wheelchair with just eye control. The current beta system has been tested and developed to the stage where it is now successfully used daily by several users.

The philosophy behind Eyedrivomatic is that it can be used on a wide range of wheelchairs, powerfully customizable to account for the differences in various wheelchair controllers. Trim and speed settings accommodate the type of wheelchair (rear, mid or front wheel drive) and their different driving characteristics. Completely customizable actions control various functions such as wheelchair power and mode.

Important!

The testing location should be suitably spacious. We appreciate the potential difficulties of outdoor testing; too sunny for eyegaze, too cold, internet access problems, but empty car parks are ideal locations. We strongly recommend no testing is done without a carer present.

Prerequisites

The Eyedrivomatic system requires an existing wheelchair mounted eyegaze system running Windows and a free USB socket, a powered wheelchair with a standard joystick either mounted on the armrest (some rear mounted attendant controllers may also work).

Hardware and Software

The hardware comprises of 5 components; the mechanical hand, the locator plate, the "mushroom" or joystick sleeve (the post that replaces your joystick top, the "brainbox" and a power pack. The software comprises 2 elements; the control application for your PC and the brain box firmware, an Arduino program which is uploaded to the brainbox. If you received your Eyedrivomatic system pre-assembled, the firmware will be preloaded. If you assembled the system yourself, you will need to download the latest firmware as described below.

If at any point these instructions don't correlate with your system or wheelchair behaviour, or if you're confused about anything, please contact us at <http://www.eyedrivomatic.org/#!/contact-us/r72kq>

Preparation

Once you have received your pre assembled Eyedrivomatic system or assembled your own and downloaded the software from <http://www.eyedrivomatic.org/> & installed you're ready for software setup and initial testing

We recommend that before you strap the electronic hand to your chair, that you get it all setup and working with the system on your lap. Nice and safe. That way you can get used to the system without worrying about crashing.

1. Set the Brain Box and Electronic Hand on your lap.
2. Connect the servo cables on the electronic hand to the corresponding cables from the brainbox, ensuring the top servo is connected to X and the lower servo is connected to Y. Also IMPORTANT ensure the connections are made so the cables wire colours match. If colours on each cable are different, connect so

dark matches dark (e.g. black with brown) and light matches light (e.g. yellow with white)

3. If your chair does not have buddy connectors, skip to step 4.
Connect the 2 control wires between the brainbox and the controller's buddy button sockets, removing any existing buddy button plugs in the process. Socket 1 (on preassembled systems, this will be red) on brainbox should be connected to the controller's on off buddy socket and socket 2 (on preassembled systems, this will be white) should be connected to the controller's mode socket.
4. On the brainbox, connect the PC USB cable to a USB socket on your chair mounted PC. Connect the servo power USB cable to the battery pack, and turn the battery on. (ensure it is fully charged).
5. Download the Eyedrivotronic software from <http://tinyurl.com/jck6sly> and install it.
6. If you received preassembled system, go to step 10.
7. Download the Arduino IDE from the internet.
<https://www.arduino.cc/en/Main/Software> This is the program that we use to upload new firmware to the brain box. Install it.
8. Using Windows file explorer go to
[C:\Program Files \(x86\)\Eyedrivotronic\Firmware](C:\Program Files (x86)\Eyedrivotronic\Firmware)
and double click the Firmware. This will launch the Arduino software.
9. Select the **TOOLS** menu and select the port which it indicates the Arduino is connected to. Verify that you have the Arduino Uno selected. Then select **Sketch -> Upload** (or press **Ctrl + U**) to upload the latest firmware. Wait for it to complete then close the application.
10. From the Desktop, run the Eyedrivotronic shortcut created by the installer.

Screen Explanation

The Eyedrivomatic application is broken up into three main areas. The Status area at the top, a column of navigation buttons on the left and a panel filling up the rest of the area. These are all described in some detail below.

Status Area



At the top of every screen is a status window. This is a vital source of information and **MUST** be checked every time before driving.

The central 3 boxes report form left to right, the duration in milliseconds of left and right buttons, the speed for forwards, backwards and diagonal movements, and the duration in milliseconds of the forwards, backwards and diagonal buttons.

In the top left corner is the Diagonal Speed Indicator, Full for tighter turning, Reduced for gentler turning.

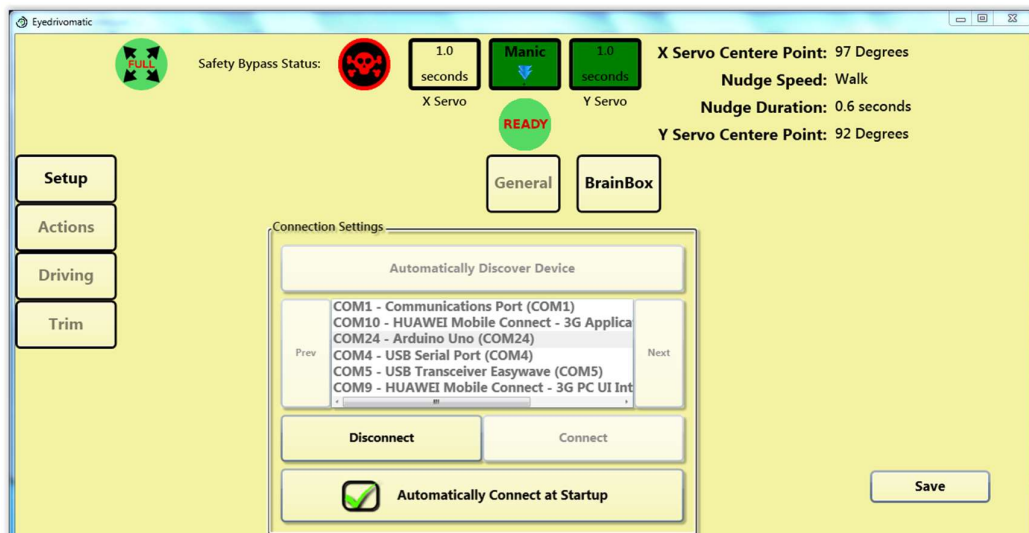
Beneath the Eyedrivomatic speed is the Driving Status. When safety mode is off, it will permanently *Ready*.

On the right are your current nudge and trim settings.

Setup Screen

The Setup screen is used to configure the Eyedrivomatic application. The setup screen has 2 sections; *General* and *Brainbox*. Use the buttons at the top to navigate to these sections.

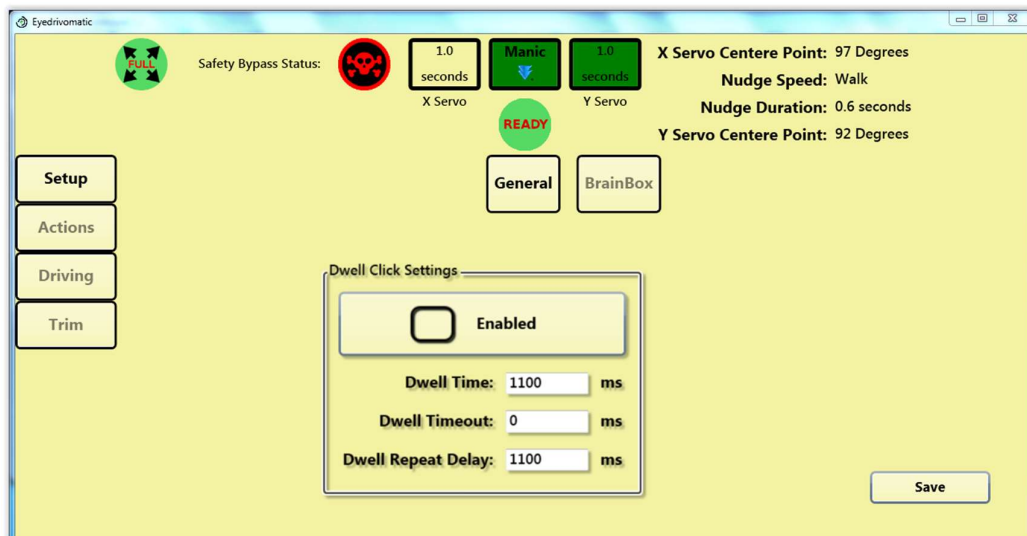
Brainbox Settings



This is where you setup the connection between the software and the Eyedrivomatic brainbox

Either click Automatically Discover Device or select Arduino from the list and click Connect. Select *Automatically Connect at Startup* and click *Save*.

General Settings

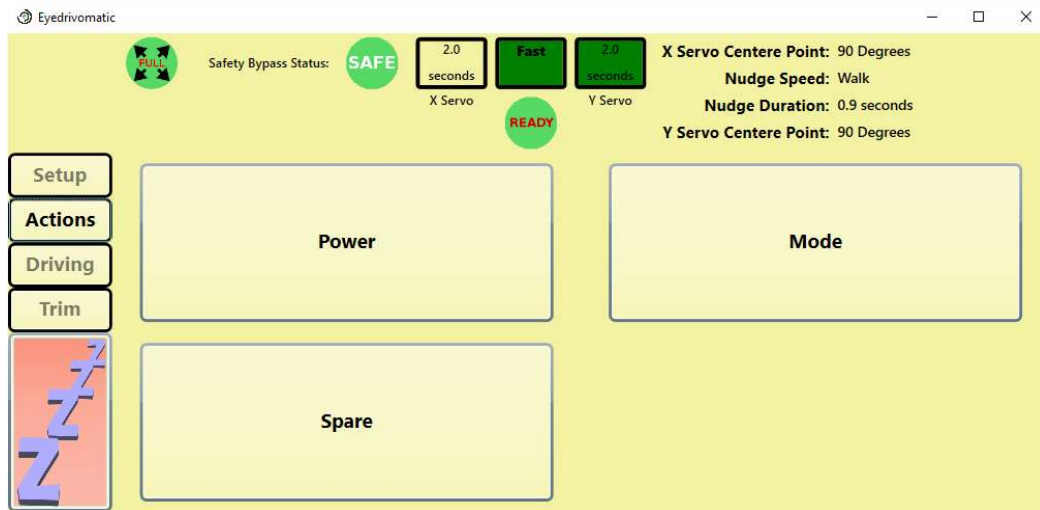


This is where you select whether you use your eyegaze select method or Eyedrivotronic's internal dwell select. In order to use the internal dwell select, you'll need to be able to set your eyegaze select method to move and hover cursor only. This configuration is specific to each eyegaze system and beyond the scope of this document.

If the internal dwell select method is turned on, you can select the dwell time in the boxes. When internal dwell select method is turned on, you'll see a [Rest](#) (Zzzz) button in the bottom left of all screens. We recommend using the internal dwell select method if possible.

Click [Save](#) after you change values.

Actions Screen



This is the standard actions screen. There is one button for each of the outputs from the Eyedrivomatic brainbox.

For these buttons to work, you'll need to be able to connect two of these outputs to your wheelchair controller (The red output labeled 1 on/off and the white output labeled 2 Mode). Although Eyedrivomatic can work without these, you'll need to be able to switch your wheelchair on/off and change mode by another method.

As standard these buttons will turn the wheelchair on/off and advance the wheelchair mode in single steps. These buttons are user configurable by editing the macros.config file in the Eyedrivomatic program folder.

Driving Screen



On the right are the available speed settings for forwards, backwards and all diagonals.

Left and right speeds are constant and unaffected by whether application speed is set at slow, walk, fast or manic.

The speed settings determine the degree of mechanical hand movement, and in conjunction with your controller speed, determines how fast you move.

The green duration buttons select how long forwards, backwards and all diagonals last with one direction button press before the joystick returns to the centre.

The yellow duration buttons on the left select how long right and left movements last with one direction button press before the joystick returns to the centre.

The reset button will instantly cancel any joystick movement.

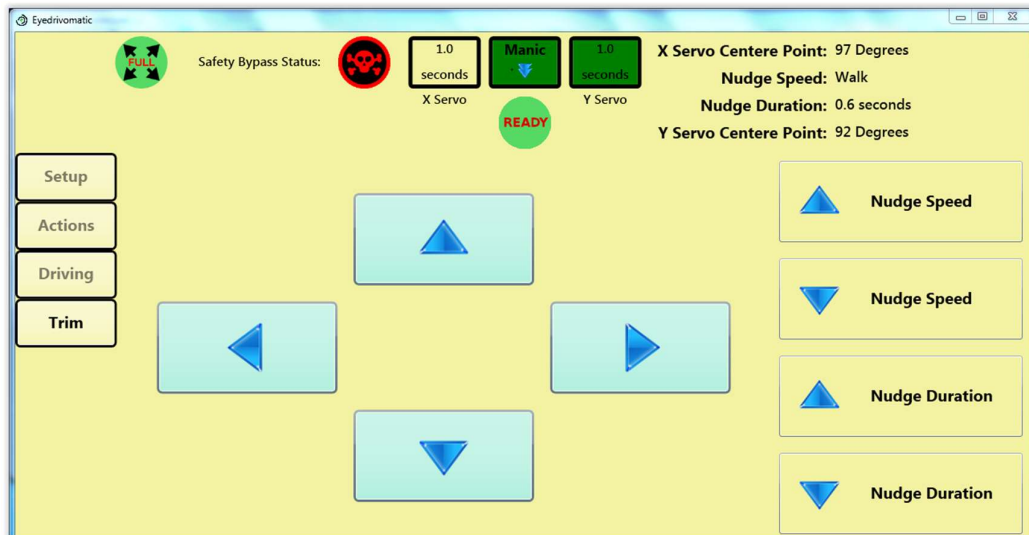
The duration setting will partly be determined by the dwell-click dwell time, if you use dwell selection. To make smooth progression you will need to select a green duration longer than you dwell time.

The nudge buttons are used in conjunction with the forward button. Once setup correctly, the wheelchair should maintain a straight course but terrain will cause the wheelchair to veer occasionally and the nudge buttons will correct this. The amount and duration of the nudge can be set in the trim screen.

The "diagonal speed reduction toggle" button in the trim screen changes the severity of the diagonal movements. The diagonal mode is shown at the top left-hand corner of the status window. The mode should be set to full when maneuvering and reduced for driving. Full diagonal mode causes all diagonal movements to be sharper and more suitable for maneuvering, whereas reduced mode should result in gentler turning when driving outside.

If the internal dwell select method is turned on in General Setup, the rest button will be displayed in the bottom left of screen.

Trim Screen



This is where you can specify nudge parameters and adjust the hand centre (rest) position.

In order not to get an error when turning the wheelchair on and to maintain a straight driving course, you'll need to adjust the trim in both directions.

The rest position of the mechanical hand will be approximately central but you'll almost certainly need to adjust the trim to get the rest position absolutely central.

Nudge settings.

These allow you to set the amount and duration of the nudge button effect.

Nudges are a very effective way of keeping the wheelchair driving straight. They simply nudge the wheelchair left or right very slightly and the effect is determined by these settings.

Pre Wheelchair Testing

Before fitting the mechanical hand to your wheelchair, test it on your lap first. Make sure all directions work on the driving screen and that pressing the buttons on the Actions screen results in audible clicks from the brainbox.

The First Test

Move to an open space with at least 5 metres (15 feet) clearance all round without any obstacles, stairs or any drops such as kerbs.

Fitting the Electronic Hand

1. **TURN WHEELCHAIR OFF!**
2. Remove the top of the existing joystick, including any dust shroud, leave just the rubber gaiter. You should now have just a thin metal shaft.
3. Before fixing anything, you need to find out if any wedges are required. Most controllers have a horizontal section around the joystick with an angled display section. It's important that the square base of the Eyedrivomatic hand isn't prevented from seating properly by the angled display section. To assist with this, there are 3 wedges to raise the locator plate. Put the locator plate over the joystick with the 4 studs positioned squarely. Place the Eyedrivomatic hand on the locator plate (with both servos facing left from the wheelchair user's perspective), ensuring all studs go completely through the corresponding holes in the hand base. Now check if the angled display section is touching the hand base preventing it seating properly. If it is, use one or more wedges to prevent it. Any wedges will need to be superglued to the bottom of the locator plate before proceeding. Be careful when applying glue to not stick the rotating mechanism of the locator plate.

4. Using the double sided foam tape, stick the locator plate centrally over the joystick. Adjust the outer ring, so the screw is facing right from the user's perspective, and the four studs are square relative to the joystick unit.
5. Place the mechanical hand on the plate with the servos on the left from the user's perspective, ensuring the 4 studs are through the holes on the hand's frame.
6. Using the Velcro straps, tie the hand tightly around the controller.
7. If necessary, manually move the hand's arm centrally over the joystick shaft then select the tightest fitting mushroom and slide it through the hole in the arm a over the shaft as far down as possible.

Running and Adjusting the Software

1. **ENSURE WHEELCHAIR IS OFF!**
2. Launch the EyedrivoMatic application from the Desktop.
3. Make sure the battery pack is on and fully charged.
4. If necessary, go to the trim screen and use the arrows to centre the joystick.
5. Go to the driving screen and set both durations to 2 seconds and speed to *Walk*. Select *Full* diagonal mode.
6. Making sure the **WHEELCHAIR IS OFF**, push each of the direction buttons and check there's a corresponding hand movement.
7. Change both durations to 0.5 seconds and speed to *Slow*. Turn wheelchair **ON** and select the controller's lowest speed. If you've plugged in the two control wires, you should be able to this using the on/off and mode buttons on the *Actions* screen.

8. If the joystick isn't precisely central when turning wheelchair on, it's likely the wheelchair will show an error. If this occurs, go to the trim screen and adjust until there's no error.
9. Push forward and assess the resulting movement. With these settings the movement should be **EXTREMELY** slight or none. If it moved significantly, **PROGRESS NO FURTHER** until you have contacted us at <http://www.eyedrivomatic.org/#!/contact-us/r72kq>.
10. Assuming the movement was slight or none, test backwards and diagonals. When testing right and left be aware that the wheelchair might respond quicker than other directions.
11. If you're happy with this stage, increase the speed to *Walk* and test again.
12. If you're still happy, increase both durations to 1 second and test again.
13. If the wheelchair movement isn't sufficient on *Walk* combined with wheelchair controller speed 1, change Eyedrivomatic speed back to *Slow* and increase the wheelchair controller's speed up a level.
14. Keep proceeding like this until you feel comfortable with the *Walk* speed.

Trim – Going Straight!

The final stage in this stage of testing is possibly the fiddliest, adjusting the trim. Once you're confident enough to drive a few metres you will probably notice the wheelchair isn't driving straight. **NOTE:** When you first start driving, the wheelchair won't necessarily initially move in the precise desired direction, because the wheelchair's castor wheels may need to straighten or flip direction. During this process the wheelchair will take a short time to steer true.

There are two methods of correcting this. The first is using the trim screen; the second is by turning the adjustable plate which the hand sits on, by loosening the screw, rotating the hand before tightening screw again. You will probably need to do a little of both to achieve a straight, forward course AND achieve a balance between right and left movements to ensure both directions turn equally.

Once you've finished, we recommend you select *Safe Mode* from the settings screen. This will require you to select the *Continue* button between every arrow press. Although it's fractionally more fiddly, it will prevent inadvertent moving.

With the correct combination of software speed and wheelchair speed, you should be able to make very smooth progress. If you're unable to find this combination, you might need to contact your wheelchair technician to adjust the internal software of the wheelchair controller. It's a very simple job and they can setup a 'soft start' configuration which reduces the initial acceleration.

And that's it. EyedrivoMatic does take time to get used to and you need to pay full attention to it. Never attempt to use EyedrivoMatic outside in sunny conditions and never use it near hazards such as curbs or steps or automotive traffic.

Appendix A – Actions

The list of actions on the *Actions* screen, and even the Power button on the *Driving* screen are fully customizable. These actions are defined in the *Macros.config* file located in the program directory (usually *C:\Program Files\Eyedrivomatic*).

Macros.config File Format

The *Macros.config* file is a UTF formatted XML document that contains a list of *Macros* (*Actions* in the user interface). Each macro in turn has a list of one or more *Tasks*. The default Macros.config file is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<Macros xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <Macro DisplayName="Power" Icon="Images/Off.png">
    <Tasks>
      <CycleRelay DisplayName="Cycle relay 1." Relay="1"/>
    </Tasks>
  </Macro>

  <Macro DisplayName="Mode">
    <Tasks>
      <CycleRelay DisplayName="Cycle relay 2." Relay="2"/>
    </Tasks>
  </Macro>

  <Macro DisplayName="Spare">
    <Tasks>
      <CycleRelay DisplayName="Cycle relay 3." Relay="3"/>
    </Tasks>
  </Macro>

</Macros>
```

In this file, there are listed three macros; “Power”, “Mode” and “Spare”. Each of these macros contain a single action that cycles a relay once.

Macro

Each Macro element defines an “Action” that will be displayed on the *Actions* screen. The first defined macro will be visible on the *Driving* screen. If an icon is specified for the macro, it will also be displayed. Each macro contains a list of one or more *Tasks*.

DisplayName: The text drawn on the action buttons.

Icon: An image to be drawn on the action button. Currently only implemented for the button on the *Driving* screen. The value of Icon should either a fully qualified path, or a path relative to the application directory.

Tasks: A list of specific steps to be executed sequentially. See Macro Tasks for information about the possible tasks.

Macro Tasks

Each macro may have one or more tasks. These tasks are executed sequentially. Each task has a defined *DisplayName*, which is not currently used in the application.

CycleRelay Task

The Cycle relay task cycles the defined relay on for 200ms then back off. If Repeat is defined and greater than one, Delay must also be defined but may be zero. The relay is cycled the specified number of times (Repeat=”1” means the relay is cycled once, which is the default) with the specified delay between cycles.

```
<CycleRelay DisplayName="Cycle relay 3 twice with a 250ms delay."  
  Relay="3" Repeat="2" Delay="250"/>
```

DisplayName: Not currently used.

Relay: The relay number to cycle.

Repeat: The number of times to cycle the relay.

Delay: The delay in milliseconds between relay cycles. (note: there are 1000 milliseconds in a second)

Delay Task

This is a simple task that simply Inserts a delay between other tasks. For example, you may wish to cycle relay 2 once, then cycle relay 3 after a half second delay.

```
<CycleRelay DisplayName="Cycle relay 2." Relay="2"/>  
<Delay DisplayName="Wait 500ms." Time="500"/>  
<CycleRelay DisplayName="Cycle relay 3." Relay="3"/>
```

DisplayName: Not currently used.

Time: The time to delay in milliseconds.