High torque, high impact braking system by using Hot Melt Adhesive (HMA)

Juanjo Fernandez Bocanegra

Keith Gunura
Supervisor
Prof. Dr. Fumiya Iida
Bio-inspired Robotics Laboratory (BIRL)
Swiss Federal Institute of Technology Zurich (ETH)

October 19, 2011
“Take things as they come. But try to make things come as you would like to take them. Das Denken ist zwar allen Menschen erlaubt, aber vielen bleibt es erspart.”

- Curts Goetz
Preface

I would like to gratefully thank my supervisor, Prof. Fumiya Iida, for his enthusiasm, for his outstanding support and providing me with a unique environment that laid the foundation for the great work I have accomplished. I would also like to thank Keith Gunura for being such a great teammate and for all the interesting discussions we have had. Finally, I would like to express my gratitude to all members of the Bio-inspired Robotics Laboratory for the collaboration as well as the great time we have spent together.

As my last words of gratitude, I would like to thank my local University, The Universitat Politècnica de Catalunya. BarcelonaTech (UPC), for the possibility to carry out my Master’s Thesis in a totally different educational environment, such is the ETH Zürich.
Abstract

When designing a new legged robot it is important to think about what the robot has to do. It is not the same a non-stop walking robot than a legged robot which has to be at the same position much longer than moving.

For the second kind of robot, it could be a great improvement to add a break in each actuated joint, so it is not necessary to wasted energy during the stopped time. And not only that, thanks to the brake, it is possible to achieve heavier loads using a specific walking pattern in which you always place the center of mass on the three legs that are not moving in that moment (with the brake activated), while make the next step with the fourth leg.

The most common braking systems are mechanic, whether using an electromagnetic brake or some other kind of friction-based brake in addition to any sort of actuator, e.g., hydraulic, electric, . . . Even though they have been deeply studied, there are some disadvantages by using mechanic braking systems, i.e., the high prices and the bad holding torque/weight relation, important aspect for a legged robot.

This thesis presents a conceptual idea for a braking system based on the thermodynamic control of Hot Melt Adhesive (HMA), as well as the basic characteristics of this material and a long leg prototype which shows the feasibility of this concept.
Contents

Abstract

List of Tables

List of Figures

1 Introduction
  1.1 State of the Art
  1.2 Design Concept
  1.3 Review of Thesis Content

2 Modeling and Analysis of Adhesion
  2.1 Properties and Definitions
  2.2 HMA Thermodynamics
    2.2.1 Heating Process
    2.2.2 Cooling Process

3 Leg Design and Control
  3.1 Mechanical Design
    3.1.1 General Strategy
    3.1.2 Overview
    3.1.3 Robot platform
    3.1.4 Connection bars
    3.1.5 Joints
    3.1.6 Braking system
  3.2 Actuation
    3.2.1 Electromechanic actuation
    3.2.2 Thermodynamic actuation
  3.3 Sensing
    3.3.1 Angle Sensor
    3.3.2 Current sensor
    3.3.3 Temperature sensor
  3.4 Control
    3.4.1 Upper joint motor control
Contents

3.4.2 Lower joint motor control ........................................ 31
3.4.3 Heating System Control .......................................... 34
3.4.4 Cooling System Control ......................................... 34
3.4.5 High-Level Control ............................................... 35
3.4.6 Autonomous Mode ............................................... 37

4 Experiments .......................................................... 39
4.1 Heating Performance ............................................... 39
4.2 Cooling Performance ............................................... 40
4.3 Torque vs Bending Angle ......................................... 42
4.4 Torque vs Temperature ........................................... 43
4.5 Walking Speed Estimation ........................................ 44

5 Conclusion ............................................................ 46
5.1 Future Work ......................................................... 47

References ............................................................. 49

A Appendix ............................................................... 50
A.1 Matlab Code ........................................................ 50
A.1.1 Matlab InitRobot.m .............................................. 50
A.1.2 Matlab SerialWrite.m .......................................... 51
A.1.3 Matlab SerialWriteServo.m .................................... 51
A.1.4 Matlab Heating.m ............................................... 51
A.1.5 Matlab Cooling.m ............................................... 51
A.2 Microcontroller Code .............................................. 52
A.2.1 Robot Main Code (LegControl.cpp) ......................... 52
A.2.2 Robot Code for Automatic Mode (AutomaticLegControl.cpp) 61
A.3 Data sheets .......................................................... 67
List of Tables

1  Definitions of used symbols and notation.  . . . . . . . . . . . . . . 7
2  List of electromechanic actuators used.  . . . . . . . . . . . . . . . 22
3  List of the different methods for measuring electrical current.  . . 26
4  List of available commands with associated parameters.  . . . . . . 37
5  List of the different braking systems in the market.  . . . . . . . . 46
List of Figures

1 Investigated designs for the braking system. . . . . . . . . . . . 4
2 Temperature profile of typical HMA. . . . . . . . . . . . . . . 7
3 Heating process for different type of heaters and configurations. . . 9
4 Effect of different configurations on cooling process. . . . . 10
5 Illustration of the general mechanical design in the 3D model. . 14
6 Simulated step of the leg. . . . . . . . . . . . . . . . . . . . . . 15
7 3D design model of the robot platform. . . . . . . . . . . . . . 16
8 Joint design. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
9 Inside part of the brake. . . . . . . . . . . . . . . . . . . . . . . 19
10 Inside part of the brake. . . . . . . . . . . . . . . . . . . . . . . 21
11 Electromechanic actuators. . . . . . . . . . . . . . . . . . . . . 22
12 Illustration of the Thermodynamic actuators. . . . . . . . . . . 24
13 Mounted angle sensor in the lower joint. . . . . . . . . . . . . . 25
14 Power resistor used as a current sensor. . . . . . . . . . . . . . 28
15 Temperature sensor used in the project. . . . . . . . . . . . . . 28
16 Standard servomotor control. . . . . . . . . . . . . . . . . . . . 30
17 DC Motor control board. . . . . . . . . . . . . . . . . . . . . . . 32
18 PWM Wave. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 33
19 Electrical circuit of the control for the heating and cooling systems. 35
20 Electrical circuit diagram of the control of the whole robot. . . 36
21 Performance of the heating process with different input power. . 39
22 Estimation of the best performance of the heating process. . . 40
23 Performance of the cooling process. . . . . . . . . . . . . . . . 41
24 Torque vs bending angle graphic. . . . . . . . . . . . . . . . . . 42
25 Torque vs temperature graphic. . . . . . . . . . . . . . . . . . 43
26 Steps simulation of the lower joint. . . . . . . . . . . . . . . . . 44
27 Steps simulation of the upper joint. . . . . . . . . . . . . . . . . 45
28 Schematic of the Arduino Duemilanove microcontroller board. . 68
1 Introduction

Nowadays, robotic research is clearly focused in legged locomotion. Whether with biped robots, four-legged robot or any other number of limbs.

This kind of locomotion is far more difficult and inefficient than the usual wheeled robots. Nevertheless it has some important advantages, e.g., the biped robots are usually People-Friendly, due to its vertical position and its ability to interact with made-for-human tools or environments. This feature allows them to stand, move and work in closed spaces with people, without danger for any of them and being an assistance for the humans. Another important advantage, of course, is that the legged robots are able to walk on irregular terrains (forest, sand,...) and overcome unexpected obstacles (stairs, pavements,...), which enables them to execute a wide variety of tasks.

1.1 State of the Art

Although the legged robots have some serious disadvantage, as it was mentioned previously, they will be certainly necessary in the robotics future, where we will need robots able to use the made-for-human tools and to move around human environments.

These kind of robots use actuated joints for moving the different limbs. There are several type of actuators suitable for this joints, e.g., pneumatic, hydraulic or electric. Of course, in each family of actuator there are different ways to solve the problem. There are, for example, pneumatic muscles ([1]) and the typical pneumatic pistons, linear electric actuators ([2]), geared down motors, high torque motors, hydraulic pistons, etc.

As is for all known, the main task of the biped robots will be helping human to carry out theirs daily tasks and to participate in the dangerous activities than human will be no longer doing. At the same time, not only biped robots will be used, also robots with a different number of limbs will be useful for different tasks, i.e., rescue robots, anti-bomb robots or carrier robots, etc.

Although all this activities seem to be dynamic, it is possible to find some situations where the relation between static and dynamic activity is much closer to the static state. Imagine a surveillance robot, walking through the forest until it reaches its vigilance position where it stops and stays still for some hours or
days. During this static period it is important to reduce the power consumption for the static locomotion in order to reduce the battery dependence. If the legged robot only has normal actuators in the joints, it will be necessary to provide energy nonstop to keep the position. However, if the robot has brakes in the joints, besides the actuators, it will be able to be at the same position as long as required, no worry about the battery.

Of course we have to talk about which kind of brake we need for this application. There are different sort of brakes in the market as shown below:

1. **Electromagnetic brake:** They use an electromagnetic field as an actuation force but the braking force is the mechanical friction [3]. They are usually heavy, with a bad relation torque to weight.
   - Simple actuation electromagnetic brake.
     - These brakes use the electromagnetic force to press a plate friction surface. They engage only when energy is supplied, so it would be necessary energy during the static phase as well, for this reason they are not suitable.
   - Multiple disk electromagnetic brake.
     - These brakes use several friction disks in order to increase the torque to size ratio.
   - Power off electromagnetic brake.
     - These brakes use springs or permanent magnets to supply the braking force, they do not need any energy in the braked state, they only need it for the disengagement.

2. **Hydraulic or pneumatic brake:**
   - Secondary system for keeping the pressure of the fluid. This issue add complexity to the design, as well as weight.
   - Normally used for big sizes systems.
   - They require energy while brake, to keep the secondary system working, this point make it not suitable for our design.
As we can see, we need a brake with a good torque to weight ratio and with no energy consumption during the static state. Any of the previous brake systems have neither of these features, for this reason we need to think about a new concept of braking system, here is where the Hot Melt Adhesive brake appears.

### 1.2 Design Concept

This work introduces a novel approach to achieve the braking function of the joints based on chemical adhesion and phase transition. The use of so called Hot Melt Adhesives (HMA) is proposed. This material has a number of interesting properties: it can be repeatedly transformed between fluid and solid states by controlling temperature; it is highly viscous at high temperature, while it exhibits solid state, with a relatively high breaking strength, at room temperature [4]. The use of this material in robotic systems have been only partially explored in the past (e.g. a robotic gripper based on HMA; [5]) and it has not been used for the purpose of the braking system in a legged robot.

The main challenge addressed in this thesis is, therefore, to propose basic principles of using HMA for braking systems and to evaluate the performance of this approach. The use of HMA for braking has found to be very interesting since its holding torque, thanks to the adhesion and solid state features, seems to outperform the normal braking systems nowadays, it emerges as suitable for control and it is cheap.

Several designs have been investigated in order to adopt a suitable solution of a brake based on hot melt adhesion. At first, important criteria were mainly referred to performance such as speed, strength, and power consumption for example. Since this idea is novel and, as far as we know, it has not been accomplished before, we have decided to rather prefer a non optimal but working system, that states our idea instead of starting to develop a complex and demanding system, which may have been successful but could not have been finished within the time frame of this project. Thus, other factors such as cost, feasibility, availability of parts and complexity have become more essential.

Finally we have decided to investigate different designs which are illustrated in Fig. [1].

The first design, shown in Fig. [1] uses steel as a material for building the brake, no heaters have been added, and only torque test has been done. After
confirming that the idea was feasible, we decided to design a new device with a heating system, which gives us the opportunity to change the state of the braking material (HMA) and, therefore, run several tests about the performance of our design.

This second design uses wire resistances to heat up the sealing box, and, subsequently, heat up the Hot Melt Adhesive and detach the brake. After a few non recorded tests, the wires broke apart due to the weakness of the device. We decided to use a stronger heater, which can withstand the required high temperatures.

The third and last design uses normal resistors as heaters. This device is placed inside the sealed box, in holes with the exact size, which provide extra resistance to any mechanical breakdown.

The inside part of the brakes has been changing from each model to the next one, obtaining different results and arriving to the conclusion that the best option is an uneven surface in both parts, having small shafts which provide extra
holding torque. If this shafts are too big, the flexibility of the brake seems to be excessive, if they are too small the positive effect is not significant. It has to be agreed a point of interest, where the flexibility is not excessive but the performance of the brake is good.

1.3 Review of Thesis Content

As follows, the thesis structure and contents are outlined. First, the basic properties of HMA are discussed. Experiments on the cooling and heating of HMA are studied.

A detailed description on the design and control of the robot is covered in chapter 3. Starting from the general strategy and the mechanical design, actuation and sensing is discussed and a section on control consists everything from low-level control to high-level control including the entire control architecture. Communication between the robot and the workstation, wherefrom control commands are issued and sensory data from the robot is processed, is explained.

In chapter 4 the performance test on robotic braking is shown. Additionally simple improvements to achieve faster behaviour without adding further elements and control are discussed and the requirements for a prediction on locomotion speed expectations are provided.

Finally, the last chapter 5 closes with a summary on the proposed approach and a discussion of problems and benefits. Future prospects and ideas for future projects are listed.
2 Modeling and Analysis of Adhesion

Our braking system makes use of the unique material property of HMA, adhesion dependent on temperature, which can be controlled by heating and cooling the material. More specifically, we will control the temperature on the outside part of the sealed box which contain the HMA, and the bidirectional transitions of HMA between the solid and liquid states will be used to attach or detach the brake. In this section, we will first characterize the material properties of HMA in practice with respect to different temperature. And then, we introduce the ways we have chosen for the two main processes of our device, the heating and the cooling process.

2.1 Properties and Definitions

HMAs are solvent-free thermoplastic solid materials characterized by the three main distinctive phases dependent on its temperature. Fig. 2 illustrates a typical temperature profile of HMAs. At high temperatures ($T_A$, typically above $120^\circ$) HMAs are liquid with a relatively low viscosity. At a temperature around 70-80$^\circ$ ($T_B$) they are in a so-called plastic state bearing a higher viscosity and they are solid with high bonding strength at lower temperatures ($T_C$, typically below 40-50$^\circ$). The open-time is the maximum amount of time after heating up the adhesive that one has to wait before applying any torque to the brake and still have an acceptable quality [4].

The idea is to exploit this three states such that the brake is attached in the solid state, having a good torque still in the plastic state, and detached in the liquid state, moving the leg while cooling again and before arriving to the plastic state again. The higher the temperature of the HMA, the lower viscosity it has and the lower friction the brake has in the free state, consequently it is needed less torque in the actuator of the joint. The lower the temperature of the HMA in the braked state, the higer holding torque has the brake.

For the sake of clarity symbols and notations referred to HMA properties and thermodynamics are defined in table 2.1.
2.1 Properties and Definitions

Figure 2: Temperature profile of typical HMA.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liquid temperature</td>
<td>$T_A$</td>
<td>Temperature where the adhesive is in liquid state, typically above 120°.</td>
</tr>
<tr>
<td>Plastic temperature</td>
<td>$T_B$</td>
<td>Temperature where the adhesive is in plastic state and the bond is easily breakable. Typically around 70-80°.</td>
</tr>
<tr>
<td>Solid temperature</td>
<td>$T_C$</td>
<td>Temperature where the adhesive is solid with a high bonding strength, typically below 40-50°.</td>
</tr>
<tr>
<td>Environment</td>
<td>-</td>
<td>The brake design (e.g. the internal shape of the brake or the material) where the HMA is attached.</td>
</tr>
<tr>
<td>Holding Torque</td>
<td>$H_T$</td>
<td>Tangential component of the torque among the center of the brake.</td>
</tr>
<tr>
<td>Bonding area</td>
<td>$A$</td>
<td>Area covered with HMA between the axis and the static casing of the brake.</td>
</tr>
</tbody>
</table>

Table 1: Definitions of used symbols and notation.
2.2 HMA Thermodynamics

2.2.1 Heating Process

The heating process is crucial for the good development of the brake. As shown later, the process takes up to one minute to reach the liquid temperature \( T_A \) depending on the design of the heater and the power used. An important point that should be considered is the design of heater being used. Since we have made a point of designing cheap and simple robots there are mainly two types up for discussion: normal electrical resistances and wire resistances. As shown by Fig. 3, electrical resistances are interesting since the process can be accelerated arbitrarily by increasing the power, and the power can be increased by changing the resistance values. This property makes the electrical resistances ideal for the design of a prototype, as you can try different heating configurations with a small modification. However, wire resistances are more complicated to set up. It is necessary to roll it around the piece you want to heat up, with thermally conductive compound to improve the heat transfer, so you have to be sure about the value you need, because, once it is installed, it is a complicated and delicate job to modify it. Furthermore, wire resistances make difficult to install heat sinks for the cooling process, due to the fact that the outside part, where you install the cooling system, is occupied with the wire itself. Talking about the heat sinks, the fact of installing them makes the heating process slightly slower, since there is more material to heat up and the area which distribute the heat passively is bigger, but contributes tremendously in the cooling process.

In our prototype we have decided to use the normal electrical resistances because they are easier to install and modify. Furthermore, wire resistances have manifest to be mechanically weak, having critical short circuits during the working time, due to the high temperatures and the weakness of the insulation of the wires.

To obtain the measurements in Fig. 3 the same heating design has been used. The temperature has been measured with a temperature sensor (NTC thermistor) adhered to the outside part of the box which contain the HMA. Because of that, the graphic is out of phase. The heat has to dissipate through the whole piece of metal before arrive to the transducer. The results show how the heating response is proportional to the heating power, so in our system we will use the highest
Figure 3: Heating process for different type of heaters and configurations. The figure clearly shows that Peltier elements prevail in terms of the priorities stated above.

2.2.2 Cooling Process

In the previous section we have discussed the heating process that is needed to detach the brake. Apart from the heating process that is required for being able to move the leg, also the cooling process is involved in the attachment of the brake, so we can leave the leg braked again in the new desired position. This task can be either performed passively by simply waiting for the HMA to become solid or in an active way by using cooling devices to accelerate the cooling process.

Fig. 4 shall provide an overview of different possibilities in terms of configurations of the cooling system. We have considered only forced air cooling systems, due to the simplicity of design and installation. In a future approach a water cooling system will be proposed as a better solution, but this topic will not be mention until the last chapter 5.1. For our design, only heat sinks and fans have been used. The procedure for the tests has been as follows: once the system has arrived to the temperature of 90° the heater is stopped and start the cool-
2.2 HMA Thermodynamics

Figure 4: Effect of different configurations on cooling process. The heat sink augmented Peltier element prevails due to its capability of rapid heat transfer.

The cooling process which keep working until the temperature arrive close to the room temperature again. Two different designs have been tested, the first one consists of two big heatsinks and a large fan, and the second one of several small heat sinks and two small fans. At the same time, passive cooling has been tested as a reference showing the significant improvement of the active cooling.

Contrary to what we have said before about the heating process and the scalability of the heating time just by increasing the power applied, in the cooling system we can only achieve better performance by improving the design of the heat sinks or by adding bigger fans, which give a higher flow rate. Due to the fact that we need to keep the weight low, it is not possible to add bigger fans or more heat sinks as much as we would like, we have to arrive to the best solution in terms of performance to weight ratio. The reason of our need for a light cooling system is because all these parts will be placed on the joint, and all together will be lifted by the motor, we want to keep the design as cheap as possible and a stronger motor would increase the budget.

As expected, temperature decreases rapidly at the beginning but saturates when arriving to the room temperature again. As it turned out, the design with
the big fan and the two big heat sinks is better than the other solution, but, at the same time, is markedly heavier. Because of that, we decided to use the heavy solution for the lower joint, in order to have the best performance because in this position the weight is not a capital decision factor, as the motor does not have to lift it, it is part of the joint itself. In the upper joint, the one which has to be lifted by the upper actuator, weight is a capital factor, for this reason we decided to use the design with the two small fans and the small heat sinks because the solution is lighter but has not so bad performance in relation with the previous design.
3 Leg Design and Control

For the design of a braking system the main feature to work with is the holding torque. Several approaches have been tested in order to find the best solution in terms of torque to weight ratio and feasibility. The time factor has not been taken into account, as what we want is a functional prototype. It does not matter, at this point of the research, the features of the design but, what really matters is to achieve a working system to validate the idea.

3.1 Mechanical Design

The overall goal is to design a leg using 2 joints and 3 degrees of chapter ??.
The leg has to show the idea of the braking system together with the actuator in each of the joints, and be able to remain in any position without the help of the actuator, once the brake is activated. The braked leg has to prove that it can support not only its own weight but also extra load, to simulate the weight of the future whole robot. The moving process of the leg consist of three main functions. They are listed following:

- Heat up the HMA until liquid temperature $T_A$ to be able to move the joint using the actuator.
- Move the leg to the desired new position using the actuators and keeping the temperature above the liquid temperature $T_A$.
- Cool down the HMA until solid temperature $T_C$, use meantime the actuators to keep the position, and switch them off once the brake is attached.

This section presents a detailed description of how those functions have been implemented in the proposed platform keeping the design simple, functional and cheap.

3.1.1 General Strategy

One of the main problems caused by the use of HMA is the loss of adhesive material when it is in the liquid state. If the adhesive, in the liquid state, was exposed to the environment, it would flow down due to its low viscosity, and it would be necessary to supply extra glue. For this reason, and because the glue
supply would make the design highly complicated, it is essential to totally isolate the HMA from the external environment. To achieve this level of isolation, a sealed case was designed to contain all the HMA for the braking system inside it.

The design of the sealed box is very important, since the viscosity of the adhesive when the temperature rise up to liquid temperature $T_A$ is low enough to flow through any fissure in the design. For solve any small leakage which would may appear, a thermal glue, which withstand high temperatures, will be used as a sealer.

In order to heat up the glue to detach the brake, we have to heat up all the sealed case using a resistor net placed in the wall of the case. This method of heating is reliable since the heater and the source are separated but, at the same time, it is inefficient in terms of heating speed and power consumption. The heating source has to heat up not only the HMA but also all the sealed box, this involve a slower system and more power consumption.

To fulfill the goal of three degrees of freedom in each joint, stated in the previous section, two joints mechanism is proposed. One, so called, upper joint, simulating the human or animal knee, with one degree of freedom, and another lower joint, simulating the hip, with two dregrees of freedom. Fig. 5 illustrates the general mechanical design of the leg.

3.1.2 Overview

Fig. 6 shows the model of the leg and a simulated step. As stated in the previous section, the leg has three dregrees of freedom, two in the lower joint and one more in the upper joint Fig. 5. The design consist of five main parts: the robot platform, two joints and two connection bars. A provisional platform has been used in order to attach it with the working desk to have a fixed prototype to do the experiments. The bars, which connect both joints and connect the upper joint with the ground, are made with fiber carbon in order to be light but, at the same time, resistant. Each of the joints are different. Even though both have a similar braking system, the upper joint has only one degree of freedom, thanks to the servomotor used as an actuator, which allows a bidirectional rotation from $-90^\circ$ to $90^\circ$. And there are two degrees of freedom in the lower joint, one vertical movement, as in the upper joint, again from $-90^\circ$ to $90^\circ$, in this case developed by a DC motor, and another horitzontal movement, which rotates the leg from
The control task is accomplished by various systems. For controlling the servomotors only a digital signal from the main microcontroller board is needed, and this signal will command the internal control electronics of the servomotors. In the DC motor case, it will be necessary a motor controller, and this will be again controlled by the microcontroller board. Several sensors are used for the controlling task, and two external power supplies provides the power for the HMA heater, the HMA cooler, the servomotors and the DC motor.

### 3.1.3 Robot platform

The entire leg is placed on a platform. In the actual prototype, a wood panel is used as a support for the leg. Wood has been chosen for the material because it is easy to work with it, it is easy to cut and drill and the accuracy offered is good enough. Moreover, wood is a good electrical insulation, which facilitates the assembling of the printed circuit boards on it.

A piece of Chipboard with a thickness of 10mm was used for the manufactur-
Figure 6: Simulated step of the leg. The figure shows the four different positions required for a step. In each position, a different pattern of torques is applied.
3.1 Mechanical Design

Figure 7: 3D design model of the robot platform: distribution of the different components of the robot platform (left) and 3D model of the whole leg installed on the platform and over the working desk (right).

On top of the supporting panel, a piece of aluminium, with a dimension of 60mm x 50mm x 20mm, is attached using four screws. Here is where the lower joint will be located, that is the reason we need to use aluminium instead of the wood, for resisting the weight of the leg. The connection between the supporting panel and the lower joint will be carried out for a simple hole in the aluminium piece and a short bar on the bottom of the joint which will perfectly fit. Besides, also on the panel, all the electronics for the control are placed.

Once we have the robot platform ready, we can attach it on the working desk using two clamps. With that we will have a stable prototype where we will can develop our future experiments.

For a better understanding of the platform, the Fig. 7 shows a simple 3D design model with the distribution of the components on the wood panel and a simulation of the whole system over the working desk.
3.1.4 Connection bars

For the connection between the lower joint and the upper joint, and between the upper joint and the ground, we need a rigid bar to simulate the bones. At first step, an aluminium hollow bar was used for this task, the strength of the material was excellent, nevertheless, the weight of the bar was 236 grams, with a length of 1000mm. As we need to make the total load on the brake and on the actuator as lower as possible, we decided to change the material for this part of the robot. The current design uses fiber carbon hollow bars for this affair.

Between the lower joint and the upper joint, a 700mm bar is used. The weight is as low as 70 grams. For the section between the upper joint and the ground we need 1000mm, with a weight of 100 grams. So, we can see than we have achieved a reduction of more than 57% from the first version, just by changing the material.

We decided the length of the bars just because this way the robot will be impressive to show to the people how strong is the braking system designed by us. The ratio between both bars is not one, it is this way in order to reduce the needed torque in the lower joint, which is the one that has to lift the whole leg.

3.1.5 Joints

These are the most important parts of the leg, the two hybrid dynamic joints. The design consist in an aluminium frame which support all the parts: brake, actuator, bearings, bar connections, heater and fans. Fig. 8 shows all the different parts with which the joint is made.

The two joints have similar design, the only differences are than in the upper joint a servomotor is used as an actuator and in the lower joint a DC motor, due to the necessity of higher torque than the one of the servomotor. For the heater, we use normal resistors placed, two in series and nine in parallel, inside the holes of the sealed box. The value used for the resistors will be explained in a following section. The cooling system are just heatsinks placed outside the sealed box of the brake and one or two fans which create a air flow around the case.

For the shaft connection between the brake and the actuator we have had to design a shaft connector, using aluminium and build it by ourself. And the connection between the joints and the fiber carbon bars is accomplished by an aluminium tube, fixed at the joint and where the bar is inserted. The last 40mm
3.1 Mechanical Design

Figure 8: 3D design model of the joint. This is actually the upper joint, in the lower joint we should replace the servomotor for a DC motor controlled thanks to a potentiometer which reads the position continuously.

of the bars are inserted in this tube, which give it enough strength. To avoid the possibility than the bar goes out of this tube, adhesive tape has been used.

The use of bearings is essential because of the high torque than the joint has to work with. Two bearings are used, with a size of 8mm in the inner axis, 22mm in the outside part and 7mm of width. Each of these bearings is placed at each side of the braking system.

In the joint with the DC motor instead of the servomotor, and extra element has to be add. A potentiometer is used to read the angle of the joint, in order to control the motor. This potentiometer is connected to the shaft at the extrem of it, on the side which does not have the shaft connection with the motor.

3.1.6 Braking system

As we can see in Fig. 10, the brake itself has a simple mechanical design in the outside part. It is a cylinder with a shaft crossing along it. It also has thirteen holes around the outside part where the heaters are places. Three of these holes are reserved for three screws which will be used for closing the sealed case.

The diameter of the case in the outside part is 50mm, and 35mm in the inside part. It has been choosed the biggest possible diameter, taking into account
Figure 9: Computer-aided design model of the inside part of the brake.

the machines we have at our disposal and the difficulty of build it. The reason because we wanted the biggest possible diameter in the brake is the holding torque to shape ratio behaviour which was discussed in chapter ??, in which it was proved than increasing the diameter has a higher impact in the holding torque than increasing the length of the sealed case.

The length of the sealed box is 30mm in the outside part, and 23mm in the inside part. The length has been estimated thanks to the several previous experiments, not exhibited in this report, where it was tested the approximate best design for the inside part of the brake and the approximate size of the brake for our desired holding torque.

Two different metallic material have been used in order to achieve the best performance of the brake. For the shaft, steel has been used because of its great strength. The main shaft has to deal with high torques and forces, due to that, an 8mm steel bar has been chosen for the shaft.

There is, attached to the shaft, an aluminum piece which is used for decreasing the amount of HMA used for the brake. As we need high torque, it is necessary to increase the radius where the adhesive actuates, so we can achieve higher torque for the same volume of glue, as was discussed in chapter ???. The use of aluminum for this piece is because we need a light joint, and here, the strength is
not critical as in the shaft, so aluminum can be used, giving better weight features. For attaching both parts, two steel screws have been used. It could have been possible to build the shaft and this piece both together, but steel would have been necessary for that and the weight would have been 146 grams instead of 70 grams of the current solution, using two pieces with different materials.

Finally, the sealed case has been made with aluminum again, since no high strength is required. The use of aluminum also improve the heat transfer, which involve a faster heating and cooling process of the HMA. Thanks to the use of Aluminum instead of Steel, the weight of this part has passed from 221 grams for the steel choice to 75 grams in the aluminum side, which is a great improvement taking into account that we are working with a joint in the range of 500 grams (brake plus actuator plus fixing parts).

Between the outside case and the shaft with the aluminum part is where the Hot Melt Adhesive is confined. As we can see in Fig. 10, both parts surrounding the adhesive have a special shape. Several half-holes have been milled which gives a better performance in terms of holding torque and joint deflection when the brake is attached.

3.2 Actuation

As discussed in chapter 3.1, the robot will have 2 joints and 3 degrees of freedom, for accomplishing this, 3 different motion actuators will be needed, plus 4 thermodynamic actuators, for the process of heating and cooling each of the joints. From now on, we will distinguish between electromechanical actuation which is needed for movement and thermodynamic actuation which covers heating and cooling.

3.2.1 Electromechanic actuation

As we said, 3 actuator will be needed. The table 3.2.1 lists the actuators used showing their specifications and functionality. Following, the actuators used will be explained in detail and in Fig. 11 we can see the two type of actuators.

In the upper joint, as we named in Fig. 5, only one actuator is required. The function of this actuator is mainly lift the last part of the leg, which only consist of a fiber carbon bar, with a weight of 100 grams, placed in an average distance
3.2 Actuation

Figure 10: Computer-aided design model of the inside part of the brake.

of 500mm. This data give a torque requirement of 0.5Nm, we chose 1Nm instead, in order to not be too demanding with the actuator. A 3Nm holding torque servomotor, with metallic gears, has been used. Despite that it could seem oversized, after empirical experiments, we related the holding torque, provide by the manufacturer, with the real dynamic torque. The ratio is 1 to 3, The dynamic torque is usually around 3 times lower than the official holding torque. The choice of a servomotor instead of a DC motor is because the servomotor requires less electronics to be controlled and it is usually simpler. Despite the higher cost of the servomotor, working with less than around 5Nm holding torque, it is worth because of its simplicity and efficiency. The final choice of the servomotor has been the KRS 2350HV from Kondo, a 3Nm holding torque servomotor, with a working range of 9V to 12V.

For the lower joint, two actuators are required, one for lifting the leg and the other one for turning it. The actuator used for lifting the leg has to be stronger than the servomotor we already used for the upper joint. The estimated weight of the joint is 550 grams, placed in a distance of 700mm, plus 70 grams of the fiber carbon bar, placed in 350mm. The estimated requirements are 4Nm of dynamic torque, which, for a servomotor, would represent around 12Nm of holding torque, too much for an economic and normal servomotor. For solving this problem, a
3.2 Actuation

Figure 11: Electromechanic actuators. Common servomotor used in the prototype (left) and the DC motor used for lifting the leg from the lower joint (right).

<table>
<thead>
<tr>
<th>Function</th>
<th>Type</th>
<th>Specifications</th>
<th>Size (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upper joint</td>
<td>Servo</td>
<td>$M = 3\text{Nm}$, Speed $= 0.13\text{sec/60}^\circ$, $V_S = 12\text{V}$</td>
<td>$41 \times 38 \times 20$</td>
</tr>
<tr>
<td>Lower joint Lifting</td>
<td>Motor</td>
<td>$M = 16\text{Nm}$, RPM $= 120$, $I_f = 50\text{mA}$, $I_s = 1800\text{mA}$, $V_S = 12\text{V}$</td>
<td>$57 \times 35$</td>
</tr>
<tr>
<td></td>
<td>Gear</td>
<td>metallic, 148:1</td>
<td>$30 \times 35$</td>
</tr>
<tr>
<td>Lower joint Turning</td>
<td>Servo</td>
<td>$M = 1.5\text{Nm}$, Speed $= 0.14\text{sec/60}^\circ$, $V_S = 6\text{V}$</td>
<td>$41 \times 36 \times 20$</td>
</tr>
</tbody>
</table>

Table 2: List of electromechanic actuators used. $M$ is the stall torque, $I_f$ the free-run current, $I_s$ the stall current and $V_S$ the operating voltage.

DC motor has been used. This actuator is heavier than the servomotor used in the other joint, 228 grams versus 78 grams, but it is, at the same time, stronger, 16Nm holding torque versus 3Nm. Since in the lower joint the weight restriction is not a conclusive parameter, 150 grams more are less important than achieving the desired torque. As an advantage of this kind of actuator, the price is lower than the servomotor, nevertheless, the electronics requirements are high and an angle sensor and a motor controller will be needed. As a second actuator for the turning of the lower joint, we use a servomotor. Around of 1.5Nm holding torque is required, so we use again the simplest solution.

3.2.2 Thermodynamic actuation

Within thermodynamic actuation we have two types of actuators that have been used on the leg. For the heating process of the HMA, in order to detach the
brake, we use normal resistors, as cited in chapter 2.2.1, while for the cooling process, for attaching the brake, fans and heatsinks have been used. Then, the heating and cooling devices are explained.

A electrical resistor, as illustrated in Fig. 12, is a two-terminal passive electronic component which implements electrical resistance as a circuit element. When a voltage is applied across the terminals of a resistor, a current will flow through the resistor in direct proportion to that voltage. The power dissipated by a resistor is calculated as the product of multiplying the voltage between the terminals and the current flowing through the resistor. This power is transformed in heat which we use for heating the HMA in our design. Therefore, the increase of the voltage applied to the resistors entails the decrease of the time of the heating process. Nevertheless, there is a technical limit for the increase of the voltage, as soon as you reach the maximum designed power of the resistor, this will get destroyed.

For our prototype we have used 2 different arrays of resistors as a heater in the two joints. In the lower joint we have used resistors with a value of 1.5Ω, placed in groups of 2 in series and 9 of these groups in parallel, which give a total impedance of 3Ω. The voltage used for powering the heaters is of 18V, so the total heating power for the lower joint is of around 108W. In order to be able to try a bigger range of different heating powers, a different value for the resistors of the heater in the upper joint has been used. The new value is of 3Ω, that, with the same net configuration as in the previous case, gives a total impedance of 1.5Ω. Applying the 18V, we obtain a heating power of around 216W for the upper joint, being this joint faster in the heating process than the lower joint.

For placing the resistors, inside the holes of the sealed case of the brakes in the joints, the holes have been made on purpose, with small tolerance respect the diameter of the resistors used. In addition to this, thermal glue has been used for sealing the holes after inserting the resistors and for improving the thermal conductivity between the resistors and the box.

Since the heaters have been already explained, now is turn of the cooling system. It has been said already than the cooler consist in a synergy between two different elements, as we can see again in Fig. 12. On one hand we have the fans, which increase the air flow, used for decreasing the temperature of the brake, so, at the same time also the temperature of the HMA. On the other
3.3 Sensing 24

Figure 12: Illustration of the Thermodynamic actuators. On the left side, the heater system which is done by normal electrical resistors, on the right, the cooling system, where we use heatsinks and fans.

hand, heatsinks have been used for increasing the surface area, and, consequently, improve the efficiency of the air flow.

The fans used are different for each joint, while in the lower joint a strong 60x60mm fan is used, in the upper joint 2 thin 50x50mm fans are used. The reason is because in the upper joint we need to be light, so the smaller fans are lighter than the big fan, even using two of them, giving a similar performance.

Regarding the heatsinks, also 2 different design have been applied in each joint. In the upper joint 6 small heatsinks added around the box is the solution, while in the lower joint two big heatsinks, placed on both side of the box, are used. The performance in both ways is similar.

3.3 Sensing

The main focus of this thesis is to present a fully functional leg with 3 degrees of freedom using a brake made of HMA, rather than bothering with a complex sensor net or a high-developed control architecture. Therefore, the design is accomplished such that as less as possible control and sensors are needed. Nevertheless, there is no way to do it with no sensors at all. For the actuation of the lower joint, since we decided to use a DC motor, an angle sensor is required for the position control of the leg. For the upper joint a current sensor is used in order to protect the expensive servomotor against failure, measuring the current flowing through the actuator and defusing it in case of overload. Moreover, 2
Figure 13: Mounted angle sensor in the lower joint. The real lower joint on the right side of the picture, with the potentiometer placed at the end of the shaft.

more sensors are required for the data acquisition and for the autonomous mode. A temperature sensor is placed outside the sealed case of each brake, in order to read an estimated value of the temperature of the HMA.

### 3.3.1 Angle Sensor

The easiest way for measuring the rotation angle of the lower joint is using a simple and cheap potentiometer. For our purpose this solution is enough since the potentiometer offers the required features: rotation angle greater than 180° and enough accuracy.

The only problems found with the use of a normal potentiometer are the difficulty of installation and the high friction which have some of them. We had to find a potentiometer with low friction and then design a frame where install it attached to the shaft. In Fig. 13 it is possible to see the potentiometer used and its installation.

The behaviour of the potentiometer is simple, it actuates as a variable resistor where the value depends on the position of its shaft. Attaching the potentiometer’s shaft and our shaft from the joint together, we can know, by reading the value of the potentiometer, the exact angle of the joint. In further chapters we will explain the control architecture for the position of the lower joint by using
3.3 Sensing

<table>
<thead>
<tr>
<th>Method</th>
<th>Current type</th>
<th>Cost</th>
<th>Difficulty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clamp meter</td>
<td>AC</td>
<td>High</td>
<td>Middle</td>
</tr>
<tr>
<td>Rogowski coil</td>
<td>AC</td>
<td>Middle</td>
<td>High</td>
</tr>
<tr>
<td>Interferometer</td>
<td>DC</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Hall effect sensor</td>
<td>DC</td>
<td>Low</td>
<td>Middle</td>
</tr>
<tr>
<td>Resistor current sensor</td>
<td>DC</td>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>

Table 3: List of the different methods for measuring electrical current.

the data from the potentiometer.

3.3.2 Current sensor

In the upper joint, a servomotor is used instead of a DC motor like in the lower joint. This servomotor has a maximum holding torque of 3Nm, as cited in chapter 3.2.1, so it seems imperative the necessity of control this parameter in order to keep the good condition of the actuator. Imagine the actuator tries to lift the leg while the brake is attached, because a software bug, it may destroy itself. Or the foot of the leg is catch on the ground, the leg will try to lift and could damage itself if we do not detect than it is using more current than the nominal current of the servomotor.

In the market exist different method for measuring the electrical current. However, most of them can not be used in this application since the current is DC. Some suitable methods for AC current are using a current clamp meter or a Rogowski coil [1whatever], systems which requires an alternating current for his correct operation. The different thecnics which can be used for measuring DC current are the interferometer, the hall effect sensor and a resistor used as a current sensor [1whatever]. In the table 3.3.2 is possible to see the most important features of each of the current measuring methods.

As our intention is not a high accuracy measure, and we try to make an easy-understandable prototipe with a low cost, the chosen way to measure the current of our servomotor has been the resistor acting as a current sensor. There are in the market specific models of resistor designed for this specific task, but again, as we were running out of time and the required accuracy was not high we decided to use a normal resistor, specifically a power resistor as the one in the Fig.[14]

The power resistors have a bigger ceramic volume, which gives them a slower thermal response avoiding the excessive heating of the material, which would
cause a divert reading since the resistance of the resistors is highly dependent on the temperature.

The way of reading current with a normal resistor is connecting it along the load, all the current used by the load will have to pass through the resistor, and the voltage drop of a resistor is linearly proportional of the current flowing through it. Once the resistor is placed, the controller will read the voltage between the terminals of the sensor and transform the reading into current knowing the resistance of the resistor. As all the current will be have to pass through it, a low-resistance resistor is required, in order no to disturb the real load with the voltage drop of the sensor. We chose 0.1Ohm for our sensor, which gives a low voltage drop in the sensor and, at the same time, is enough for being measured by the ADC of the microcontroller with an acceptable resolution.

The operation mode of a servomotor consist of applying short pulses in high frequency with the desired relation between maximum voltage and 0 voltage to achieve a specified average voltage. Due to this behaviour, the supply line for the servomotor is highly noisy, what makes difficult the current measurement. For this reason, the frequency of the measurement is high, of the order of 1000 readings per second. The microcontroller is in charge of making all these measurements, saving them in memory and, afterwards, performing an average with at least 100 of them, which decrease the noise of the value.

This average current value is then processed and is checked if the maximum current of the servomotor is achieved, when that happens the servomotor is instantaneously switched off for a short period of time (around 200ms) and then switched on again, checking against the current to see if the environment changed and now the servomotor does not need overcurrent to achieve its desired new position. This loop is done a certain number of times (around 10), and if anything changes the servomotor is permanently switched off and an alarm is triggered.

An extra current sensor has been used as well in the lower joint, where the DC motor is used as an actuator. Thanks to this extra sensor, all these data could be send anytime to the PC to produce graphics. It is also possible to check the current used by the DC motor, avoiding overcurrent or excessive self heating, which may occur in the destruction of the actuator.
3.3 Sensing

3.3.3 Temperature sensor

In order to acquire data for the graphics, the most important value should be the temperature, since our brake performs in relation with this characteristic. The best solution would be adding a temperature sensor inside the brake, to measure the exact temperature of the HMA, but due to the difficulty of that, because the high pressure inside the device and the consequent problems of leakage that this would cause, we decided to place the sensor in the outside case, having an indirect measurement.

One temperature sensor has been added in each joint, in the low part of the outside case. The sensor is read by the microcontroller and after process the values, the data is sent to the PC, which generates the graphics with the rest of the data, such as torque, angle, current.

We have chosen a PT100 sensor [whatever], as the showed in the Fig. 15, for measuring the temperature. This sensor is a thermistor, which is a type of resistor whose resistance varies significantly with the temperature. This variation is the element used to read the temperature, reading the voltage drop in the sensor. There are two types of thermistors, depending on the sign of the variation, PTC
thermistor for a positive temperature coefficient and NTC thermistor for the ones with a negative temperature coefficient. The thermistors are useful over a limited temperature range, usually between $-90^\circ C$ and $130^\circ C$. And, in this case, the temperature coefficient is equal to $0.00385\%/^\circ C$. The resistance of the sensor in $0^\circ C$ is exactly $100\text{Ohm}$. The formula which relates the temperature of the sensor and its resistance is as follows:

$$\Delta R = K \Delta T$$ \hfill (3.1)

Where

- $\Delta R$ = change in resistance
- $\Delta T$ = change in temperature
- $K$ = temperature coefficient

The voltage drop in the sensor is read by the microcontroller, which applies this value into the formula plus the nominal resistance and obtain, as a result, the temperature. This new value is then sent to the PC, which generates the graphics.

### 3.4 Control

As already mentioned before the proposed approach not requires a high effort of control, since we mostly use a simple feedforward control. The only closed-loop control systems are used for the angular positioning of the lower-joint and for the control of the temperature of the Hot Meld Adhesive. In this section will be explained all different control systems, from the simplest but essential ones, to the high-level with close-loop control, ending with the highest level of control which covers the complete architecture of the robot, giving the possibility to execute a demonstrative step autonomously.

#### 3.4.1 Upper joint motor control

The upper joint, as we could see in the previous Fig. 8 in section 3.1.5, a servomotor is used for the control of the joint. Servos usually have a built-in feedback controller based on a potentiometer and a simple PI-controller and may therefore be driven by a simple control signal. This signal consist of a 0 level base signal with a 1 level pulse signal every 20ms, depending on the length of this positive
pulse, the servomotor will place its shaft in the desired angle. In Fig. 16 this method of control is explained in an easy and intelligible way.

The servomotor has three pin connections, one for ground, a second one for the power (12V), and the last one for the driving signal. This control signal is provided by our microcontroller board. The microcontroller generates it autonomously, with the information of the desired angle position. This desired angle is comming from the HMI in the PC or from the automatic behaviour model, which will be explained in a further section.
3.4.2 Lower joint motor control

In the case of the lower joint, as it was exposed in section 3.1.5, two degrees of freedoms are used. One degree for the vertical rotation and a second degree for the horizontal movement. The horizontal rotation is performed by a standard servomotor, as we saw in the previous section, but for the vertical movement we need to use a DC motor with a high torque, in order to be able to move the mass of the whole leg up and down, which requires a considerably higher torque. In the next subsections will be explained both control systems.

- Horizontal control

For the horizontal movement, the required torque is smaller than for the rotation of the previous upper joint. Because of that, even if both actuation systems are from the same type, servomotors, they are different brand and features. In the Table 3.2.1 we can see that, whereas for the upper joint we used a 12V servomotor with a torque of 3Nm, for the rotation of the lower joint is enough with a 6V servomotor with half of the torque, 1.5Nm. The control system of this servomotor is the same than for the bigger one, the microcontroller in sending through the three cables of the servomotor the control signal.

- Vertical control

As we already said, the vertical rotation of the lower joint is the most complicated movement in the whole leg, due to the high torque required. There are actually servomotors with the torque we need for this task, but they are extremely big and expensive, and the problem can be solve with a strong DC motor with a high gear reduction. Our DC motor is for 12V and, with a reduction of 148 to 1, gives a momentary torque as high as 16Nm. With this motor we would be able to lift a 1000mm length bar with a load heavier than 1500 grams at the end, this is enough for our leg, since the connection bar between the lower joint and the upper joint has a length of 700mm and the load at the end is of around 600 grams. The only disadvantage with the DC motor is the control system, which is not built-in, and you need an external power board to drive the motor and the microcontroller will control this new board. In the Fig. 17 we can see the motor control board chosen for our project.
Figure 17: DC Motor Control Board. This power board is used for driving the DC motor. It has a half H-bridge circuit with two Mosfet which makes possible to rotate the motor only in one direction. This is enough for our design since the gravity will push down the leg when we do not apply any energy to the motor, and then we will attach the brake to hold the robot when this makes a step.
3.4 Control

Figure 18: PWM Wave. Here we can see how the PWM signal works. The frequency of the switching is of around 3KHz. This speed is enough to simulate an analog signal, as marked by the red line

This power board accept an output of a maximum of 12V, which fits perfectly with our DC motor, and a maximum output current of 6Amps. which is large enough for dealing with the energy peaks of our motor.

For controlling the amount of energy given to the DC motor, this board has an input, from which depends the output to the motor. The input is a low energy signal between 0V and 12V, with a lineal dependency to the output. When in the input signal we have 2V, the output will be as well 2V but with high power capability, the same if the input is 6, 7, or any other value in the range. For generating this control signal, the microcontroller generates a PWM signal, as shown in Fig. [18] which is assumed as an analog signal, due the high frequency and the existence of a capacitance in the input line of the DC Motor control board.

Now that we know how to apply the desired voltage to the motor, it is necesary to calculate how much voltage do we need, in order to place the leg in the desired angle. Thanks to the angle sensor attached to the shaft of this joint, as already
explained in the previous section 3.3.1, we have a feedback for this control. With the output and the feedback we can design a PID-controller, as proposed on the following paradigm:

\[
u(t) = K_p \left[ e(t) + \frac{1}{T_N} \int_0^t e(\tau) d\tau + T_V \frac{d}{dt} e(t) \right], \quad e(t) = \alpha(t) - \alpha_d(t) \tag{3.2}\]

where \( e(t) \) is the error, \( \alpha(t) \) is the measured angle and \( \alpha_d(t) \) is the desired angle. We have found that choosing \( T_V = 0 \) to be sufficient for our experiments and that it results in a much simpler implementation. Once this control system is programmed in our controller board, we can give some test values and, by experimentation, modify them until finding good values, where the leg does not have unstability, and the desired angle is reached in a short time. However, there is a small problem arising from this approach. In the extremes of the angle range, the behavior of our closed-loop control is not good enough, due to the proximity of the end of the range, for this reason, we have decided to introduce an extra control code, which overcome this problem by modifying the parametres of the PI-controller when the angle of the leg is approaching the extremes.

3.4.3 Heating System Control

The heating system, as it was explained in previous chapters, is carried out by a group of resistances placed in the walls of the box which contains the glue for the brake. To heat up this resistances, we need a system which has to be able to give power to this resistances or stop it when the controller board needs it. Switching this device on and off has been achieved by a simple electronic circuit, based on relays and power transistors. In Fig. 19 is possible to see how the circuit is done in order to be able to apply voltage to the heating system only when the control board wants.

3.4.4 Cooling System Control

As with the heating system, the cooling system only needs a control circuit which applies, or not, certain voltage in the input of the fan used for cooling down the brake. The circuit for this control works exactly the same way as the circuit in Fig. ?? which was showed in the previous section.
Figure 19: Electrical circuit of the control for the heating and cooling systems. The heating system (in the left), as well as the cooling system (in the right), are controlled by relay designs. The relay is switching on or off the power to the device, and this relay is controlled by the microcontroller board through a transistor.

3.4.5 High-Level Control

The high-level control architecture of the robot is given by the control diagram shown in Fig. 20. As a main controller system, the Arduino Duemilanove, a microcontroller board based of the ATmega168 microcontroller from Atmel. The arduino board is a commercial board based in an open-source electronics prototyping platform, intended to be easy-to-use both hardware and software. This specific design has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, the USB connection to connect to the computer (for programming or communication), and a reset button. The microcontroller comes preburned with a bootloader and can therefore easily be programmed through the USB port of any computer.

The USB communication is used, apart for the programming of the board, for communicating with the computer, in order to receive the orders and to send back the sensor information. The behaviour of the USB connection is emulating a serial protocol, with a baudrate of 9600bps. Thanks to this emulation the arduino board can connect with a simple protocol to the workstation, communicating directly to the MATLAB software. From the MATLAB interface, it is possible to write a code which will be transfered to the arduino board and which will
3.4 Control

Figure 20: Electrical circuit diagram of the control of the whole robot. The computer is the highest level of control, this machine sends the control commands to the microcontroller board. This board controls all the rest of the electronic, both the DC motor control board and the relays board. All the not intelligent devices (motors, sensor, actuators) are hanging from one of this boards. Three different voltage sources are required to supply energy to the robot, due to different power requirements of each part, one of 5V for the electronics, 12V for the servomotors and the DC motor, and the last one of 18V for the heaters, which consume a lot of energy.
3.4 Control

<table>
<thead>
<tr>
<th>Command function</th>
<th>COM</th>
<th>PAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activates the heating procedure in the upper joint</td>
<td>10</td>
<td>-</td>
</tr>
<tr>
<td>Activates the cooling procedure in the upper joint</td>
<td>11</td>
<td>-</td>
</tr>
<tr>
<td>Move the servomotor of the upper joint to the desired position</td>
<td>12</td>
<td>20-160</td>
</tr>
<tr>
<td>Switch off the supply of power to the servomotor of the upper joint</td>
<td>13</td>
<td>-</td>
</tr>
<tr>
<td>Activates the heating procedure in the lower joint</td>
<td>20</td>
<td>-</td>
</tr>
<tr>
<td>Activates the cooling procedure in the lower joint</td>
<td>21</td>
<td>-</td>
</tr>
<tr>
<td>Move the CD Motor of the lower joint to the desired position</td>
<td>22</td>
<td>17-170</td>
</tr>
<tr>
<td>Switch off the supply of power to the DC Motor of the lower joint</td>
<td>23</td>
<td>-</td>
</tr>
<tr>
<td>Move the servomotor of the lower joint to the desired position</td>
<td>32</td>
<td>-</td>
</tr>
<tr>
<td>Switch off the supply of power to the servomotor of the lower joint</td>
<td>33</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4: List of available commands with associated parameters. **COM** is the command byte whereas **PAR1** is the associated parameter byte.

produce a reaction in our robot. The complete list of commands to perform the different actions in our design can be checked in Tab. 4. Each command can have an associated parameter, and, if so, it says the range of each of them.

For moving the leg, the order of the commands than the user should send through the computer to the arduino board would be: first, the robot should heat up the joint than wants to move, wait a certain time until the brake is detached and send the command to move the servomotor or DC Motor to the new position, then cool down the joint, wait some time while cooling and, as the last step, switch off the supply of power to the motor, since with the brake is enough to maintain the position in this joint, and in this way we save energy and avoid a possible damage if the software tries to move the actuator while the brake is attached.

### 3.4.6 Autonomous Mode

In order to show the operation of the robotic leg, it has been decided to program a code for moving autonomously the device, doing endless steps in the same
position. For achieve this behaviour, we have fixed the robot on a desk, as it was shown in a previous Fig[7]. Once we have power up all the power supplies than the robot needs, it will start running the following commands:

1. Both joints are heated up until reach the melt temperature of the HMA.

2. The lower joint lifts the leg until the vertical position and remains there.

3. The upper joint flex the leg until both parts of the leg are in vertical position, one next to the other. This is not completely necessary but decrease the strength that the servomotor of the lower joint has to apply in order to move the leg.

4. Depending on the current position of the leg in the horizontal plane, the servomotor of the lower joint will rotate the whole leg to the position in the other extreme.

5. Once the leg is in the other extreme, both joints move until the initial position, resting the leg on the ground, in an approximately right angle.

6. Both joints are cooled down until reach the limit of the elastic behaviour of the HMA.

7. The process starts again in the opposite direction.

Thanks to the leds placed on the joints, one red and one blue led on each joint, the audience is able to see when each joint is doing the heating or the cooling phase. The power supplies are, as well, visible for the public to check at any moment the current consumption of the sources when the robot is doing each of the actions.
4 Experiments

In this section we analyze the braking performance of the proposed approach. Based on the real-world performance test, we will then extend our analysis to discuss the expected performance of a possible robot with six of these legs.

4.1 Heating Performance

In the Fig. 21 we can see the different performance for all the input powers tested on the heating system. A regulated digital power supply was used for set the different inputs. Thanks to this information we can choose which power is the one with the best performance, or in other words, which one gives the best relation between the speed of the heating and the consumption of energy.

![Heating Process](image)

Figure 21: Performance of the heating process with different input power. It was necessary to try the heating system with different intensities to find out which one has the best ratio between the performance and the consume.

With these results a new graphic has been made to show which one is the relation between the data obtained. We have decided to make our own pattern to find a valid result, which can give us a clue about which power we should use in the heater. This study is shown in Fig. 22 where, as well, it is explained in more
4.2 Cooling Performance

detail. A temperature of 70°C has been chosen as a temperature of reference for the study, due to the fact that is the limit in which the HMA is soft enough for the motor to move the respective joint. We must say that this temperature, as it was previously exposed, is not the temperature of the braking material, is the temperature of the outside of the case, being the inside part fairly warmer.

<table>
<thead>
<tr>
<th>CASES</th>
<th>HEATER POWER</th>
<th>TIME TO 70°C</th>
<th>POWER RELATION</th>
<th>TIME RELATION</th>
<th>BEST ESTIMATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>16.94</td>
<td>440.00</td>
<td>0.21</td>
<td>1.00</td>
<td>0.21</td>
</tr>
<tr>
<td>2</td>
<td>77.00</td>
<td>220.00</td>
<td>0.30</td>
<td>0.90</td>
<td>0.17</td>
</tr>
<tr>
<td>3</td>
<td>77.00</td>
<td>160.00</td>
<td>0.45</td>
<td>0.30</td>
<td>0.15</td>
</tr>
<tr>
<td>4</td>
<td>103.50</td>
<td>115.00</td>
<td>0.45</td>
<td>0.30</td>
<td>0.13</td>
</tr>
<tr>
<td>5</td>
<td>151.00</td>
<td>75.00</td>
<td>0.70</td>
<td>0.17</td>
<td>0.12</td>
</tr>
<tr>
<td>6</td>
<td>216.00</td>
<td>52.00</td>
<td>1.00</td>
<td>0.11</td>
<td>0.11</td>
</tr>
</tbody>
</table>

Figure 22: Estimation of the best performance of the heating process. From the obtained data, we keep the worse value as a reference value, from which all the relations are created. In the power case, the 216W are the reference, and working with the time until 70°C, the longest time is 440 seconds. From these starting points, a lineal relation is done for both parameters. The result is a unitary value for each of the worse cases and a lineal progression descending in values lower than one. With these two series of data, we multiply them and obtain a table of values, all lower than 1. The lowest value from this table is the one with the best performance of the parameters chosen. In this case, we can see than the highest power input, the best. The maximum power input for our heater system, without burning it, is the 216W. This is the value we have used for our experiments, since is the best one we can use.

4.2 Cooling Performance

As with the heating process, we can analyze the behavior of our cooling system reading the temperature in relation with the time and in both joints. In each
joint we can activate or deactivate the active cooling system. All this is what we can see in Fig. 23.

![Cooling Process](image)

Figure 23: Performance of the cooling process. We can see two different implementations of the cooling system, one for the lower joint and one for the upper joint. In each of them, we measure the temperature of the system cooling in two different states, one with only passive cooling, and the other one with the active cooling activated.

Thanks to these data, we can say that both prototypes are quite similar even being slightly different designs. One of them has two small fans in serie as an active cooling system. Each fan is in one of the sides of the metal case which contains the HMA. The other design consists in only one big fan in one of the sides of the aluminum case.

What can be certainly said is that the active cooling is much better than only the passive system, arriving to be four times faster for the important case for us, in the range between 70°C and 45°C, the limits between brake attached and detached. In this case, the passive systems takes around of 550 seconds, and the active system takes only 125 seconds, more than four times faster.

As the consumption of the fans is negligible in relation with the energy used in the heating process, the use of the active cooling is very beneficial, due to that, for our next experiments, we have always worked with the active cooling.
4.3 Torque vs Bending Angle

An important feature for a brake design is the braking torque it can withstand. A normal brake is rigid, the brake suddenly breaks with no possible repair. Our design is different, the brake can never be breaking because an excessive force, it will only be a momentary breakage, coming back to the initial state once the HMA has been heated up. This important property also causes another situation, and is that the brake bends in relation of the applied torque, due to its intrinsic flexibility.

![Torque vs Angle Graphic](image)

Figure 24: Torque vs bending angle graphic. In this graphic we can see the obtained data as the blue dots, and a linearization line in green color. The room temperature during this experiment was of 23°C.

In the previous graphic, Fig. 24 we can see how the data from the experiments is quite lineal. Thanks to that we can say that a higher torque will only produce a higher bending angle. In each one of the situations, the limit of the bending angle is one or another, for this reason is important to know how should we expect the angle in relation with the torque applied.
4.4 Torque vs Temperature

For a correct estimation of the possible behaviour of our leg, two good parameters to relate would be the torque in relation with the temperature, in order to, later, be able to anticipate the attachment and detachment of our brakes. In Fig. 25, we can see the graphic result of the experiment.

![Torque vs Temperature graphic](image)

Figure 25: Torque vs temperature graphic. In this graphic we can see the obtained data as the blue dots, and a linearization line in red color. The room temperature during this experiment was of 23°C. We can see as the experimentation data follow a negative cubic curve.

For executing this experiment, we decided to set as a torque, the one which bend the leg in an angle of 15°, since, due to the flexibility of the system, was the best solution for having a fair result. As we can see, the temperature influence in the maximum torque in a cubic relation, which means that as long as we can wait longer for the joint to cool down, the posible torque you will be able to apply to the brake will encrease in a cubic speed.

Of course, there is a limit in the amount of torque we can apply, since the ambient temperature is normally around 23°C, and that even in close values to this temperature, the behaviour is not lineal, but tending to the maximum torque in 15°, around 14Nm.
4.5 Walking Speed Estimation

In the last experiment we have carried out, we try to simulate the behaviour of a possible finished robot using our brakes in all of the joints. The first step for estimating this behaviour is experimenting with each one of the joints by itself. For that, we have done an experiment in which why try to simulate steps in each individual joint. When the brake is attached, we heat up until it is detached and then we wait until is cooled down and detached again, moment when the process starts again. We repeat this pattern for a few steps to see how could it work in an stationary state. In Fig. 26 we can see the experiment for the lower joint and in Fig. 27 for the upper joint.

![Graph](image)

Figure 26: Steps simulation of the lower joint. As we can see in the stationary pattern, the lower joint needs to be heated up until 55-60°C to be detached, and you have to wait until is as cold as 35°C to consider the brake again attached.

We can see a substantial difference between the joints. This is due to the way of considering attached or detached the braking systems. For our experiments, we consider the detached state when the joint bends because the gravity force. And the attached state was when the brake was able to keep the position of the joint in 90°. Because of this way of considering the states, the lower joint, which has connected the whole leg, detaches earlier because of the weight, and attaches
later because we have to wait until a lower temperature, for the brake to be able to support more weight. For the same reason, the upper joint works in the other way around.

![Simulation of steps of the upper joint](image)

Figure 27: Steps simulation of the upper joint. As we can see in the stationary pattern, the upper joint needs to be heated up until 60-65°C to be detached, and you have to wait until is as cold as 45°C to consider the brake again attached.

As a conclusion of the experiment we can say than the slowest joint is the lower one, which needs around 200 seconds to do a step. Considering a length of a step of 80cm., we can say than the maximum speed of our imaginary robot is of 14.4 m/h.

Each of the joints is different, the lower joint consumes 104W while heating and 2W while cooling. The upper joint, by contrast, only consumes 216W while heating and 2 during the cooling process. With these values we can calculate the total consume of each step, and, at the end, the locomotion energy cost of the robot.

For the lower joint, the energy required for doing one average step is of 1520mWh, and for the upper joint is 1350mWh. In total, the locomotion energy cost of steps of 80cm is of 3587Wh/Km. For having a comparison, the locomotion energy cost of a human walking is 20Wh/Km.
5 Conclusion

This thesis proposed an approach to braking systems based on hot melt adhesive (HMA). HMA, which classifies within chemical adhesion, has so far not been considered as an braking technique for robots in the literature. Based on this new technique we developed a simple robotic platform which enabled us to investigate the proposed approach and evaluate it.

Commercially available HMA was used and its thermodynamical properties have been analyzed. We have found that the temperature of the HMA and the strength of the brake are related, the higher the temperature, the lower the braking strength in a square root relation (Fig. 25). For this reason, it is really important the waiting time while cooling down the joint, in order to have enough resistance. The normal resistors have been found to be very suitable for the heating process, because of the fact that different resistance values can be choosen for each application, depending on the voltage and current source, and in the desired speed of the process. Whereas the electric fans have been found not very suitable, having long cooling times and not being hable to speed up the process by giving a high power input. A new approach will be discussed in the next section regarding this issue.

An analysis of the braking strength revealed good results in comparison with different braking systems in the market. As an example, the ratio between the torque of the brake and its weight has been calculated for different kind of brakes, in the following table, Tab. 5, we can see the results.

<table>
<thead>
<tr>
<th>Brake types</th>
<th>Torque to weight relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electromagnetic brake</td>
<td>19.44 NmKg</td>
</tr>
<tr>
<td>Electromagnetic power off brake</td>
<td>2.00 NmKg</td>
</tr>
<tr>
<td>Pneumatic brake</td>
<td>13.75 NmKg</td>
</tr>
<tr>
<td>Hydraulic brake</td>
<td>17.40 NmKg</td>
</tr>
<tr>
<td>HMA based brake</td>
<td>24.00 NmKg</td>
</tr>
</tbody>
</table>

Table 5: List of the different braking systems in the market. Our approach is in the last line.

For the previous table we have chosen the results of our experiments when the leg was bending 15°C, since the flexibility of our robot is one of the important features. Thanks to the fact that our design can not be broken due to an excessive torque applied to the joint, since the HMA will behave as before once we have
reach the melting temperature again, we can say that the nominal torque of
the brake is the same as the maximum torque. In the other mechanical design,
since they do not use any elastic part in the system, a security margin has to be
given in order to prevent a failure in the device. Having this in consideration, our
approach has shown a better performance than any other kind of brake in the
market, meaning a feasible future for its use in different areas where the attaching
and detaching time of the brake is not primordial.

Along with this positive result, we must mention the fact that only two of
the mentioned systems have the capability of remain in an attached position with
absolutely no power input. This two devices are, in one hand the Electromagnetic
power-off brake, in which the attaching force is coming from a spring, which
lead to a low torque to weight relation. And, in the other hand, our design using
the HMA approach, which once is cold does not need any extra power input for
remaining in this position as long as required.

5.1 Future Work

In this section we propose some possible focuses for future improvements in the
mechanical design in order to achieve a better performance of our approach. After
all the information we have given about our design, we can conclude than the
main points to value the performance of the brake are in order of importance:
the braking torque, the attaching time, the detaching time and the consumption.
We can be proud of the braking torque reached but we have three more aspects
than have to be improved in order to create a fully functional robot.

The consumption of our device depends directly on the amount of HMA and
other materials which have to be heated up and cooled down in each cycle. For
reducing the energy waste we suggest to reduce the quantity of materials exposed
to the heating source. The first point would be to reduce the amount of HMA
used inside the brake, by reducing the volume of the cavities between the internal
mobile part and the external case with a new design, where both parts are closer
to each other. But most part of the energy is not going to the adhesive, but to
the metallic parts, because this material has better performance in terms of heat
transfer. To try to solve this loss of heat, the inside part should be made with a
kind of material with a bad heat transfer value, as, for example, a ceramic part.
In this way we would avoid almost all the losses through the internal axis. Since
we can not do the same for the outside part, because we would block the heat transfer during the cooling cycle, we have thought in a double improvement, which will, at the same time, speed up the detaching speed. A thin sheet of aluminum is surrounding the HMA, next, a net of slim pipes, where cooling water flows, is placed in contact with the sheet with lines of resistors in between, and to cover everything, an insulating material wraps the pipes, avoiding the cold or the heat to be transmitted to the exterior.

The way of working of this new design would be as following: for the heating phase, the cooling water stops and the heaters, so the resistors, are switched on. In this case, we only have to heat up the pipes with a small volume of water, the thin sheet of aluminum and the HMA itself. Increasing notably the speed of the heating process because the amount of material to heat up is much lower. For the cooling phase, the heaters are stopped, and the water is forced to flow, being this cooled down in a main chiller. In this case, the improvement in the speed of the cooling process should be much bigger, since we are reducing the quantity of material to be cooled down, but, at the same time, we are improving the cooling device, by replacing forced air by forced water, which has a higher ability to work as a cooling fluid.

As final points for this sections of future work we could mention the problem that we have had with the leakage of HMA in our devices. A better building process has to be used in order to not let any play to appear, and, in this way, eliminate all the possible leakages that could end in an insufficient amount of adhesive inside the braking system. And, for ending, it would be interesting to build the whole robot, with its all six legs and with the new improved joints, in order to show the strength of this new approach in the field of robots.
References


A. Appendix

A.1 Matlab Code

A.1.1 Matlab InitRobot.m

```matlab
fclose(s);
clear all;
clc;

% Leg experiment - servo + glue brake

s = serial('COM18','baudrate',9600); % define serial port and baudrate
set(s, 'terminator', 'LF'); % define the terminator for println
set(s, 'timeout', 3);
fopen(s);
pause(2);

try
    % use try catch to ensure fclose
    display('Starting Loop');

    while 1

        userEntry = input('Do you want Servo1: 10-13 , DCMotor: 20-23 or Turning: 32? ');
        if isempty(userEntry)
            userEntry = 0;
        end

        fwrite(s,userEntry,'uint8');

        w=fscanf(s,'%s'); % must define the input % d or %s, etc.

        if (w=='e')
            display('Overcurrent');
            % fprintf(s,'%s\n','J'); % establishContact just wants
            % something in the buffer
        end

        if (w=='o')
            display('Everything OK');
            % fprintf(s,'%s\n','J'); % establishContact just wants

        end

    end

end
```

% something in the buffer
end
pause(1);
end
display(' se acab!');
fclose(s);
catch me
close(s); % always, always want to close s
end

A.1.2 Matlab SerialWrite.m

function serial_write(s,value)
fwrite(s,value,'uint8');
end

A.1.3 Matlab SerialWriteServo.m

function serial_writeServo(s,servo,pos)
fwrite(s,servo,'uint8');
fwrite(s,pos,'uint8');

A.1.4 Matlab Heating.m

function heating1()
%HEATING1: activate the heater 1 and switch off all the fans from system 1
fwrite(s,010,'uint8');
end

A.1.5 Matlab Cooling.m

function cooling1()
%cooling1: activate the fans 1 and switch off the heater from system 1
fwrite(s,012,'uint8');

end

A.2 Microcontroller Code

A.2.1 Robot Main Code (LegControl.cpp)

#include <Servo.h>
#include <PID_Beta6.h>

#include <Servo.h>
#include <PID_Beta6.h>

/*******************
* IO Port definition
*******************/
// create servo object to control a servo
Servo servo1;
Servo servo3;

//Init variables
int servo1port = 4;       //Servomotor port
int servo2port = 5;       //DC Motor port
int servo3port = 9;       //Servomotor for turning the leg port

const int pot1 = 2;       // Analog input pin that the potentiometer 1 is attached to
const int shunt1 = 0;     // Analog input pin that the shunt 1 is attached to
int heater1 = 2;
int cooler1 = 3;

const int pot2 = 3;       // Analog input pin that the potentiometer 1 is attached to
const int shunt2 = 1;     // Analog input pin that the shunt 1 is attached to
int heater2 = 6;
int cooler2 = 7;

int led = 8;

const int Kp = 6;
const int Ki = 30;
const int Kd = 50;
A.2 Microcontroller Code

33 /***************
34 * Global variables
35 ********************/
36 float shunt1Value = 0;
37 double value1 = 0;
38 int resetServo1Value = 90;    //
39 int maxCurrentServo1 = 4000;  //mA
40 int commandByte;
41 int parameterByte;
42
43 float shunt2Value = 0;
44 double value2 = 0;
45 int pot2Value = 0;
46 int resetServo2Value = 75;    //
47 int maxCurrentServo2 = 4000;  //mA
48 int stateServo2 = 0;
49
50 int resetServo3Value = 90;    //
51 int maxCurrentServo3 = 4000;  //mA
52 int error;
53 int errorLast;
54 double control;
55
56 /**************
57 * Setup
58 ********************/
59 void setup()
60 {
61    analogWrite(servo2port, 255); //Switch off the motor, 0V.
62    //Init GP I/O
63    pinMode(heater1, OUTPUT);
64    pinMode(cooler1, OUTPUT);
65    pinMode(heater2, OUTPUT);
66    pinMode(cooler2, OUTPUT);
67    pinMode(servo2port, OUTPUT);
68    pinMode(led, OUTPUT);
digitalWrite(heater1, HIGH); //Heaters off
digitalWrite(cooler1, HIGH); //Coolers off
digitalWrite(heater2, HIGH);
digitalWrite(cooler2, HIGH);
digitalWrite(led, LOW);

//Activate the heaters, in order not to break the motors
//when they try to go to their reset position
// digitalWrite(heater1, LOW);
// digitalWrite(heater2, LOW);
// delay(30000); //the program will be stopped here 3min while heating
// digitalWrite(heater1, HIGH);
// digitalWrite(heater2, HIGH); //Switch them off again

//Init Servos
 servo1.attach(servo1port); // attaches the servo on correspondent pin.
 servo1.write(resetServo1Value);

 servo3.attach(servo3port); // attaches the servo on correspondent pin.
 servo3.write(resetServo3Value);

//Init Serial Communication
 Serial.begin(9600);
 establishContact(); // send a byte to establish contact until receiver responds
}

/**************************
* Main function
**************************/
void loop()
{
  if (commandByte == 0) //READ INCOMING COMMAND byte, byte 0 is set whenever
  {
    if (Serial.available()>0)
    {
      commandByte = Serial.read();
    }
A.2  Microcontroller Code

```c
else //PERFORM ACTION
{
    //-----------------------------
    //SERVOMOTOR 1 CONTROL
    //-----------------------------
    if (commandByte == 10) //Heating system 1
    {
        digitalWrite(heater1, LOW);
        digitalWrite(cooler1, HIGH);
        commandByte = 0;
        //Action successfull
    }
    else if (commandByte == 11) //Cooling system 1
    {
        digitalWrite(heater1, HIGH);
        digitalWrite(cooler1, LOW);
        commandByte = 0;
        //Action successfull
    }
    else if (commandByte == 12) //Control Servo 1
    {
        if (Serial.available() > 0)
        {
            parameterByte = Serial.read();
            commandByte = 0;
            servo1.attach(servo1port); // attaches the servo on correspondent pin.
            if (parameterByte > 170) {
                parameterByte = 170;
            }
            else if (parameterByte < 17) {
                parameterByte = 17;
            }
            int j = servo1.read();
            if (j != parameterByte) {
                resetServo1Value = servo1.read();
            }
            servo1.write(parameterByte);
            //Action successfull
        }
    }
```
else if (commandByte == 13) //Reset Servo 1
{
    servo1.detach();
    commandByte = 0;
    //Action successful
}

------------------------------------
//DC MOTOR CONTROL
------------------------------------
else if (commandByte == 20) //Heating system 2
{
    digitalWrite(heater2, LOW);
    digitalWrite(cooler2, HIGH);
    commandByte = 0;
    //Action successful
}
else if (commandByte == 21) //Cooling system 2
{
    digitalWrite(heater2, HIGH);
    digitalWrite(cooler2, LOW);
    commandByte = 0;
    //Action successful
}
else if (commandByte == 22) //Control DCMotor
{
    if (Serial.available() > 0)
    {
        parameterByte = Serial.read();
        commandByte = 0;
        if (parameterByte > 170) {
            parameterByte = 170;
        } else if (parameterByte < 17) {
            parameterByte = 17;
        }
        resetServo2Value = parameterByte - 23;
        analogWrite(servo2port, 255); //Switch off the motor, 0V.
        stateServo2 = 100;
        //Action successful
    }
}
else if (commandByte == 23) //Reset DCMotor
{
    stateServo2 = 0;
    resetServo2Value = resetServo2Value;
    commandByte = 0;
    //Action successfull
}

//SERVOMOTOR FOR TURNING CONTROL

else if (commandByte == 32) //Control Servo for turning the leg
{
    if (Serial.available()>0)
    {
        parameterByte = Serial.read();
        commandByte = 0;
        if (parameterByte > 155) {
            parameterByte = 155;
        }
        else if (parameterByte < 25) {
            parameterByte = 25;
        }
        servo3.write(parameterByte);

        while(resetServo3Value < parameterByte)
        {
            resetServo3Value = resetServo3Value + 1;
            servo3.write(resetServo3Value);
            delay(15);
        }
        while(resetServo3Value > parameterByte)
        {
            resetServo3Value = resetServo3Value - 1;
            servo3.write(resetServo3Value);
            delay(15);
        }
        resetServo3Value = parameterByte;
        //Action successfull
    }
}
```c
else
{
    commandByte = 0;
}

//------------------------------------
//SERVOMOTOR CONTROL --- CURRENT CONTROL --- SERIAL WRITTING
//------------------------------------

value1 = 0;
shunt1Value = 0;

for (int x = 0; x < 50; x++) //Read the sensor a lot of times and make the average
{
    value1 = value1 + analogRead(shunt1);
}
shunt1Value = value1 / 50;
shunt1Value = (shunt1Value / 1023) * 5 * 10 * 1000; //Show the mA that consumes the servomotor

if (shunt1Value > maxCurrentServo1)
{
    Serial.println('e'); //e == Error, current exceeded
    // Serial.println(shunt1Value);
    digitalWrite(led, HIGH);
    servo1.detach();
    delay(500);
    servo1.attach(servo1port);
}
else
{
    Serial.println('o'); //0 == System OK
    // Serial.println(shunt1Value);
    digitalWrite(led, LOW);
}

//------------------------------------
//DC MOTOR CONTROL --- PID
//------------------------------------
```
pot2Value = analogRead(pot2);
// map it to the range of the mechanical system:
pot2Value = map(pot2Value, 0, 1023, 0, 180);

// while(pot2Value > resetServo2Value)
{
    // control = pot2Value - 2;
    //
    // error = control - pot2Value;
    //
    // control = Kp*error + Ki*errorLast + Kd*(error-errorLast);

    // if (pot2Value > 90)
    // {
    //     control = ((180 - pot2Value)^2)/10;
    // }
    //
    // else
    // {
    //     control = (pot2Value^2)/10;
    // }

    // if (control < 0)
    // {
    //     control = 0;
    // }
    // if (control > 255)
    // {
    //     control = 255;
    // }
    // if (stateServo2 == 0)
    // {
    //     analogWrite(servo2port, 255);
    // }
    // else
    // {
    //     analogWrite(servo2port, 255 - control);
    // }

    digitalWrite(led, HIGH);
    delay(50);
    digitalWrite(led, LOW);
    delay(50);
// pot2Value = analogRead(pot2);
// // map it to the range of the mechanical system:
// pot2Value = map(pot2Value, 0, 1023, 0, 180);

control = resetServo2Value;

error = control - pot2Value;
control = Kp*error + Ki*errorLast + Kd*(error-errorLast);

if (control < 0)
{
    control = 0;
}
if (control > 255)
{
    control = 255;
}
if (stateServo2 == 0)
{
    analogWrite(servo2port, 255);
}
else
{
    analogWrite(servo2port, 255 - control);
}
errorLast = error;

/********************
* External Functions
********************/

//void establishContact() {
//  while (Serial.available() ≤ 0) {
//      Serial.println('A', BYTE); // send a capital A
//      delay(300);
A.2.2 Robot Code for Automatic Mode (AutomaticLegControl.cpp)

```cpp
#include <Servo.h>
#include <PID_Beta6.h>

/* This program is totally autonomous, its behaviour consist of doing steps non stop. */

/*****************
* IO Port definition
*****************/
// create servo object to control a servo
Servo servol;
Servo servo3;

//Init variables
const int TempSensor1port = 5; // Analog input pin for the Temperature sensor 1
const int TempSensor2port = 4; // Analog input pin for the Temperature sensor 2
int servolport = 4; // Servomotor port
int servot2port = 5; // DC Motor port
int servot3port = 9; // Servomotor for turning the leg port
const int pot1 = 2; // Analog input pin that the potentiometer 1 is attached to
const int shunt1 = 0; // Analog input pin that the shunt 1 is attached to
int heater1 = 2;
int cooler1 = 3;
const int pot2 = 3; // Analog input pin that the potentiometer 1 is attached to
const int shunt2 = 1; // Analog input pin that the shunt 1 is attached to
int heater2 = 6;
int cooler2 = 7;
int led = 8;
const int Kp = 6;
const int Ki = 30;
const int Kd = 50;
```
/********************
* Global variables
********************/
float shunt1Value = 0;
double value1 = 0;
int resetServo1Value = 90; //
int maxCurrentServo1 = 3000; //mA
int commandByte;
int parameterByte;

float shunt2Value = 0;
double value2 = 0;
int pot2Value = 0;
int resetServo2Value = 75; //
int maxCurrentServo2 = 4000; //mA
int stateServo2 = 0;

int resetServo3Value = 90; //
int maxCurrentServo3 = 4000; //mA
int TempSensor1 = 0;
int TempSensor2 = 0;
int error;
int errorLast;
double control;

int stateLeg = 0; //0 = heating 1 = cooling

/*****************
* Setup
*****************/
void setup()
{
    analogWrite(servo2port, 255); //Switch off the motor, 0V.
//Init GP I/O
pinMode(heater1, OUTPUT);
pinMode(cooler1, OUTPUT);
pinMode(heater2, OUTPUT);
pinMode(cooler2, OUTPUT);
pinMode(servo2port, OUTPUT);
pinMode(led, OUTPUT);

digitalWrite(heater1, HIGH);  //Heaters off
digitalWrite(cooler1, HIGH);  //Coolers off
digitalWrite(heater2, HIGH);
digitalWrite(cooler2, HIGH);
digitalWrite(led, LOW);

//Init Servos
// servo1.attach(servo1port);  // attaches the servo on correspondent pin.
// servo1.write(resetServo1Value);

servo3.attach(servo3port);  // attaches the servo on correspondent pin.
servo3.write(resetServo3Value);

//Init Serial Communication
Serial.begin(9600);

// establishContact();  // send a byte to establish contact until receiver responds
}

/*****************
* Main function  
*****************/
void loop()
{
  //------------------------------------
  //HEATERS, COOLERS AND MOTORS CONTROL / AUTOMATIC BEHAVIOUR
  //------------------------------------
  if(stateLeg == 0)
  {
    if(TempSensor1 < 70)
```c
{ 
  if (TempSensor2 < 65)
  {
    digitalWrite(heater1, LOW); // Heaters on
    digitalWrite(cooler1, HIGH); // Coolers off
    digitalWrite(heater2, LOW);
    digitalWrite(cooler2, HIGH);
  }
  else
  {
    digitalWrite(heater1, LOW); // Heater 1 on
    digitalWrite(cooler1, HIGH); // Cooler 1 off
    digitalWrite(heater2, HIGH); // Heater 2 off
    digitalWrite(cooler2, HIGH); // Cooler 2 off
  }
}
else
{
  if (TempSensor2 < 65)
  {
    digitalWrite(heater1, HIGH); // Heater 1 off
    digitalWrite(cooler1, HIGH); // Cooler 2 off
    digitalWrite(heater2, LOW); // Heater 2 on
    digitalWrite(cooler2, HIGH); // Cooler 2 off
  }
  else
  {
    digitalWrite(heater1, HIGH); // Heater 1 off
    digitalWrite(cooler1, LOW); // Cooler 1 on
    digitalWrite(heater2, HIGH); // Heater 2 off
    digitalWrite(cooler2, LOW); // Cooler 2 on
    resetServo2Value = 130;
    servo1.write(140);
    delay(10000); 
    if (servo3.read() < 90)
    {
      servo3.write(180);
    }
    else
    {
      servo3.write(0);
    }
  }
```
A.2 Microcontroller Code

```c
160     
161     delay(500); 
162     servo1.write(85); 
163     delay(100); 
164     resetServo2Value = 80; 
165     stateLeg = 1; 
166     
167     
168     
169     else 
170     { 
171       if(TempSensor1 < 48) 
172       { 
173         if(TempSensor2 < 35) 
174         { 
175           stateLeg = 0; 
176         } 
177       } 
178     } 
179     
180     // digitalWrite(led, HIGH); 
181     // delay(50); 
182     // digitalWrite(led, LOW); 
183     
184     //------------------------------------ 
185     //TEMPERATURE READING 
186     //------------------------------------ 
187     double average = 0; 
188     for (int i = 0; i < 80; i++) 
189     { 
190       average = average + analogRead(TempSensor1port); 
191     } 
192     average = average / 80; 
193     average = ((average / 1024)*4.95)*99.2 ; 
194     average = ((average/100)-1)/0.00385; 
195     TempSensor1 = average; 
196     // Serial.println(average); 
```
```c
average = 0;
for (int i = 0; i < 80; i++)
{
    average = average + analogRead(TempSensor2port);
}
average = average / 80;

average = ((average / 1024)*4.95)*99.2 ;
average = ((average/100)-1)/0.00385;
TempSensor2 = average;
Serial.println(average);

//SERVOMOTOR CONTROL --- CURRENT CONTROL --- SERIAL WRITTING

value1 = 0;
shunt1Value = 0;
for (int x = 0; x < 50; x++) //Read the sensor a lot of times and make the average
{
    value1 = value1 + analogRead(shunt1);
}
shunt1Value = value1 / 50;
shunt1Value = (shunt1Value / 1023) * 5 * 10 * 1000; //Show the mA that consumes the servomotor

if (shunt1Value > maxCurrentServo1)
{
    Serial.println('e'); //e == Error, current exceeded
    Serial.println(shunt1Value);
    digitalWrite(led, HIGH);
    servo1.detach();
delay(500);
    servo1.attach(servo1port);
}
```
else
{
  Serial.println('0');   //0 == System OK
  Serial.println(shunt1Value);
  digitalWrite(led, LOW);
}

//------------------------------------
//DC MOTOR CONTROL --- PID
//------------------------------------
pot2Value = analogRead(pot2);
// map it to the range of the mechanical system:
pot2Value = map(pot2Value, 0, 1023, 0, 180);
control = resetServo2Value;
error = control - pot2Value;
control = Kp*error + Ki*errorLast + Kd*(error-errorLast);

if (control < 0)
{
  control = 0;
}
if (control > 255)
{
  control = 255;
}
if (stateServo2 == 0)
{
  analogWrite(servo2port, 255);
}
else
{
  analogWrite(servo2port, 255 - control);
}
errorLast = error;

A.3  Data sheets
Figure 28: Schematic of the Arduino Duemilanove microcontroller board [7].