

H-Drive manual

Citation for published version (APA):

Janssen, B. R. A. (2005). *H-Drive manual*. (DCT rapporten; Vol. 2005.091). Technische Universiteit Eindhoven.

Document status and date:

Published: 01/01/2005

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

H-Drive manual

B.R.A. Janssen

DCT.2005.91

In addition to my master's thesis "Output Regulation for a Nonlinear Mechanical System: From Design to Experiments", DCT no. 2005.41

EINDHOVEN UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF MECHANICAL ENGINEERING
DYNAMICS AND CONTROL GROUP

Eindhoven, 14 June 2005

Contents

Preface	v
1 History	1
2 Hardware	5
2.1 Coordinates	5
2.2 Rotational Arm	5
2.2.1 Motor (de-) Mounting Instructions	8
2.2.2 Allowable Gearbox Loads	9
2.2.3 Additional Holes	10
2.2.4 Front Lock plate	10
2.2.5 Bearing Adjustment	10
2.3 Electronics	11
2.3.1 H-bridge	11
2.3.2 Pendulum	11
3 Operation	15
3.1 General Control Layout	15
3.1.1 Improvements	17
3.1.2 Emergency Stop	20
3.1.3 Friction and Cogging Compensation	21
3.1.4 Initialization Procedures	22
3.2 Getting Started	22
3.3 Known Problems	23
3.3.1 Offset in C-code	23
3.3.2 Computation time	24
A System States	27

Preface

This report, “The H-bridge Manual”, covers all the things one needs to know to operate the H-bridge, including the actuated pendulum that has been fitted onto the H-bridge X-carriage within the scope of the master’s thesis study of the author, “Output Regulation for a Nonlinear Mechanical System: From Design to Experiments”, [5]. The manual is meant to give a good basis to start experiments on the H-bridge. I hope this manual will avoid a long search for machine data, known bugs and operating instructions that the author has experienced during his master’s thesis study on the H-bridge.

This manual has been written with the intention to transfer as much knowledge and data as possible in order to make the startup of practical research on the H-bridge as convenient as possible for future users. Since the time to write this manual is limited, this may come at the cost of neatness and clarity in the manual. My apologizes for any sloppiness, however, in my opinion completeness is more important here.

Many studies have been performed on the H-bridge since it has become operational (See Chapter 1, which describes the H-Bridge history). Also many practical issues concerning the H-bridge have already been sorted out and set down in master’s thesis and PhD reports. For some issues, therefore, this manual refers to earlier reports. This manual covers a general overview that is required for startup of the H-bridge setup and a more detailed description of features that are not covered in earlier reports.

Since you are reading this manual you are probably planning to do some experiments on the H-bridge. I would like to wish you good luck. The H-bridge is a bit of a laborious machine to operate, but once it all works, it will give you good experimental data. I hope this manual makes the startup of experiments on the easier to you than it has been to me.

Bart Janssen

Eindhoven, June 2005

Chapter 1

History

The H-bridge has been donated to the DCT lab by Philips around 2000. The H-bridge is the main part of a component pick-and-place machine, the ACM Micro (part of original service manual is available, see [11]). This machine is originally intended to pick small electronic components and to place them on printed circuit boards that pass through the machine on a conveyor belt. The original H-bridge electronics have been removed and replaced by a dSpace system to be able to actuate the H-bridge with custom built controllers.

Important to know is that a similar H-bridge setup is available at the TU/e department of Electrical Engineering (building E-laag). At the time the manual is written this setup is operated by Nelis van Lierop (C.M.M.v.Lierop@tue.nl), Helm Janssen (J.W.Janssen@tue.nl) and some other guy (contact information is lost).

To get an idea of what research has been done on H-bridge I'll give a short description of the H-bridge history (Tabel 1.1 gives a summary of this overview). Around 2000 the H-bridge has been donated to the DCT lab by Philips. During the first projects on the H-bridge the original electronics have been removed and the setup has been (electronically) adapted such that it can be actuated and controlled via the dSpace system completely. Moreover, an initialization procedure (in order to “synchronize” the linear motor commutation and to home and to initialize the encoder readings) and a safety layer (including current and speed limitations, angle violation detection, virtual airbags and emergency stop procedures) are implemented in C-code. This work has resulted in a H-bridge setup that is fully operational with an easy to use Simulink block in which all the hardware connections, initialization procedures and safety layers are implemented. The studies that cover the major part of this work are those of Antoine Verweij [12], A.F. Rovers [8] and S.G.M. Hendriks [3]. Moreover the working principle of the linear motors is explained in the reports mentioned above. (see also Table 1.1).

In subsequent research the H-bridge has been mainly used for friction and friction identification experiments. Especially Ron Hensen [4] (but also Bjorn Bukkkems [2]) has performed elaborate experiments on these topics. Pages 120-130 in [4] give a good description of the friction and cogging phenomena and their identification process.

For the study of N.P.I. Aneke (“Control of Underactuated Mechanical Systems”, [1]), the H-bridge X-sled has been fitted with a small non-actuated rotational arm in the horizontal plane. However this rotational arm has later been removed, the rotational arm, including its mount to fit it on top of the X-sled “electronics box” and a special piece of interpolation hardware is still available in one of the drawers near the H-bridge. If desired it can be re-mounted easily, even when the large actuated rotational arm, which has been mounted recently, is in place. Also the encoder cable is still in place and could easily be reconnected.

Due to lack of free disk space, in December 2003, there has been a major file removal on the host PC that was connected to the dSpace hardware at that time. Unfortunately the person that performed the file deletion did not backup the removed data. This has resulted in an important loss of information. In the beginning of 2004 there has been a hardware and software update. The dSpace power PC cards have been removed from their host PC and they have been placed in a separate box that is connected to the host PC through an optical bus (orange cable). Moreover, the host PC has been replaced by new Dell machine. The latter replacement significantly increases the "building" speed with which the Simulink models are compiled. The data from the original host PC that survived the file removal can be found on the current Dell host PC on drive D: in the folder `Old H-bridge PC`. Along with the hardware update, a software update has been performed. Matlab 5.3 has been replaced by version 6.1 and also the version of Control desk has been updated. This makes that some older stuff does not work on the new setup anymore.

Because the cogging and friction identification data had been lost during the file removal, both have been re-identified (for the X-sled only) by Yasin Talukder [10] in the beginning of 2004.

In 2004 the H-bridge setup at the department of electrical engineering has been made operational in a similar way as this has been achieved for the machine in the DCT lab. Nelis van Lierop and Helm Janssen that perform this operation note however have informed me about some improvements they have implemented. Firstly, there is a temperature sensor in the coils of the linear motors. In the setup in the DCT lab this sensor is currently not connected. It is advisable, however, if one is going to operate the H-bridge in a more demanding way (very large accelerations, large "holding" forces, etc.), to connect the sensor in order to be able to monitor the coil temperature and prevent overheating. So far, the H-bridge has not been used "to the limit", which makes it was not necessary to connect the sensor. Connecting the sensor should be not too hard, especially since the hardware and the cables are already in place. Secondly, the C-code that encompasses the H-bridge initialization and the safety layers has been re-written by Nelis and Helm in order to make it more neatly arranged. Thirdly they implemented a more accurate coil commutation protocol that not only uses sled positions, but also sled velocity to determine the proper coil commutation. This commutation protocol achieves higher cart position accuracy with respect to the current commutation protocol on the H-bridge (see [8] and [3]), which only takes into account cart position. If one requires high-precision H-bridge performance it is advisable to have a look at this protocol that is used by Nelis and Helm.

The most recent study on the H-bridge has been the author's master's thesis study. For this study the H-bridge X-sled has been fitted with a large actuated rotational arm. The main idea is to actuate the H-bridge X-sled via the reaction forces of the rotational arm on this X-sled. More information on this study, as well as many data on the H-bridge and rotational arm hardware can be found in [5] (especially appendix H). During the last study one has tried to make up for all the lost information and to "clean things up". This manual is part of the results.

Table 1.1: Studies on the H-bridge: an overview.

Number	Study	Author	Useful info
2000.37, [3]	Iterative Learning Control on the H-drive	S.G.M. Hendriks	Commutation protocol
2000.26, [6]	Learning Control on the H-drive	J. Janssens	-
2000.??, [12]	Control of a Permanent Magnet Linear Motor with dSpace and Matlab/Simulink	A.H. Verweij	-
2002.12, [8]	Controlling the H-Drive	A.F. Rovers	Initialization and safety layer implementation
2001.35, [2]	Friction Induced Limit Cycling: An experimental Case Study	B.H.M. Bukkems	Friction and Cogging identification
2002 (PhD), [4]	Controlled Mechanical Systems with Friction.	R. Hensen	General description setup, friction and cogging.
2002.31, [13]	LPV Control Based on a Pick and Place Unit	A-J. v.d. Voort	-
2003.78, [7]	Experimental Modal Analysis of the H-drive	R. Merry	Modal analysis.
2003 (PhD), [1]	Control of Underactuated Mechanical Systems.	N.P.I. Aneke	X-sled angular DOF
2004	File removal, hardware and software update	P. Lambrechts	(no report)
2004.68, [9]	Observer Design for Systems in Second-Order Chained Form	R. van der Steen	-
2004.37, [14]	Input Output Linearization on the H-Drive Including Tilt of the Beam	R.M.T. Wouters	-
2004.??, [10]	Identification of Cogging and Friction in the H-Bridge X-Axis.	Y. Talukder	Cogging and Friction Identification
2005.41, [5]	Output Regulation for a Nonlinear Mechanical System: From Design to Experiments.	B. Janssen	Rotational arm data, general H-bridge data overview.

Chapter 2

Hardware

The current H-Drive setup consists of several parts: the “original” H-Drive, the rotational arm that has been added in a recent study on the H-Bridge [5] and the amplifiers and dSpace hardware that control the setup. For more information on the “original” setup one is referred to the original manual [11] but also section 5.1 in[5] gives a description of its main features. Below is a description of the coordinate system, the electronics and the rotational arm.

2.1 Coordinates

During the initialization procedure of the H-bridge (see also Section 3.1.4 for details on the implementation of this initialization procedure) the encoders are reset, the machine zero position is determined and the control zero position is set. Also the direction of the axis is set. The zero position and direction of axis has been fixed as indicated in figure 2.1. Note that this coordinate configurations implies that one always operates on the “negative” x-axis.

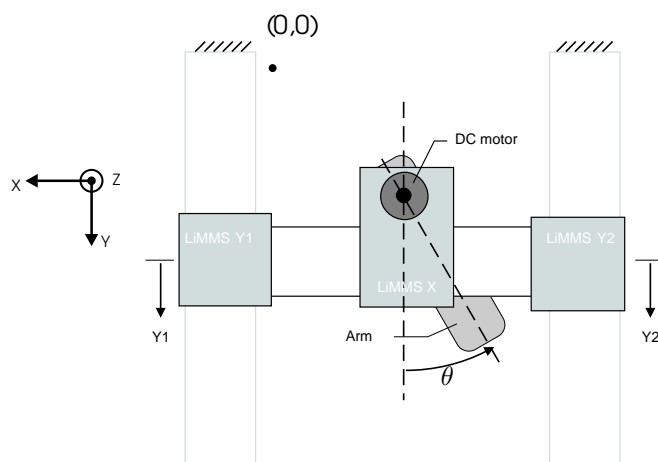


Figure 2.1: H-bridge schematically, direction of axis and zero position.

2.2 Rotational Arm

Figure 2.2(a) shows a CAD drawing of the rotational arm design and Figure 2.2(b) shows the rotational arm as it has been realized. Table 2.1 gives the part names and numbers.

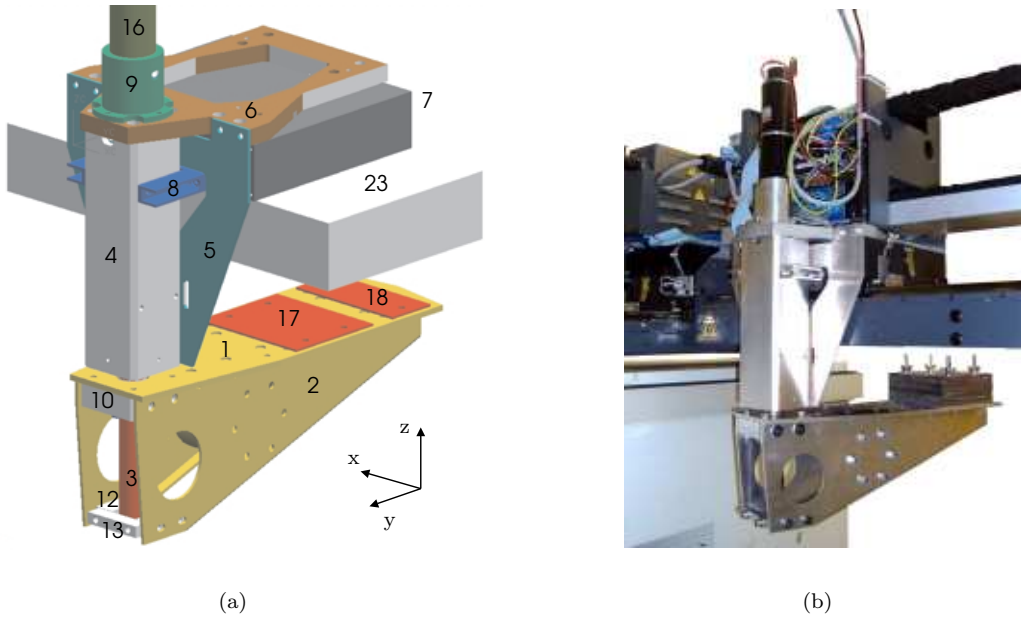


Figure 2.2: The TORA (Translational Oscillator with a Rotational Actuator) system: design (a) and the actual setup (b).

Table 2.1: Part names and numbers.

Part No.	Part	Part No.	Part	Part No.	Part
1	pendulum table	9	motor support	17	weight large
2	pendulum side	10	clamp-hub block	18	weight half
3	pendulum shaft	11	clamp-hub	19	bearing-lock-nut
4	bearing house	12	connection block	20	angular contact bearing
5	support plate	13	lower pendulum lock	21	deep groove bearing
6	base plate	14	torsion tube	22	lock ring
7	cart translator	15	connection tube	23	X-guidance
8	mounting block	16	motor-gearbox assembly	24	flexible coupling

Figure 2.3(b) gives a cross section of the rotational arm and its support structure. Figure 2.3(a) gives depicts a more detailed cross section of the pendulum bearings, the coupling and the connection to the motor.

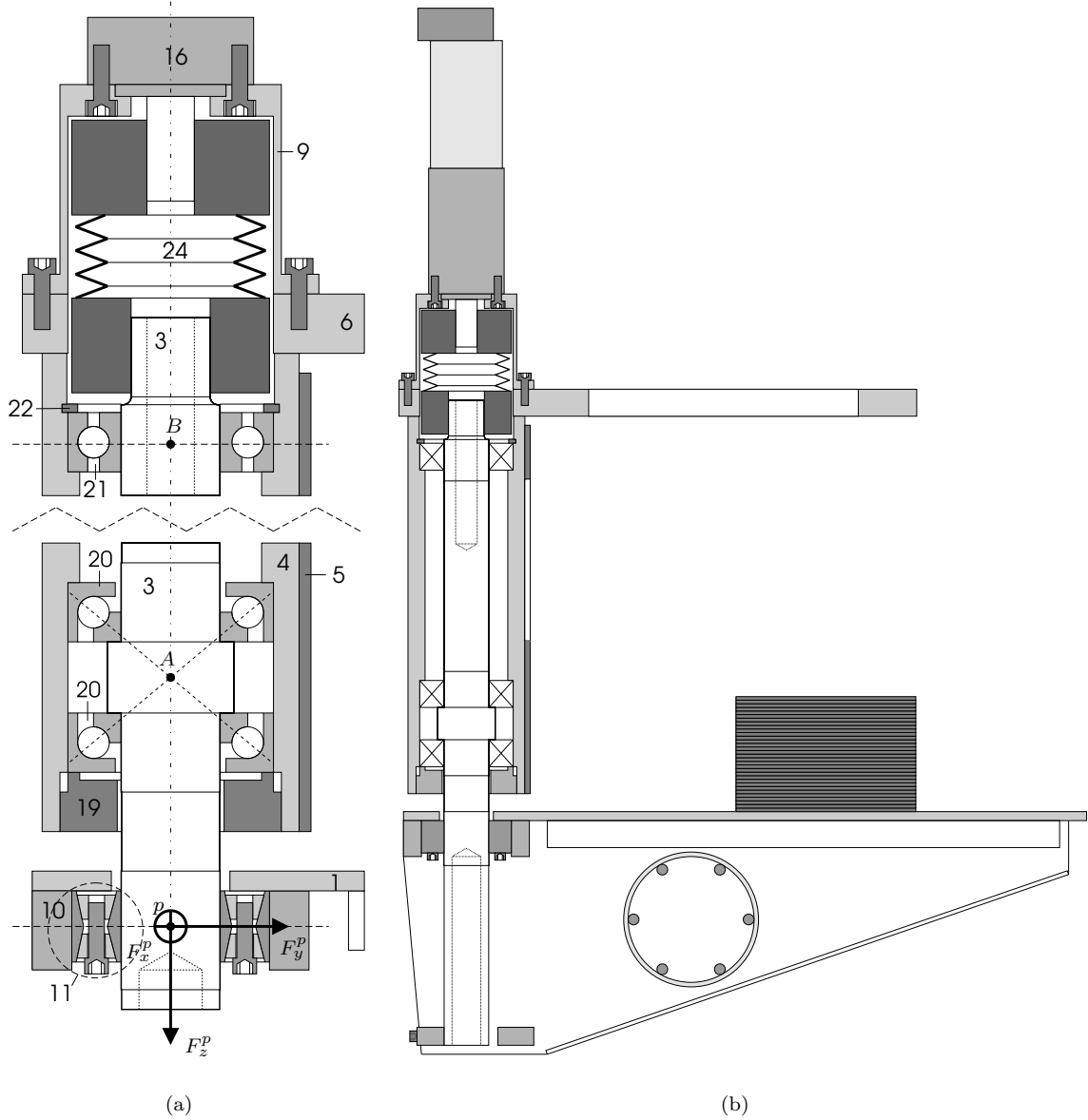


Figure 2.3: Pendulum design cross section (Y-Z plane). Total of pendulum, support structure and motor (b), bearing and flexible coupling detail (a).

Moreover, figures 2.4(a) to 2.4(c) show the structure of the actual arm.

For detailed component drawings (including all dimensions), one is referred to Appendix G in [5]. Appendix F in [5] gives the specifications of the motor, gearbox, encoder, clamp-hub and flexible coupling. Finally, tables H.2-H.3 in Appendix H in [5] give the values of the masses and inertias of the different parts and components. It should be noted, however, that many mass and inertia parameters in these tables are based on the CAD drawings and have never been verified or identified for the actual rotational arm as it has been constructed.

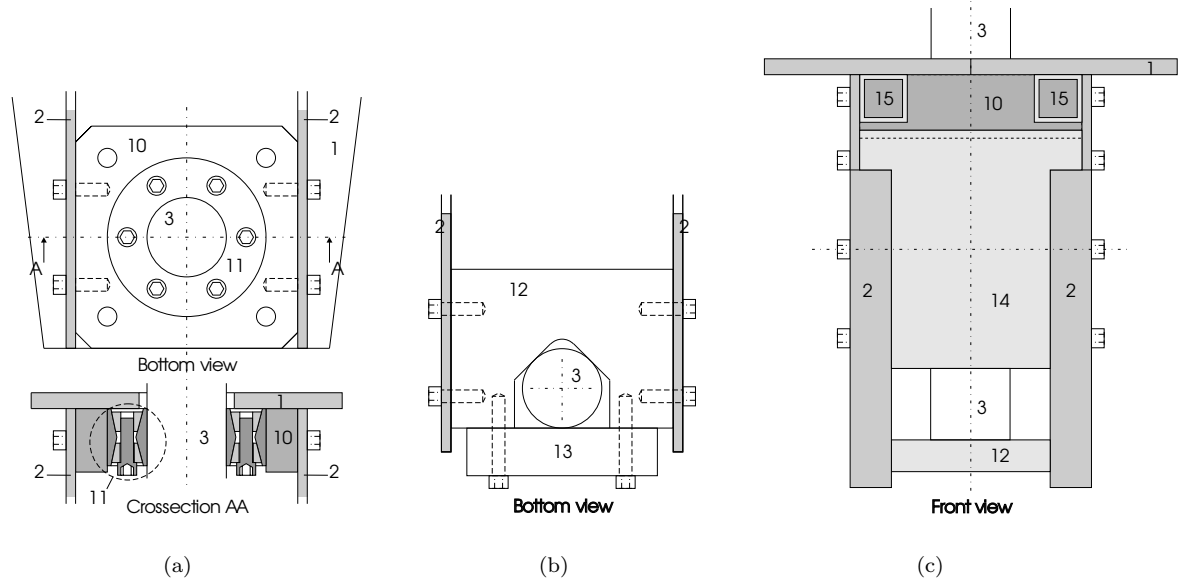


Figure 2.4: Upper pendulum-shaft connection bottom view and cross section (a), lower pendulum-shaft connection bottom view (b), pendulum front view (c).

The figures presented in this section, together with the detailed drawings in Appendix G in [5] should be sufficient to understand the working of the rotational arm. A complete description of the design can be found in Section 4.5 in [5]. Some features however need to be discussed in more detail.

2.2.1 Motor (de-) Mounting Instructions

Consider the connection of the shaft to motor via the flexible coupling and the connection of the motor to the base plate via the motor support, which are depicted in detail in in Figure 2.3(a).

Both the connection of the flexible coupling to the motor shaft and the connection of the flexible coupling to the pendulum shaft are realized using the clamp hubs at each end of the flexible coupling (see also figure F.5 in [5]). However the rated torque for this type of coupling (with a motor and pendulum shaft diameter of 12 and 24 mm respectively) is 42 Nm (which is sufficient for operation of the rotational arm), slip did occur in the clamp hup connecting between the motor shaft and the flexible coupling¹. To solve the slip problem, the flexible coupling has been fitted with a slot. Together with the gib that has been placed in the motor shaft this has resulted in a solid slip free connections. Note that the clamp hub connection between the motor shaft and the flexible coupling is still operational, but now only provides an additional clamping force to the gib an cotter connection.

Tot demount the motor/gearbox combination one needs to go through the following steps:

1. Manually turn the rotational arm in the right position such that the bolt of the clamp hub connecting the flexible coupling to the motor shaft is visible through the hole in the side of the motor support.

¹A clamp connection had been chosen intentionally in order to prevent play that would possibly result from a gib and cotter connection

2. Unscrew the bolt in the clamp hub through the hole in the side of the motor support. It only needs to be unscrewed for one or two circumvolutions. If it is unscrewed too far, the rotation of the whole coupling will be jammed.
3. Manually turn the rotational arm in the right position such that the bolt of the clamp hub connecting the flexible coupling to the pendulum shaft is visible through the hole in the side of the bearing house, just below the base plate.
4. Unscrew the bolt in the clamp hub through the hole in the side of the bearing house. It only needs to be unscrewed for one or two circumvolutions.
5. Unscrew the four bolts connecting the motor support to the baseplate.
6. Carefully lift the whole structure of motor, motor support and flexible coupling from the pendulum shaft.
7. Because of the presence of the gib in the connection of the flexible coupling and the motor shaft, the flexible coupling will not come loose from the motor shaft directly .
8. Gently force the flexible coupling from the motor shaft using a small screw driver as lever that can be put between the “bottom” of the motor support and the upper part of the coupling through the hole in the side of the motor support. Note that to large axial loads, or hammering on the gearbox shaft and/or the flexible coupling will destroy the gearbox and/or coupling!
9. The combination of motor and gearbox can now be unscrewed from the motor support. *The motor and gearbox cannot be separated!*

For mounting of the motor/gearbox assembly one goes through the above steps in reverse order.

2.2.2 Allowable Gearbox Loads

Apart from the problem of slip between the motor shaft and the flexible coupling, the only other mechanical problem that has been encountered during operation of the H-bridge is the “malfunction” of the gearbox. To understand what is going on, the history of the gearbox is given below.

In the design process the Maxon GP42C has been selected as the best gearbox to drive the pendulum. Its torque ratings are 15 Nm continuously and 22.5 Nm intermittently. According to the manufacturer, exceeding of these ratings would not yield permanent damage to the gearbox, but only shorten its lifetime. The predicted gearbox loads (see Chapter 4 in [5]) are quite large compared to the gearbox torque ratings, but, as one can see from the experimental results (see Chapter 4 in [5]) the torques applied during the actual operation of the rotational arm are far below the continuous torque rating.

In one of the first trials after installation of the rotational arm, however, the gearbox failed: it jammed. There are three possible causes:

1. The mounting and de-mounting of the flexible coupling in order to fit the gib, as described in the previous subsection 2.2.1 has caused damage to the motor/gearbox.
2. The gearbox has been damaged by an overload.
3. The manufacturer torque ratings are too optimistic.

Since the applied torque during the first tests is completely unknown, the exact cause cannot be determined. Maxon, the motor manufacturer, was willing to help and put a new gearbox to the motor in order for us to be able to proceed with the experiments directly. This motor/gearbox assembly has been remounted to setup and is still in place. Ever since the repaired unit has been

in place the applied motor torques have been limited to continuous torque limit of the gearbox (15 Nm). Despite the torque loads have never exceeded the continuous torque limit of 15 Nm, the gearbox again starts making strange noise. It tends to get worse, but as long as it does not jam this will not be a problem.

Because of the jamming of the first gearbox, Maxon also supplied us with a complete new unit of motor, gearbox and encoder. This “splinternieuwe” unit is still in a drawer near the H-Bridge setup and has never been used. In case the current gearbox fails, it could serve as a backup.

The conclusion is that one should be careful with the gearbox. It seems that it cannot handle the magnitude of the torques it is rated for. If it would jam, however, there is a spare unit available.

2.2.3 Additional Holes

One possibly wonders what the purpose is of the holes in the side of the bearing house, just below the beam that forms the H-bridge X-axis. These are holes that can be used to mount additional parts in order to solve any strength or stiffness problems in the pendulum support structure in Y direction that might occur. Fortunately, during operation of the pendulum, pendulum strength and stiffness have always been sufficient.

If one, for some reason, does require additional strength / stiffness in Y-direction one could do the following. One could add a similar structure as the bearing house to the front of the base plate and the pendulum X-sled. Two additional “mounting blocks” to do so are available in the drawer near the H-bridge. The two “bearing houses” then can be connected by some bars just below the beam forming the H-bridge X-axis, thus forming a structure that is stiffer and stronger in Y-direction than the current “single bearing house” structure.

2.2.4 Front Lock plate

The aluminium plate in front of the X-sled originally was meant to fix the pendulum when operating the H-bridge only *and* to serve as a “fixed” point onto which the pendulum could be aligned. The latter function now has been replaced by the pendulum initialization procedure which is described in Section 3.1. The plate still has the function of locking the pendulum motion. Originally the magnet on the plate was intended to hold the pendulum table (via a small piece of steel at the pendulum side). This magnet, however, turned out not to be strong enough to fix the pendulum during violent H-bridge motion. The current solution is to fix the pendulum to the plate with the use of tie-wrap. This suffices to fix the pendulum, but is not a neat solution. It would be nice to make a new lock plate, which is similar to the current one, but with a bent border in order to increase strength and stiffness and with a fast lock/unlock mechanism to fix the pendulum position.

2.2.5 Bearing Adjustment

As can be seen from Figure 2.3(b) the main bearings that support the pendulum shaft are a set of angular contact bearings in X-configuration. The play or pre-tension of this set of bearings can be adjusted using the bearing-lock-nut that is screwed in the bottom of the bearing house. A specially fabricated tool to turn the bearing-lock-nut is available in the drawer near the H-bridge. Note that the use of this tool does not require the actual arm to be demounted from the shaft. It may be a bit of a hassle, but it is possible to get the tool in the right position in the space between the pendulum table and the bearing house in order to loosen or tighten the bearing-lock-nut.

The bearings have been adjusted such that they are under slight pre tension.

2.3 Electronics

Figure 2.5 presents a schematic representation of all the control hardware components of the current H-bridge setup. Table 2.2 lists the component specifications.

2.3.1 H-bridge

Lets first discuss the H-bridge hardware. The H-bridge linear motor units are fed 300 VDC, which is generated by the H-bridge amplifier (the large white box on the left side of the table). The three phase motor current for the linear motor itself is generated by the linear motor amplifier that is onboard every linear motor. The current level and the commutation are prescribed by reference signals from the dSpace system. All amplifier gains are given in Table 2.2.

The H-bridge position is measured using linear encoders with a 1 μm resolution. These encoders are connected to the standard dSpace encoder inputs (Inc1, Inc3 and Inc5).

Moreover the H-bridge is equipped with some safety sensors (see [8] for details).

- End Of Stroke (EOS) sensors (micro switches) indicate the ends of the stroke of the X, Y1 and Y2 axis.
- End Position Detectors (EPD) (inductive) indicate the H-bridge zero position.
- A set of sensors monitor the tilt of the beam that forms the X-axis.

As mentioned in Chapter 1, the linear motor coils are equipped with temperature sensors. These, however, are not connected.

2.3.2 Pendulum

The pendulum is driven by a small DC motor fitted with a gearbox. The selection of the motor a Maxon RE40, 48 V, 150W is extensively addressed in [5]. Appendix E in [5] gives the full motor specifications. The motor is fed via an analogue current amplifier (white box on top of the stack of amplifiers). This current amplifier is a 24 volt amplifier that can generate up to 4 Amps continuously. It needs a reference signal between -2.5 V and + 2.5 V. This reference signal is prescribed by the controller and then generated by the dSpace system. The amplifier gains and motor constant are in Table 2.2.

Then there is the digital Maxon 4-quadrant current amplifier. This amplifier was originally intended to use with the RE40 motor. Because this amplifier is a 48 V amplifier (whereas the analogue one is only 24 V), this would have resulted in maximum motor performance. The use of the digital amplifier, with its 50 kHz switching frequency resulted in noise in the encoder readings of the H-bridge. The 48V signal from the amplifier to the motor interfered with the signals in the H-bridge encoder cables. Even the use of shielded cable did not solve the noise problem. Also feeding the motor with an external cable (via the ceiling in stead of squeezing the cable in the gutter with all other cables) did not solve the noise problem. With the use of the analogue amplifier the noise problems were solved. Since the external cable was already in place from the attempts to feed the pendulum motor via the digital amplifier, this external cable has also been used for the analogue amplifier. Probably the signal from the analogue current amplifier can also be fed to the motor via the cable that is squeezed in the gutter with all other cables, but this has never been tested. The noise problems connected to the use of the digital amplifier have never been studied in full detail. With some decent knowledge on electromagnetic fields interference, however, it should be possible to solve the problem.

The former use of the digital amplifier has lead to some remains in the current setup. There is still al large DC power supply to provide the digital amplifier with 50 VDC and there are some

connections that are not used anymore. Besides the current reference signal, a physical enable signal needs to be sent to amplifier and the amplifier is equipped with a current sensor which makes it possible to read back the current that is actually realized by the amplifier.

One should note that the specifications on the digital Maxon amplifier, which are with the H-bridge indicate a peak capacity of 10A. This is incorrect. Since the amplifier outputs only 0.8 A/V of the reference signal and the maximum reference signal potential is 10 V, the maximum amplifier output is only 8 A (this has been ascertain by the people form Maxon).

The Maxon motor / gearbox assembly is fitted with a Maxon incremental encoder. This encoder has 500 lines per revolution and is fitted with a line driver, which makes the encoder signal less sensitive to any disturbances form other signals or interference. For the exact specifications and wiring schemes one is referred to Appendix E in [5].

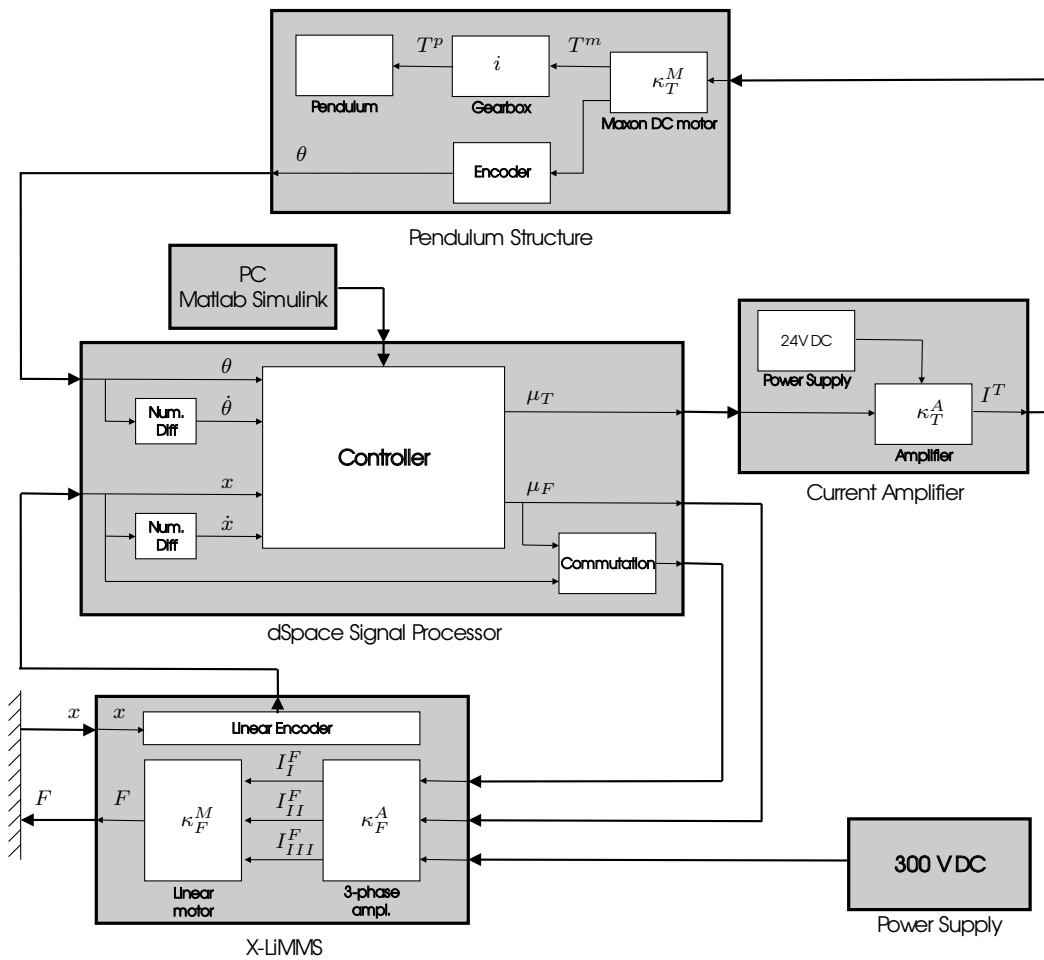


Figure 2.5: Schematic representation of the control hardware components and connections.

Table 2.2: Controller hardware component specifications.

Feature	Make	Type	Specs
Linear encoder	Heidenhain	LIDA 201	resolution: $1\mu\text{m}$
Rotational encoder	Maxon	HEDL55	resolution: 0.18° deg
Pendulum analogue current amplifier	?	?	amplification: $\kappa_T^A = 1.6 \text{ A/V}$ output voltage: -24 to 24 V DC input voltage: -2.5 to 2.5 V DC
Pendulum digital current amplifier ^a	Maxon	ADS 50/5	amplification: 0.8 A/V output voltage: -48 to 48 V DC input voltage: -10 to 10 V DC
Gearbox	Maxon	GP42C	gear ratio: $i = 1 : 113$
DC motor	Maxon	RE40	amplification: $\kappa_T^M = 60.3 \cdot 10^{-3} \text{ N/A}$ nominal voltage: 48 V DC
X-LiMMS	Philips	?	amplification: $\kappa_F^M = 74.4 \text{ N/A}$ voltage: 300 V DC
LiMMS power supply	?	?	output: 300 V DC
Signal processor	dSpace	DS1103	-

^aNot used in experiments due to noise problems.

Chapter 3

Operation

The H-bridge operation is explained using the experiment called `CleanTemplate` in the folder “Template”. This template can be used as a basis to construct a user defined experiment.

It is assumed the reader has some basic knowledge on dSpace and the software that its real-time control interface “Control Desk”. The main parts of the example experiment `CleanTemplate` are the Simulink file containing the control layout (`.mdl`), the variable file that contains the data connections (`.sdf`) which is automatically generated by Simulink when one builds the model, the real-time instrument panel layout (`.lay`) which contains the layout of the instruments (and their data connections) that are used in “Control Desk” and finally, an experiment file (`.cdx`) that is nothing more than the overall file referring to the data connections and the instrument layout files.

*** AT THE MOMENT THIS MANUAL IS PRINTED THERE SEEMS TO BE SOMETHING WRONG WITH THE MODEL `CleanTemplate`. THE H-BRIDGE INITIALIZATION DOES NOT WORK CORRECTLY AND MAKES THE H-BRIDGE Y-AXES GO CRAZY DURING THE H-BRIDGE INITIALIZATION PROCEDURE. THE PROBLEM HAS NOT YET BEEN IDENTIFIED. DUE TO CIRCUMSTANCES THE AUTHOR IS NOT IN THE OPPORTUNITY TO FIND AND FIX THE BUG. THE EXPLANATION BELOW, HOWEVER, IS STILL VALID.***

Note that if one alters for example the Simulink file name, the variable file that will be built will bear the new name. Since this is not the same name as the old file name, the accompanying layout files (`.lay`) will be useless and need to be reconnected, i.e. all the instruments in the instrument panel need to be manually (re) connected to the variables from the variable file. There is probably some neat trick to prevent the data loss when altering the Simulink file name, but a less sophisticated method of preventing any re-naming problems is the following. If you want to save a newer version of your controller, do not change the Simulink file name. Instead copy te set of “necessary” files to another folder, with another folder name. This set of necessary files are the folowing: `.mdl`, `.lay`, `.cdx`, `.cdc` and `.cdd`. The result is that the different versions of your controller are in different folders, but all have the same name. Therefore one should always be aware that one is in the correct Matlab working directory when building!

3.1 General Control Layout

Figure 3.1 depicts the general Simulink Control layout of the file `CleanTemplate.mdl`. As one can see it consists of five components, which are depicted in the subsequent Figures 3.2 to 3.6. These components are the “HDrive hardware”, the “Pendulum Hardware”, “Pendulum Initialize”, “Controller H-Bridge General” and “Pendulum Controller General”. The corresponding Control Desk experiment `CleanTemplate.cdx` contains a set of instruments to operate the main features

of the this model.

In general this control layout works as follows. The block "HDrive hardware" can be used as a plug and play component. It contains all the connections to the H-bridge I/O and hence to the H-bridge hardware. Just give in a requested current vector (In_I) and this current will be fed to the X, Y1 and Y2 linear motors. As one can see in Figure 3.1 its outputs are the H-bridge position (X, Y1, Y2) in meters (note the coordinate definition as discussed in Section 2.1), the H-bridge speed (X, Y1, Y2) in meters per second, time, the system state, a "ready" signal and an emergency signal. Inside the "HDrive hardware" block (Figure 3.2) you can see two subsystems called "subsystem" and "3-axis system". The latter contains the connections to the dSpace I/O and some hardware settings. It outputs the signals from the End Of Stroke sensors, End Positions sensors, the X-beam tilt sensors and the Emergency stop (from the emergency buttons). The other subsystem called "subsystem" can be seen as a filter: before the required current In_I is actually sent to the H-bridge it is run through the S-function `HD_V4_HDrive` (see Figure 3.8) which, during normal operation, does not do anything, except when some safety limits are exceeded. (To monitor the H-bridge and to determine whether or not safety limits are exceeded, this S-function needs the signals from the EOS sensors, the EPD sensors etc.) The C-code that is behind the S-function is in the folder `C:\HDrive`¹ A detailed description of what this C-code does can be found in [8]. What it comes down to is that this S-function (and the underlying C-code) is responsible for two things. Firstly it accommodates a safety layer. If the H-bridge speed exceeds some limits it will cut the current feed, if the alignment of the Y1 and Y2 axis is not within certain limits, this will be corrected, if a sled is heading for its end stops it will be slowed down (virtual airbag) etc. etc. Secondly it encompasses the complete H-bridge initialization procedure. When one sets "In_Start" high, the H-bridge will start aligning completely autonomously and will give a high signal on "Out_Ctren" when the initialization has been finished successfully. A very convenient way to monitor what is going on is to display the output "Out_state". This output (integer) gives the current H-bridge state. If the system state is 0 everything is OK, if the system state is something else, there is something else going on. See Appendix A for an overview of all system states. See [8] for all the details concerning the S-function.

The block "Pendulum Hardware" (see Figure 3.3) encompasses the hardware connections for the pendulum. The first input ("set encoder position") sets the pendulum encoder to the right position after initialization. The second input, "Emergency stop" has its origin in the Hbridge hardware block: if something goes wrong (system state is not equal to zero) the emergency signal is set high and, as a result the current that is fed to the pendulum amplifier is set to zero by the "product" blocks. The third input is the desired pendulum motor current. This current signal is correctly scaled using the input and output range of the current amplifier that is in use and then directly fed to the current amplifier via the dSpace I/O. The encoder is read from the I/O board (port Inc 4) and then correctly scaled to obtain the pendulum angle in radians. As one can see the pendulum velocity is determined from the pendulum position by numerical differentiation followed by a 5 point average. Not the neatest method, but good enough for the study for which the pendulum was intended [5]. Also the pendulum speed and position is calculated in degrees. This is only to be able to monitor the real time signal in degrees (which is a lot more convenient than in radians).

In the block "Pendulum Initialize", Figure 3.4, some motion tools are implemented in order to perform the pendulum initialization procedure. The pendulum initialization procedure is required in order to correctly initialize the pendulum angle and correctly set the encoder zero position. The whole procedure encompasses different movements of the H-bridge and pendulum and is similar to the H-bridge initialization procedure. First the H-bridge is moved to the pendulum

¹Do NOT change this C-code on the C drive! Since all models of all users refer to this same set of C-code, other users will get in trouble when the general C-code on the C-drive is altered. If you do want to make changes to the C-code, copy all the files in the folder `C:\HDrive` to the folder that accommodates your simulink model and make changes to your own set. When building your model, Simulink will first use the C-code in the same folder as the model (if there is any).

initialization position (which currently is at (0, 0.9)). The trajectory for this motion is generated by the block “Ref Generator H-Bridge” in the upper right corner, which is a second order trajectory generator. When the pendulum initialization position is reached, this position is actively held and the “Init_Pend” signal is set high in order to enable the second step of the initialization procedure, the pendulum home search, which is performed by the block “Pend home search”. This block just puts a -100 mA current to the pendulum motor, which makes the pendulum turn left slowly. The pendulum will then hit the end stop (aluminium plate). In this position the pendulum angle is known (-4324 encoder lines). After the velocity has dropped below a certain threshold (due to the collision with its end-stop), the “encoder set” signal is set high, which makes the encoder reading is set to the right value (the setting of the encoder position is done in the block “Pendulum Hardware”) and the pendulum angle is calibrated at -4342 lines. After the encoder has been set the third part of the initialization procedure is enabled. In this third part both the pendulum and the H-bridge are moved to a user defined home position (“X_home” and “Y_home” in the upper left corner). For the H-bridge this is achieved using the same trajectory generator as the one that moved the H-bridge to the pendulum initialization position. Another trajectory generator “Ref Generator Pendulum” (lower left corner) is used to move the pendulum to its zero position. When both the H-bridge and the pendulum are in position the “Ready Signal” is set high.

One needs to note how the correct pendulum and H-bridge feed are selected. This is done in the overall control layout, see Figure 3.1. Before any initialization has taken place the “H-bridge feed selector” and the “Pendulum Feed Selector” pass inputs one, which are the outputs of the user defined controllers in the blocks “Controller H-bridge General” and “Pendulum controller General”. When the pendulum initialization procedure is started (by setting “Start Pendulum Align” high), however, the feed selectors both pass input two, which is the signal from the pendulum initialization procedure. After this procedure has finished, the “Ready” signal is set high and both feed selectors both return to the state in which they pass input one. Also the user defined pendulum and H-bridge controllers are enabled by the ready signal.

The block “Pendulum controller General” encompasses the user defined controller to control the H-bridge. This can be anything. Currently the cogging and friction compensation are implemented for the X-axis (see section 3.1.3 for more detail). Also a virtual spring is implemented for the X-sled, while the Y axis are both kept at standstill. The X-sled can then be fed with a small force peak from the “hammer” function. In this way one can visually judge the effects of the friction and the cogging compensation, which is implemented via look-up table in the block “H Friction + Cogging Comp”. In this block also the friction compensation can be adjusted by tuning of the reduced- and dead zones in the X-sled friction compensation (see [5] for details).

The block “Pendulum controller General” encompasses the user defined controllers to control the pendulum. This can be anything. Currently a manual current feed and a constant velocity mode are present in the block. For the latter one gives in the pendulum stop position (deg) and a reference speed (deg/s). A (un-tuned) PID controller then makes the pendulum rotate at the desired velocity to the prescribed pendulum stop position. This is a remain of the constant velocity experiments that have been performed within the scope of the study of [5].

3.1.1 Improvements

If one is planning to use the current template as the basis of a new set of experiments it is advisable to implement the following improvements that will make things more neat and contribute to a decrease of the required computation time (see also Section 3.3.2).

Firstly it would be desirable to implement the pendulum initialization procedure in an S-function, just like the H-Bridge initialization procedure. This, however, is something that still has to be done. Nevertheless, the current solution works fine, but costs a lot of computation time.

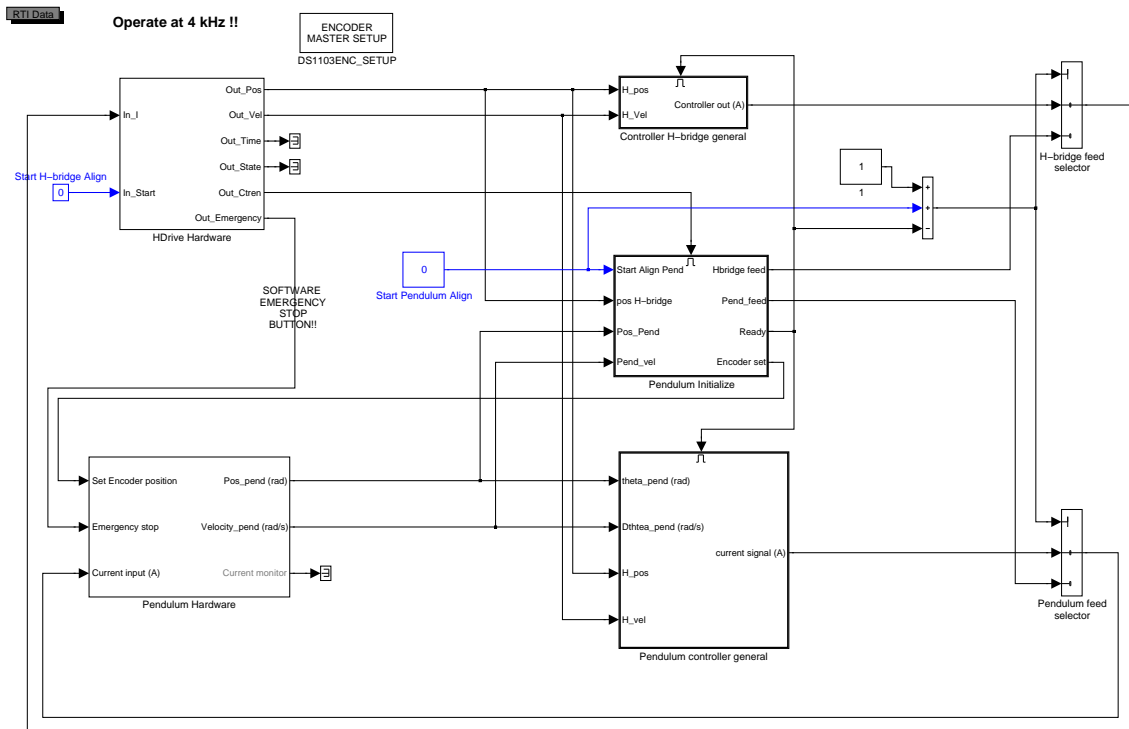


Figure 3.1: Simulink overall control layout from the file CleanTemplate.mdl

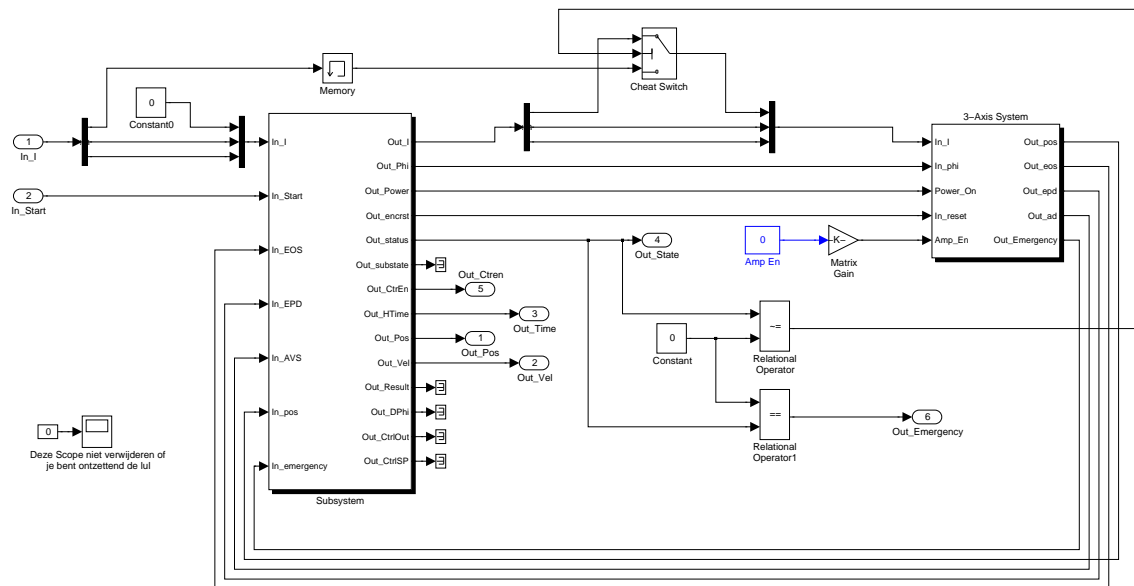


Figure 3.2: Layout of block “HDrive Hardware”

Secondly, in stead of the “time generators” that are used to generate the reference trajectory of the H-bridge and pendulum during initialization one can use the already existing ”time” signal that is available as the output of the block “H-bridge hardware”.

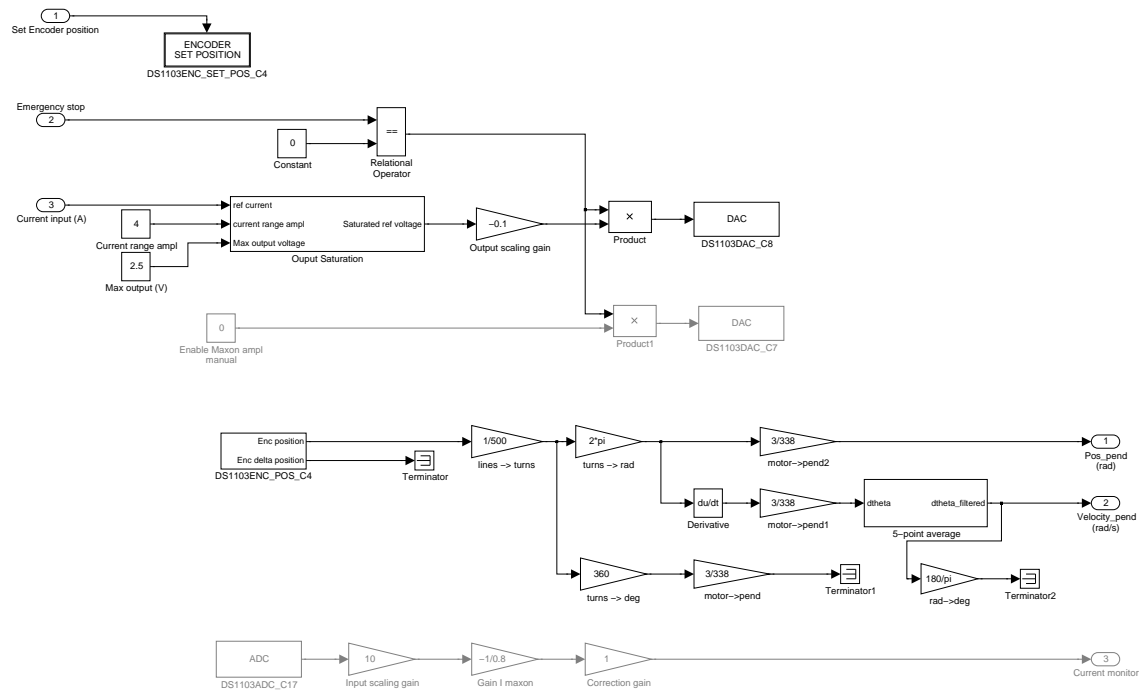


Figure 3.3: Layout of block “Pendulum Hardware”

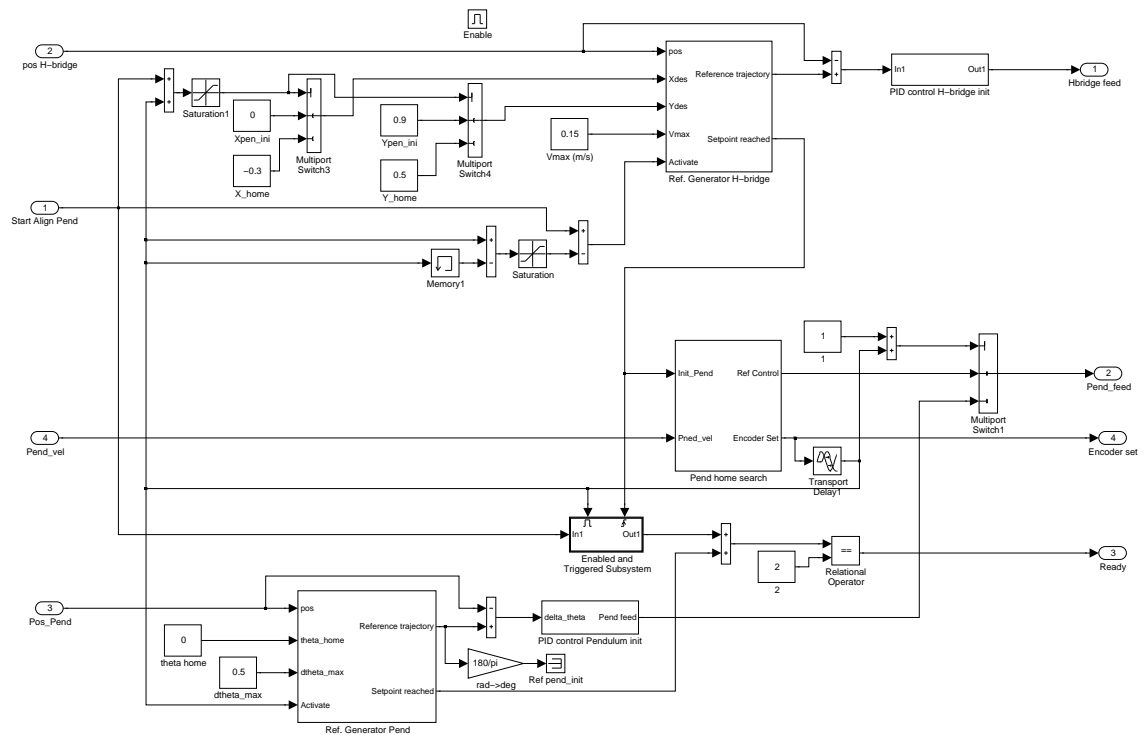


Figure 3.4: Layout of block “Pendulum Initialize”

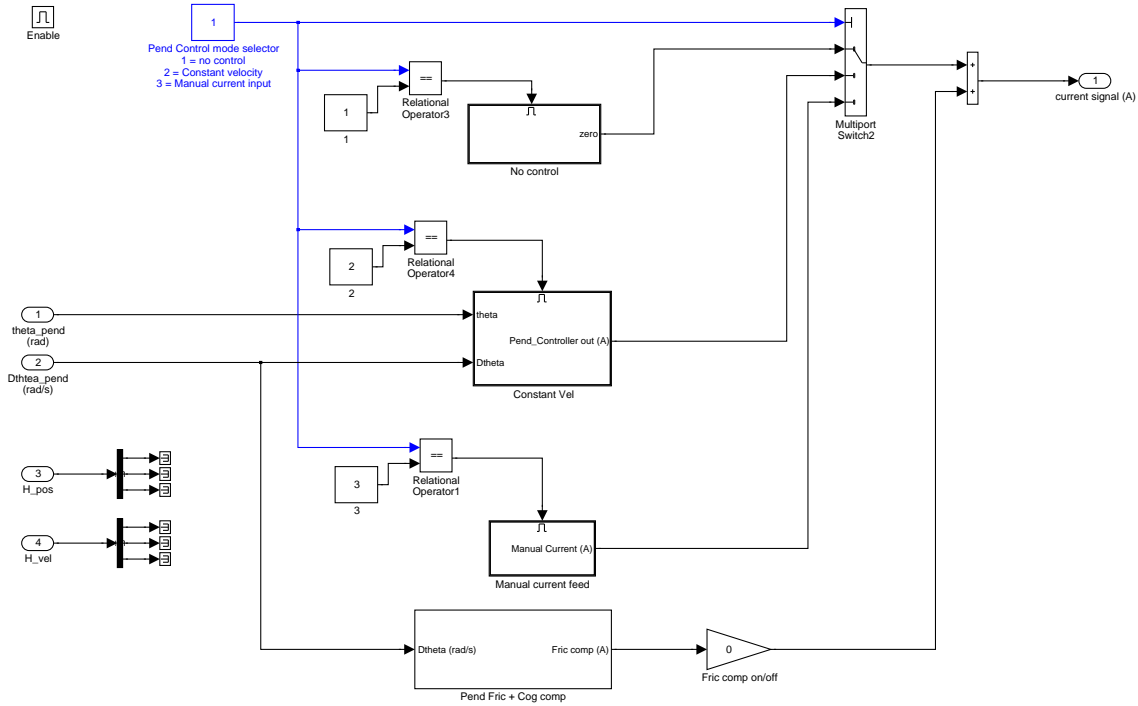


Figure 3.6: Layout of block “Pendulum Controller General”

3.1.3 Friction and Cogging Compensation

The H-bridge X and Y sleds both suffer from cogging and friction (see for example [4] and [5] for details). These phenomena have been identified within the scope of several studies on the H-bridge. The resulting data, however, has been lost because of the file removal in 2004 (see Section 1). Within the scope of the latest study on the H-bridge, the cogging and friction forces in the H-bridge X-sled have been re-identified (see [10] and [5]). Also the friction in the pendulum has been identified. (see [5]). The resulting data is available in three different `.mat` files.

- `Cogdata new.mat` contains two columns. The first column contains the X-sled position in meters, the second column contains the current (in Amp) that corresponds to the cogging force at the corresponding position.
- `Stribecknew1.mat` contains two columns. The first column contains the X-sled velocity in meters per second, the second column contains the current (in Amp) that corresponds to the friction force at the corresponding X-sled speed.
- `PendStribeck radA1.mat` contains two columns. The first column contains the pendulum velocity in meters per second, the second column contains the current (in Amp) that corresponds to the friction torque at the corresponding pendulum speed.

The above `.mat` files can be used in a look-up table to compensate for the cogging and friction phenomena. The `.mat` files can be found in the directory `Template`.

Note that the X-sled identification procedure has been performed after the H-bridge C-code offset bug had been worked around with a bypass (see Section 3.3.1 for details).

3.1.4 Initialization Procedures

H-Bridge

As discussed in the Section 3.1 the H-bridge initialization procedure is performed by the S-function `HD_V4_HDrive` and is enable setting the “Start H-Bridge” (in the model root) high. The procedure the H-bridge goes through once the initialization procedure is enabled is the following. First the offset of the coil commutation of the liner motors needs to be correctly set (see [8] and [3] for details). In order to do so the H-bridge is actuated by a sine sweep of increasing frequency, which makes a really unpleasant noise. After the commutation for the three motors has been correctly set, the X and Y axis are moved to the zero position. At this position the encoders are reset and te Y1 and Y2 motors are aligned. Finally the system moves to the starting position, which is at (-0.3, 0.5).

It should be noted that the sine sweep procedure in order to determine the correct coil commutation is not always successful. In that case the H-bridge initialization procedure is aborted. Sometimes it works to just restart the initialization procedure. If that does not work, one manually needs to move the H-bridge X, Y1 and Y2 sleds to a slightly different position and try again.

Pendulum

The pendulum initialization procedure has been discussed in Section 3.1 and does not need further attention.

3.2 Getting Started

To get started on the H-bridge:

1. Read this manual.
2. Find the folder `Template` in the directory `users` on the `D:` drive on the H-bridge computer and open the Simulink file `CleanTemplate.mdl`
3. Create your H-bridge and pendulum controllers, as desired, in the blocks “Controller H-bridge General” and “Pendulum controller General”
4. Save your model (take into account the remarks in the introduction of this chapter!).
5. Build your Simulink model.
6. Open Control Desk and the experiment called `CleanTemplate.cdx`, which also is in the folder `Template`
7. Make sure you connect the necessary instruments but also the instruments needed to operate the user defined controller. The necessary instruments are the following:
 - The H-bridge system state (connect to a display). The signal is available as an output of the the block “HDrive Hardware”
 - The turnaround time (also see Section 3.3.2). The signal, that can be found in the “folder” `\Task Info\Timer Task 1`, can be connected to a display.
 - The H-bridge amplifier enable signal (connect to a radio button; 0 = off, 1 = on). The signal can be found *in* the block “HDrive Hardware”
 - The “Start H-Bridge Align”. Can be found in the model root (connect to a radio button; 0 = off, 1 = on).
 - The “Start Pendulum Align”. Can be found in the model root (connect to a radio button; 0 = off, 1 = on).

8. Switch on the dSpace system (switch in the upper left corner of the the back of the dSpace box)
9. If the dSpace system is switched on after Control Desk has been started it is necessary to initialize the communication between the host and target PC. To do this, select the following option in Control Desk: **Platform -> Initialize -> refresh platform connection**. If Control Desk is started after the dSpace system is switched on, there is no need to re-initialize the connection between host and target PC.
10. Manually turn the pendulum in an angle that is between -10 and + 190 degrees in order to prevent collisions of the pendulum with the end-stops during initialization of the H-bridge.
11. Switch on the main power supply for the whole H-bridge setup.
12. Switch on the H-Bridge amplifier (first turn switch, then turn key).
13. Switch on the analogue amplifier (first the main amplifier power, then the switch to “lock”).
14. In Control Desk, switch to “animation” mode. Now the whole setup can be controlled from the instrument panel in Control Desk:
 - (a) Check wether system status is 8 (=ready)
 - (b) Enable H-Bridge amplifier (a quite loud high beep signal should be hearable).
 - (c) Start H-Bridge Align (see Section 3.1.4 for details)
 - (d) Start Pendulum Align (see Section 3.1 for details)
15. If both the H-bridge and Pendulum initialization have been finished successfully the system state should be 0 and the user defined controllers can be switched on.

3.3 Known Problems

3.3.1 Offset in C-code

In the latest study on the H-drive, [5] it has been noticed that there is a problem with the H-bridge current feed. As described in the previous section, there is a piece of C-code that forms the S-function `HD_V4_HDrive` (see Figure 3.8) that completely handles the H-bridge initialization and accommodates a safety layer. In normal operation, the H-bridge system status is 0. In this status there should be no safety measures active and the current that is requested from the H-bridge (`In_I` in Figure 3.8) should be equal to the current actually fed to the H-bridge (`Out_I` in Figure 3.8). It is observed however, that this is not the case!

In constant speed experiments of the X-sled (which were required to identify friction levels) one observes a speed dependent offset (`In_I-Out_I`). Figure 3.7 shows the experimental results for the X-sled. It should be emphasized that Figure 3.7 presents the difference between the S-function current input and current output during normal operation in which the system state is zero!

The cause of the above problem has never been found, the clues we have are all contradictory. Some observations that have been made in the search for the cause of the problem:

- In the studies performed on the H-bridge before the soft- and hardware update (see Chapter 1) no one has ever mentioned such a problem. This suggests that the problem somehow depends on the soft and hardware configuration. The C-code that is involved however, has not been changed during the upgrade.
- The offset seems to be perfectly linear dependant on the H-bridge speed (if one considers distinct speeds), which suggests there is some D-action active in the S-function

- If the observation is correct, the question is whether this D-action has been implemented deliberately to compensate for some H-bridge hardware effect that are unknown to us.
- If one prescribes a “random” current signal, the offset created by the S-function does not comply with the linear speed dependency as suggested by Figure 3.7
- The whole offset phenomenon is not present in the H-bridge setup at the faculty of Electrical Engineering (see Chapter 1).

The only way to find out what is going on is to go through the C-code step by step. Due to lack of knowledge of C-code of the author of this manual and [5], this did not lead to any conclusions.

In order to correctly identify the friction in the X-sled within the scope of the study of [5], the problem has been worked around with a cheap hack, as one can see in Figure 3.2. If the machine status is 0, the S-function is bypassed and the requested current is directly fed to the H-bridge amplifiers. However, if the H-bridge goes into “safety mode” (system status unequal to zero), the current is fed via the S-function and the S-function can perform its safety measures.

Note that only the H-bridge X-axis has been studied, and also, the bypass has only been implemented for the X-axis.

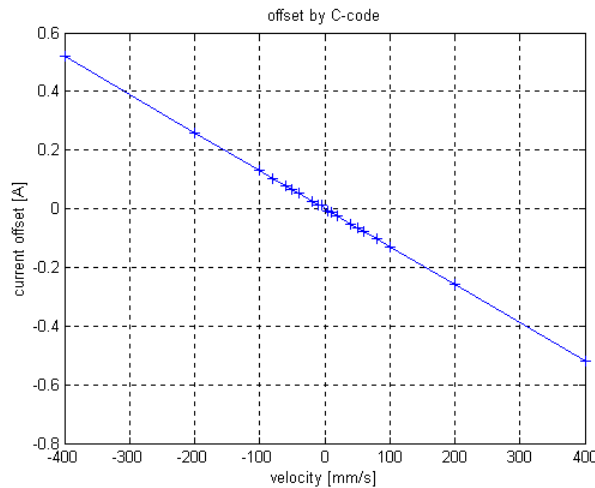


Figure 3.7: Offset due to some unknown phenomena in Ccode the S-function HD.V4.HDrive.

3.3.2 Computation time

Since we are dealing with a real-time application, computation time is an issue of concern. The standard sample frequency that has been used in most previous studies on the H-bridge is 4000 Hz. This means there is 0.25 ms available for each iteration. If an iteration takes more than 0.25 ms, the whole system gets an overflow and the process running on the dSpace system just gets stuck which results in an H-bridge that goes crazy. In this case one does need the hardware emergency button (see Section 3.1.2).

By displaying the “turnaround time” on a display in Control Desk one can monitor the required

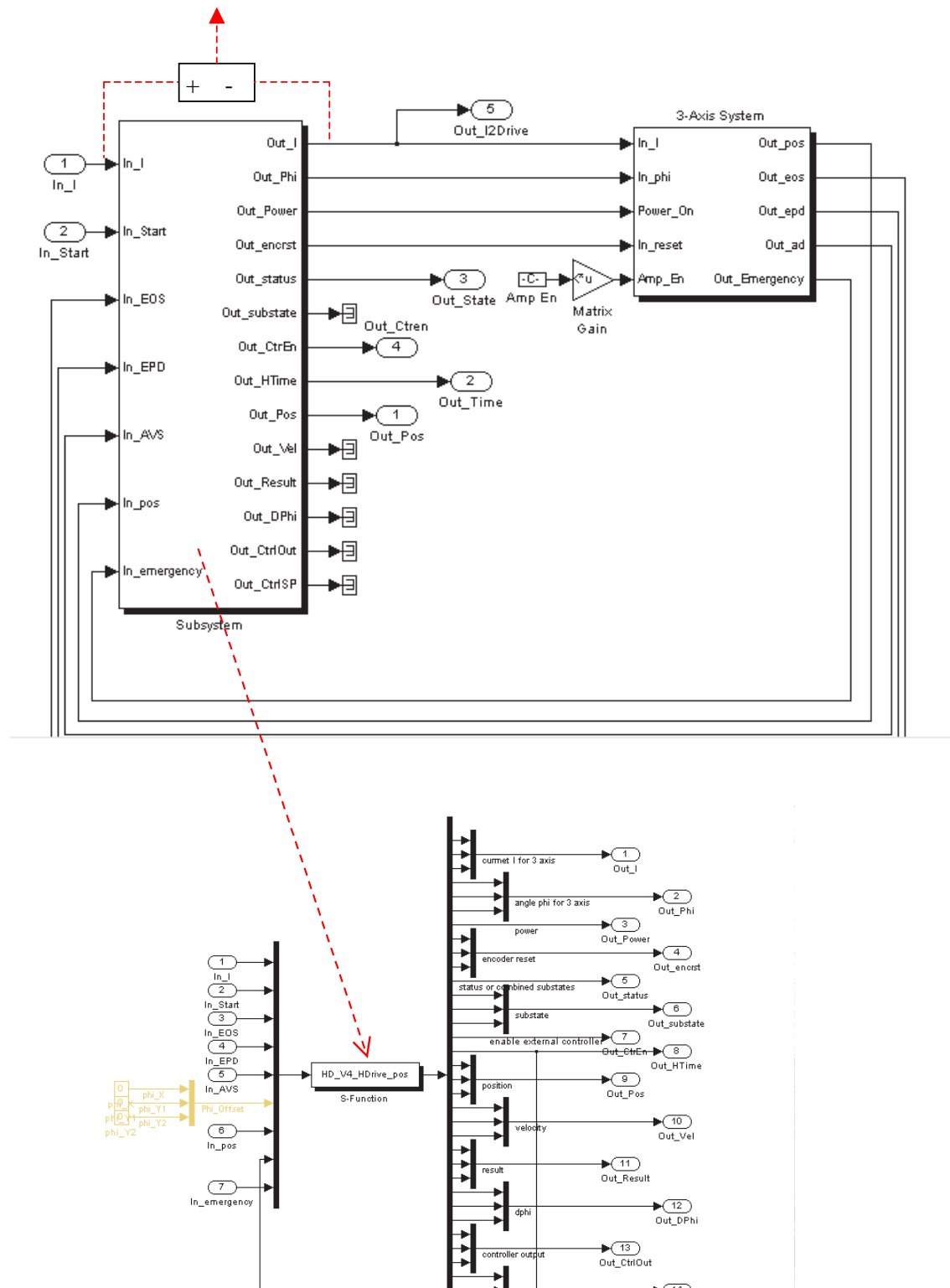


Figure 3.8: The S-function HD.V4.HDrive and its implementation.

time to complete one iteration.

In previous studies (for example [5]) one has encountered the problem that for some models the computation time per iteration is more than 0.25 ms. Sofar this has been solved by cutting down the models to the bone and to keep only the most vital parts in.

Appendix A

System States

state	meaning
0	ready
2	end of stroke
3	position violation
4	velocity violation
5	current violation
6	angle violation
7	aligning failed
8	waiting for start
11	test
12	zero search
13	Y align
14	homing
15	moving
17	emergency stop

Bibliography

- [1] Aneke, N.P.I. *Control of underactuated Mechanical Systems*. PhD thesis, Technical University of Eindhoven, 2003.
- [2] Bukkems, B. Friction induced limit cycling; an experimental case study. Technical report, Eindhoven Technical University, Eindhoven, 2001. Internal DCT Report 2001-35.
- [3] Hendriks, S.G.M. Iterative learning control on the H-Drive. Master's thesis, Eindhoven Technical University, 2002.37.
- [4] Hensen, R. *Controlled Mechanical Systems with Friction*. PhD thesis, Eindhoven Technical University, Eindhoven, 2002.
- [5] Janssen, B.R.A. Output regulation for a nonlinear mechanical system: From design to experiments. Master's thesis, Eindhoven Technical University, 2005.41.
- [6] Janssens, J. Learning control on the h-drive. Master's thesis, Eindhoven Technical University, 2000.26.
- [7] Merry, R. Experimental modal analysis of the h-drive. Master's thesis, Eindhoven Technical University, 2000.
- [8] Rovers, A.F. Controlling the H-Drive. Master's thesis, Eindhoven Technical University, 2002.12.
- [9] Steen, Rene van der. Observer design for systems in second-order chained form. Master's thesis, Eindhoven Technical University, 2004.68.
- [10] Talukder, Y. Identification of cogging and friction in the H-bridge X-axis. Technical report, Eindhoven Technical University, Dynamics and Control Technology Group, 2004.
- [11] unknown. ACM micro service manual. Technical report, Philips Semiconductors. Volume 2/3.
- [12] Verweij, A.H. Control of a permanent magnet linear motor with dSpace and matlab/simulink. Master's thesis, Eindhoven Technical University, 2000.
- [13] Voort, A-J. van de. LPV control based on a pick and place unit. Master's thesis, Eindhoven Technical University, 2002.31.
- [14] Wouters, R.M.T. Input output linearization on the h-drive including tilt of the beam. Master's thesis, Eindhoven Technical University, 2004.37.