

Technical Report

IRI-TR-10-04



iCub Platform

IIT Workshop in Genova

Sergi Hernández Juan
Guillem Alenyà Ribas



Institut de Robòtica i Informàtica Industrial

Abstract

This document outlines the most important concepts presented during a workshop about the iCub robot done at the Instituto Italiano di Tecnologia in Genova. Mechanical, electronic as well as firmware and software issues are presented, and the basic procedures to detect and solve the most common problems are described.

The most important goal of this workshop was to get the necessary skills to perform the most basic maintenance of the robot without having to depend on the support from IIT. Also a brief introduction to the main issues of the control of the robot were provided.

Institut de Robòtica i Informàtica Industrial (IRI)

Consejo Superior de Investigaciones Científicas (CSIC)

Universitat Politècnica de Catalunya (UPC)

Llorens i Artigas 4-6, 08028, Barcelona, Spain

Tel (fax): +34 93 401 5750 (5751)

<http://www.iri.upc.edu>

Corresponding author:

Sergi Hernàndez Juan

tel: +34 93 401 1901

shernand@iri.upc.edu

<http://www.iri.upc.edu/people/shernand>

1 Mechanical assembly

Eventhough the robot comes completely assembled, some times it is necessary to replace some steel cable that brakes, and, in general, replacing a cable would imply disassembling part of the robot and then assembling it again. Therefore, it is useful to see how it is done, which tools are required and specific tips and tricks to make things easier.

First a list of the necessary tools both to assemble and disassemble the robot and to build and replace the cables are listed in section 1.1. Then the standard procedure to build a cable replacement is explained in section 1.2, and the particular details of cable replacing for both the shoulder and the set of hand and fore arm are described in sections 1.3 and 1.4 respectively.

In this document the torso of the robot, as well as the legs and the head are not dealt with because they where not explained during the workshop. For details on these parts, procedures to assemble them and information on the required steel cable lengths, see the iCub platform documentation [1].

1.1 List of necessary tools

In this section a list of all the necessary tools are listed, and their main function explained. Most of the tools listed are normally found in any workshop, but some are more specific.

- **Allen key set:** These keys are needed to place and remove all the allen type screws present in the robot. The robot has hundreds of screws and most of them belong to this group. Most of the small sized keys will be required, from 1.5 *mm* to 3 *mm*.
- **Phillips screwdriver set:** These screwdrivers are needed to place and remove all phillips type screws present on the robot. Although most of the screws are allen type, some belong to this group.
- **Slotted screwdriver set:** These screwdrivers are needed to place and remove all slotted type screws present on the robot. Although most of the screws are allen type, some belong to this group.
- **Gripping Pliers:** A lot of small parts have to be held in place when placed and these tools could prove useful for this task. Also, they are needed to tighten the knots made to the steel cables. Both straight and curved pliers are needed.
- **Small gripping pliers:** These tools are useful to hold parts in place when they are assembled, and also to put and remove screws from their holes and to correctly accommodate the steel cables in the pulleys. Several different types may be required.
- **Cutting pliers:** These tools are mainly used to cut the steel cables to the desired length. Also they are useful for the electric wiring of the robot. Several different sizes are required depending on the size of the cable to cut.
- **Stripping pliers:** These tool is specifically used to remove the teflon cover of some of the steel cables. These tool is adjustable so it can be used with all the diameters present in the robot.
- **Files:** These tools are mainly used to correct small defects in the different parts of the robot and also to polish the end caps of the cables when building them.
- **10 metric tons press:** This machine is only used to crimp the end caps of the steel cables. To make this caps, it is also necessary the use of a custom made part which can be found at the iCub platform repository [1]. See the picture in Fig. 1.

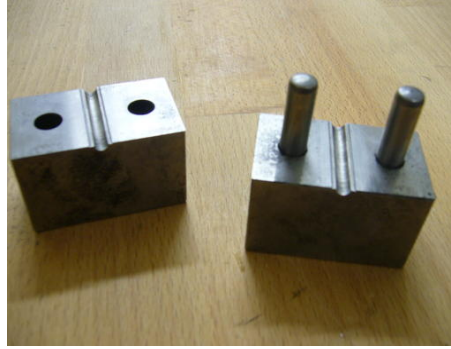


Figure 1: The two temperate steel parts used to crimp the end caps with the steel cables using the press. These two parts must be properly aligned using the marks in one of their sides (0).

- **Glue:** glue is used in several places, to keep in place the pins used as motion limits on many joints and also to strengthen the knots at the end of some steel cables.

A part from the tools themselves, it will be also necessary to have a complete set of screws of all sizes and types required and also steel cable with the necessary diameters (see the iCub documentation for more detailed list of required screws and steel cables [1]).

1.2 How to build a cable

The process described in this section explain how to build a replacement of a steel cable of the robot. The process itself is pretty easy and can be completed in a few minutes. In general the difficult part is actually replacing it when the robot is fully assembled.

This procedure applies to all cables (wrist, knee, ankle, shoulder, etc ...) except to those of the hand and forearm, which are made in a different way, as explained in section 1.4. All the cables covered in this section have a fixed length which is stated in the iCub platform documentation [1], and that must be reproduced with the maximum accuracy to avoid assembly problems.

All cables covered in this section have a teflon cover to reduce the friction when they move along the pulleys of the robot. In some places it is necessary to remove the teflon cover surrounding the steel cable. To do that, the stripping pliers must be adjusted to the actual diameter of the steel cable (a drill bit of the proper diameter may be used). Two different types of steel cables are used:

- 1.5 mm diameter with a teflon cover which gives a total diameter of 1.75 mm.
- 1.2 mm diameter with a teflon cover which gives a total diameter of 1.5 mm.

To avoid having to measure the length of all the cables each time, it is recommended to build a ruler with marks for all the lengths together with the corresponding identification number. Also, this ruler could have a small part to hold the steel cable in place while measuring it. An example of such ruler is shown in Fig. 2, but there exist no design to reproduce it in the iCub platform documentation.

To build a new robot cable follow the following steps in order:

1. Cut the steel cable a little longer than the required length to allow for possible errors to happen.



Figure 2: Ruler with marks for all the lengths of the steel cables used to avoid having to measure them each time.

2. With the stripping pliers, remove the teflon cover from only one side of the steel cable. Remember to adjust the tool to the required diameter before.
3. Place an end cap to the unprotected end (see Fig.3(a)), and crimp it with the press. Pay attention to the direction in which the cap is inserted into the cable because one of its sides has a little wider hole to facilitate the insertion of the cable (see Fig. 3(b)).



(a) Example of one side of the steel cable without the teflon cover and the end cap put in place.



(b) Detail of the two sides of the end caps used.

Figure 3: Details of the end caps in b) and how it should be inserted into the cable to crimp it properly in a).

4. Remove the exceeding cable and the small defects from the cap with the files. In some of the cables (marked with numbers 7 to 10 it will also be necessary to shorten the end caps to a length of 4 mm. A saw or lathe may be used.
5. Measure the length of the required cable or place the cable in the ruler shown in Fig. 2 and mark the appropriate position.
6. Again, remove the teflon cover and crimp an end cap to the other side of the steel cable. Also remove the exceeding steel cable and the small defects with the files.
7. Check the actual length of the cable with the ruler. Discard the steel cable if it is too long or too short ($\pm 1\text{ mm}$) and repeat the procedure from step 1.
8. Otherwise, that's it!!

On one of the ends where the steel cables are attached to the robot structure, there is a screw to adjust their tension. This allows keeping the tension when the steel cables increase their length, in the long term, and also to allow for small inaccuracies in their length.

In the second case, eventhough small errors can be corrected, a cable too long or too short may cause problems because it could be impossible to get the necessary tension for the proper

operation of the robot. So, it is recommended to be as precise as possible when building the replacements of the steel cables for the robot.

1.3 Shoulder and elbow

A picture of the shoulder fully assembled, but without any electronics, is shown in Fig. 4. It has 3 motors but only two independent controllable degrees of freedom because two of the motors are coupled to emulate the real motion of the human body.



Figure 4: Picture of the shoulder of the icub robot fully assembled without the electronics.

This part houses 3 brushless motors, one for each degree of freedom. Each motor actuate two different cables, one to pull in one direction on the other one to pull in the other direction, thus achieving high strength in both directions. The transmission of the motion from the motor to the actual joint is carried out through pulleys where the cables are attached.

At one end, the cable is attached to the pulley by a hole (see Fig. 5(a)), while at the other end, the cap is placed inside a moving part (right most part in Fig. 5(b) that can be adjusted with a screw, and the whole set is attached to the pulley by the left most part shown in Fig. 5(b)). For short cables, when the moving part is small, the end cap have to be shortened to increase the range of adjustment (this is the case of cables 7 to 10).

Both the moving part and the pulley have a hole to insert the cable, an it is better, although not imperative, that both holes end up at opposite sides of the cable as shown in Fig. 6(a). This way the cable is more firmly attached to the structure and it is harder for it to detach. The picture in Fig. 6(b) shows the other option which is not recommended. It is also important to place the end cap of the cable as deep as possible into the pulley's hole to avoid it jumping out of position when tension is applied.

The motors used are standard commercial ones (see the icub platform documentation for the specific reference), but one of its main features is that they came with no housing, so the shoulder structure itself serve as housing for them. The stator (Fig. 7(a) and the rotor (Fig. 7(b)) of the motor are shown in Fig. 7.

Another feature of the shoulder is that it uses harmonic gears in each motor, which, although expensive, have no backlash which would decrease the final performance of the robot. The ones used in this case (shown in Fig. 8) achieve a gear ratio of 100 with a reduced size.

All the motors used for the shoulder and elbow use external magnetic encoders, which requires

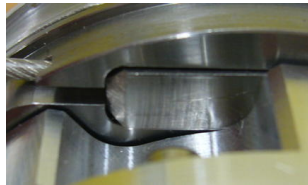


(a) Hole in the pulley with the shape of the end cap of the cable to attach it to the structure.

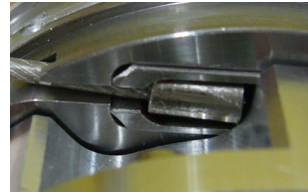


(b) Moving part where the other end of the cable is attached to adjust the tension of the cable.

Figure 5: Cable attachment system of the pulleys.

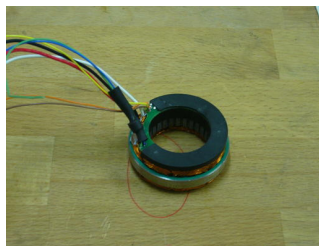


(a) Recommended position of any moving part with respect to the pulley.



(b) The other possible position of the moving part which is not recommended.

Figure 6: Possible positions of any moving part with respect to the pulley. a) shows the recommended position and b) shows the other option.



(a) Stator of the motor which is attached to the robot's structure and has all the control electronics embedded, except for the encoder.



(b) Rotor of the motor which is attached to the harmonic gears to drive the cables.

Figure 7: Details of the stator (a) and rotor (b) of the motors used for the degrees of freedom of the shoulder.



Figure 8: Detail of the three parts that make the harmonic gear used in the shoulder joints.

a magnet to be placed on the rotor of the motor. Because of the magnetic interferences of the motor when it is moving, it is necessary to add an iron plate between the motor and the encoder to avoid erroneous readings of the joint position. This plate is not present by default, and it needs to be build and placed after receiving the robot.

Attached to the shoulder there is the structure that provides the elbow motion, which is shown in Fig. 9. The elbow is also driven by a brushless motor with harmonic gears, and follows the same principle than the shoulder: it drives two cables, one to pull from each direction.

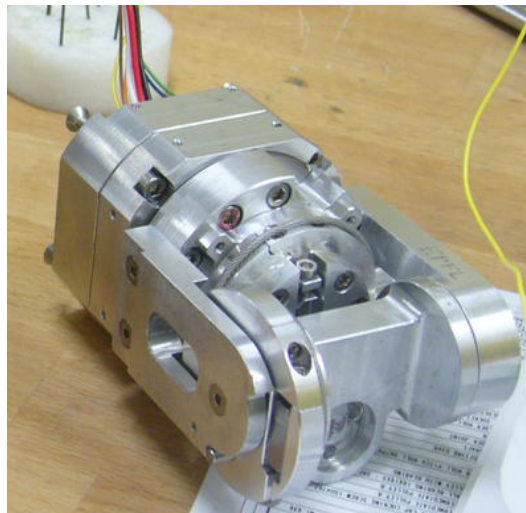


Figure 9: Structure of the elbow of the robot arm before being assembled.

1.3.1 Replacing cables 9 and 10

Cables 9 and 10 control the motion of the elbow of the arm of the robot. These steel cables are thinner and more likely to break than the other cables in the shoulder structure, except for cables 7 and 8 which have the same diameter and are as fragile as these. The structure of the elbow is pretty independent from the rest of the arm, so it is possible to change these cables without having to disassemble any part of the robot.

In Fig. 10 there are two views of the elbow part completely assembled with the steel cables in place. These two pictures can be used as reference of how the steel cables of the elbow must be assembled. The procedure to replace these cables is as follows:

1. Detach both cables, even if only one of them is broken, by releasing the screws to adjust

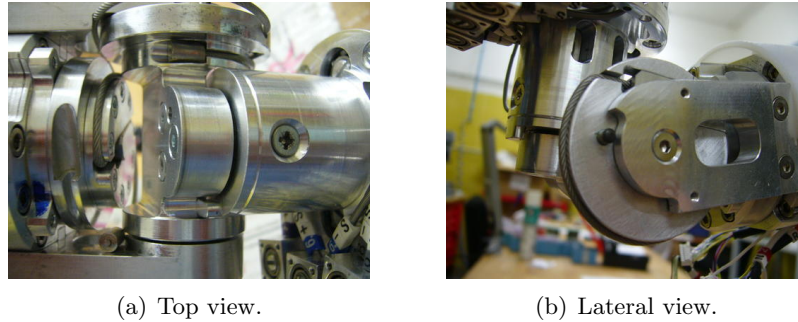


Figure 10: Two detailed view of the elbow fully assembled and with all wires in place.

the tension of the cable and removing the moving part. At this point the joint will be free to move.

2. Place the short caps of the new cables in the moving part and place them inside the pulley. At this point the screw to adjust the tension should only be inserted a few threads to facilitate the assembly process.
3. Place the other end of cable 9 in the corresponding hole. See the iCub platform documentation [1] for detailed drawings of the cable placement. This should be easy since the other cable is not yet attached to the structure.
4. Place the other end of cable 10 in the corresponding hole. If the lengths of both cables are right, a little force should be applied to put this second cable in place since the other cable is already attached. If the cables are too short, it would be impossible to fulfill this step, and the cable must be rebuilt. In this case follow the procedure detailed in section 1.2 and start over from step 1.
5. Increase the tension of both cables using the screws in order to remove any backlash that may exist in the joint. If the cables are too long, it would be impossible to achieve enough tension, and the cables must be also rebuilt. In this case follow the procedure detailed in section 1.2 and start over from step 1.

This task is easy to perform when the elbow part is alone and without electronics. However the complexity increases when the robot is fully assembled. As a general rule, for both the arms and the legs, if one cable brakes in one side, take a look to the other side for reference on how to replace it.

1.3.2 Replacing cables 5 to 8

In general cables 7 and 8 are the most likely to break because of its reduced section and its low radius of turn, so they are also the most likely to need to be replaced. In order to replace them, it is also necessary to remove cables 5 and 6, even if they are not damaged. In this section, the procedure to replace cables from 5 to 8 is detailed.

In Fig. 11 there are two views of the part of the shoulder related to cables 7 and 8 completely assembled with the steel cables in place. These two pictures can be used as reference on how the shoulder must be assembled. The procedure to change these cables is as follows:

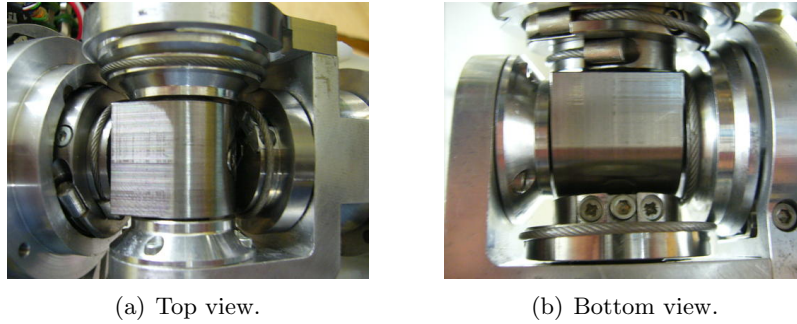


Figure 11: Two detailed view of the part of the shoulder related to cables 7 and 8 fully assembled and with all wires in place.

1. Remove all the steel cables, even if they are not damaged, by releasing the screws to adjust their tension and removing the moving part. At this point the joint will be free to move.
2. Insert the short end cap of cables 7 and 8 into the moving part and attach the whole set to the corresponding pulley (see the documentation for the iCub platform for the drawings [1]).
3. To assemble the cables in the right position it is necessary first to partially disassemble the shoulder structure. To do that it is necessary to first remove the part shown in Fig. 12(a) and then the one shown in Fig. 12(b), which is behind the first one.

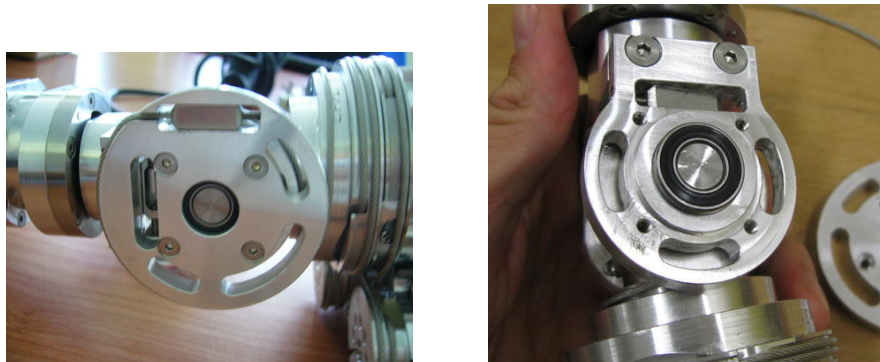


Figure 12: A detail of the two parts that need to be removed in order to disassemble the shoulder structure and put cables 7 and 8 in place.

4. At this point it is possible to move apart the pulleys so that the cable may be placed in the right position. After placing them, the two parts removed in the previous step need to be reassembled to keep the cables in place.
5. Place the long end cap of cable 8 in the hole of the corresponding pulley. See the iCub platform documentation [1] for detailed drawings of the cable placement. This should be easy since the other cable is not yet attached to the structure.

6. Place the long end of cable 7 in the hole of the corresponding pulley. If the lengths of both cables are right, a little force should be applied to put this second cable in place since the other cable is already attached. If the cables are too short, it would be impossible to fulfill this step, and the cable must be rebuilt. In this case follow the procedure detailed in section 1.2 and start over from step 1.
7. Increase the tension of both cables using the screws in order to remove any backlash that may exist in the joint. If the cables are too long, it would be impossible to achieve enough tension, and the cables must be also rebuilt. In this case follow the procedure detailed in section 1.2 and start over from step 1.
8. Assemble cables 5 and 6 following the same procedure described in this section, steps 5 through 7.

1.3.3 Replacing cables 1 to 4

These cables are not likely to brake in normal circumstances, however, the procedure to replace them is a little tricky, so it is detailed in this section.

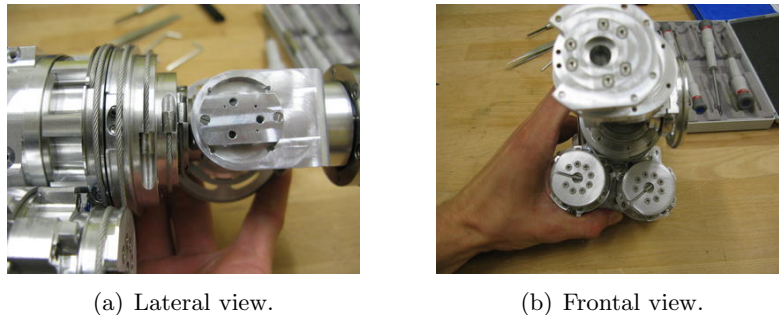
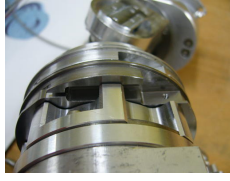


Figure 13: Two detailed views of the part of the shoulder related to cables 1 through 4 fully assembled and with all steel cables in place.

In Fig. 13 there are two views of the part of the shoulder related to cables 1 through 4 completely assembled with the steel cables in place. These two pictures can be used as reference on how the shoulder must be assembled. The procedure to change these cables is as follows:

1. Release the tension on cables 5 and 6 and remove its end caps from the holes of the pulley, but do not completely remove them. This step is necessary because cables 5 and 6 constraint the motion of some of the parts.
2. Detach all 4 cables, even if only one of them is broken, by releasing the screws to adjust the tension of the cable and removing the moving part. At this point all associated joints will be free to move.
3. In order to both remove and install cable 1, it is necessary to place the left most pulley from Fig. 13(a) as shown in Fig. 14(a). When the end cap of the cable is in place, attach it to the pulley through the moving part to adjust the tension.
4. Cables 2 and 3 need to be placed and removed at the same time before being attached to the corresponding pulleys. Otherwise it would be impossible to assemble or disassemble cable 3. The two pulleys of the shoulder must be places as shown in Fig. 14(b). When



(a) Position of the left most pulley in Fig. 13(a) with respect to the shoulder structure.



(b) Relative position of the two pulleys of the shoulder to place cables 2 and 3.



(c) Detail of the special shape of the third pulley from the left in Fig. 13(a) to insert cable 4.

Figure 14: Three details of the positions of the pulleys of the shoulder in order to be able to assemble cables 1 through 4.

the end caps of both cables are in place, attach them to the pulleys through the moving parts to adjust the tension.

5. To be able to place and remove cable 4, the third pulley from the left in Fig. 13(a) has a special shape as shown in Fig. 14(c). When the end cap of the cable is in place, attach it to the pulley through the moving part to adjust the tension.
6. Attach the other end of cable 1 to the lateral hole of the right most small pulley shown in Fig. 13(b). When the cable goes from one pulley to the other, pay attention to keep the cable straight.
7. Place the plastic part shown in Fig. 15 in the end cap hole of the right most small pulley shown in Fig. 13(b) to prevent cable 2 from getting into the hole and therefore loose the required tension.



Figure 15: Plastic part used to fill the hole for the end caps of cables 1 and 3 to prevent cables 2 and 4 from getting into them.

8. Attach the other end of cable 2 to the top hole of the right most small pulley shown in Fig. 13(b). When the cable goes from one pulley to the other, pay attention to keep the cable straight.
9. Attach the other end of cable 3 to the lateral holes of the left most small pulley shown in Fig. 13(b). When the cable goes from one pulley to the other, pay attention to keep the cable straight.

10. Place the plastic part shown in Fig. 15 in the end cap hole of the left most small pulley shown in Fig. 13(b) to prevent cable 4 getting into the hole and therefore loose the required tension.
11. Attach the other end of cable 4 to the top hole of the right most small pulley shown in Fig. 13(b). When the cable goes from one pulley to the other, pay attention to keep the cable straight.
12. Follow steps 5 through 7 from section 1.3.2 to assemble cables 5 and 6 again.

If the assembly is correct, and all the cable lengths are right, all the pulley holes for the four cables must be more or less aligned as shown in Fig. 16.

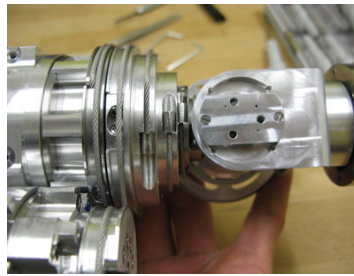


Figure 16: Relative position of all pulleys of the shoulder if all the cables have the correct length and have been assembled right.

1.4 Hand and forearm

The hand and the forearm are the most complex parts of the robot due to its small size and number of degrees of freedom and moving parts. The hand has a total of 9 degrees of freedom (1 shared by the pinky and ring fingers, 2 for each of the middle and index finger, 3 for the thumb and 1 to move all fingers apart), and the wrist has 3 additional degrees of freedom. The set of hand and forearm is shown in Fig. 17.

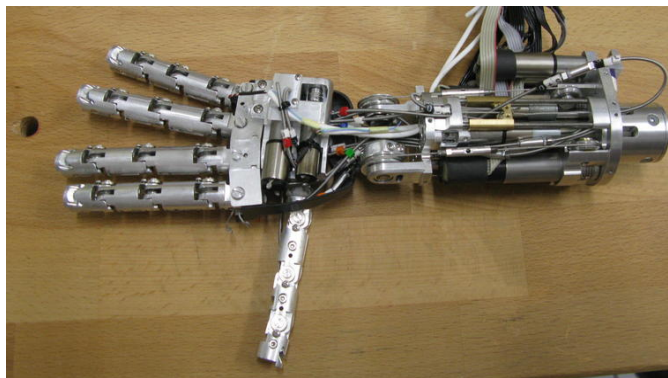


Figure 17: Detail of the hand and forearm fully assembled.

All the cables used to drive all the joints of the hand and forearm have a cover composed of a thin tube of teflon (to reduce friction) and a spring like metallic layer (to help in the transmission of the motion). See Fig. 18(a) for an sketch of the cover. Each cover is attached with glue, at both ends, to the structure of the robot, from the motor to the corresponding joint, and has a

brace in the middle to change the tension of the cable in case it is necessary, which is shown in Fig. 18(b).

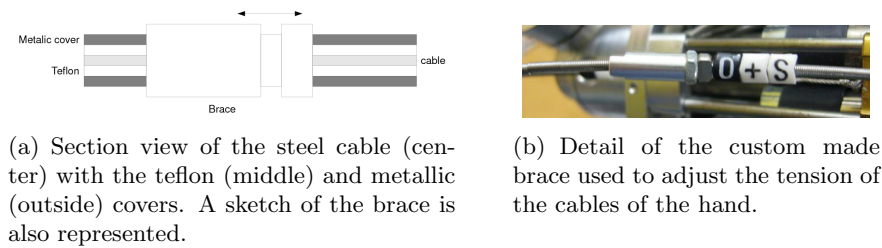


Figure 18: Tension elements for the cables of the hand.

Notice from Fig. 18(b) that all cables and covers are labeled so that they can be easily identified when the robot is fully assembled. As a general rule, for both the arms and the legs, if one cable brakes in one side, take a look to the other side for reference on how to replace it.

1.4.1 Distal joints

The distal joint set includes the last phalange of the middle, index and thumb fingers, and also all the joints of the pinky and ring fingers (these two fingers share a single degree of freedom for all their joints). All these joints are driven by a single cable which pulls the phalange to close it, and when the motor releases it, the phalanges tends to return to their default position (completely extended) due to a spring.

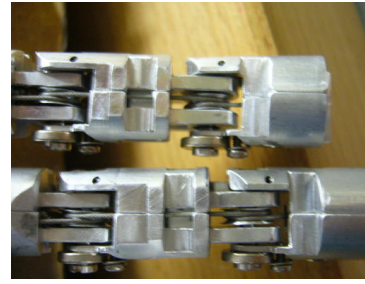
This joints are the most likely to break in the hand, but also, they are the most easy to replace. The steel cable used in these joints also has a teflon cover to reduce friction in the pulleys and its diameter is 0.61 mm and 0.45 mm for the teflon and steel respectively.

The procedure to replace a distal cable is detailed next:

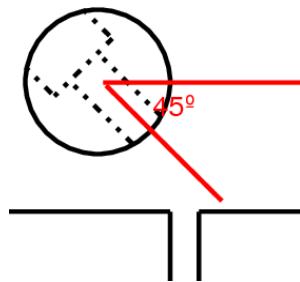
1. Cut a piece of steel cable longer than the required length to account for possible errors.
2. Make a knot in one of the ends of the cable and glue it to improve its mechanical properties. It is important to tight as much as possible the knot using two pliers before applying the glue, as shown in Fig. 19(a).
3. Place the cable into the hole of the pulley of the corresponding motor and pass it through the teflon and metallic cover up to the base of the finger.
4. Place the cable around the pulleys of the other finger joints until the tip of the finger. See Fig. 19(b) for a detail on how the cable must be placed around the pulleys of the other finger joints.
5. With the pulley in the fully extended position (see the sketch in Fig. 19(c)), and with the finger also fully extended, mark the other end of the cable with a pen.
6. With the finger in the folded position (see Fig. 19(d)), make a knot at the marked position and glue it as in step 2.
7. Test the motion of the joint by turning the screw attached to the pulley of the motor with a screwdriver. If the joint does not move, try applying the power supply to the motor because some times the motors get blocked.



(a) Step 2 - Procedure to tighten the knot in the distal cables using two pliers.



(b) Step 4 - Right position of the cable around the pulleys of the other joints of the finger.



(c) Step 5 - Sketch of the position of the pulley of the motor in the fully extended position. Frontal view.



(d) Step 6 - Finger in the folded position to make a knot in the cable of the distal joint.

Figure 19: Details of some of the most important steps to replace distal cables.

8. If the joint has backlash, it means the cable has not enough tension. Try adjusting the brace, but if it is not enough, remove the cable and start over from step 1.

For the distal joints it is important to pay attention to the direction of rotation of the motors. From the fully extended position sketched in Fig. 19(c), the motor should turn so that the cable gets around the pulley (clockwise if the hole is on the left and counter-clockwise if the hole is on the right). Otherwise, the cable will most probably break after a few movements.

1.4.2 Proximal joints

The proximal joint set includes the first joint of both the middle and index fingers and the first two joints of the thumb, as well as the abduction motion. All these joints are driven by two cables, each one to pull the joint in each direction, like the joints in the shoulder presented in section 1.3.

This joint rarely break, but replacing the cables involved may prove much more difficult than for the distal joints. The steel cables used in these joints have a diameter of 0.63 mm and have no teflon cover. The procedure to replace a distal cable is detailed next:

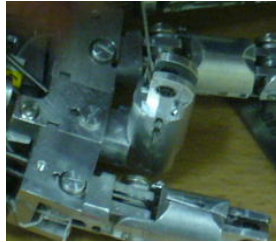
1. Cut a piece of steel cable longer than the required length to account for possible errors.
2. Place the cable around the pulley of the motor as shown in Fig. 20(a), taking care of centering it to have the same distance at both ends. It is important that the cable exit the pulley to enter the holes in the structure in a straight line.



(a) Step 2 - Detail of the cable placed around the pulley of the proximal joints motors.



(b) Step 4 - Position of the proximal joint to make the first knot in the cable.



(c) Step 5 - Position of the proximal joint to make the second knot in the cable. In this case the power supply is applied to the motor

Figure 20: Details of some of the most important steps to replace proximal cables.

3. Pass the cable through the teflon and metallic cover up to the base of the finger.
4. With the motor and pulley in its fully extended position (follow the same criteria used in step 5 of the distal joints procedure) and the finger also in its fully extended position (see Fig. 20(b)), attach the cable to the proximal joint using a knot. Glue it to improve its mechanical properties.
5. Apply the power supply to the motor and move the joint to the folded limit (see Fig. 20(c)). By applying the power supply to the motor, the joint moves a little further than normally and it makes it easy to prestress the cable.
6. Make a knot as close as possible to the proximal joint attachment point and glue it.
7. Test the motion of the joint by turning the screw attached to the pulley of the motor with a screwdriver. If the joint does not move, try applying the power supply to the motor because some times the motor get blocked.
8. If the joint has too much backlash, it means the cable has not enough tension. Try adjusting the brace, but if it is not enough, remove the cable and start over from step 1.

1.4.3 Forearm

One of the 3 motors of the forearm drives the joint directly and does not need any cables. The 2 remaining degrees of freedom (belonging to the wrist) follow the same principle as the proximal joints of the hand. Fig. 21 shows some views of the degrees of freedom of the wrist.

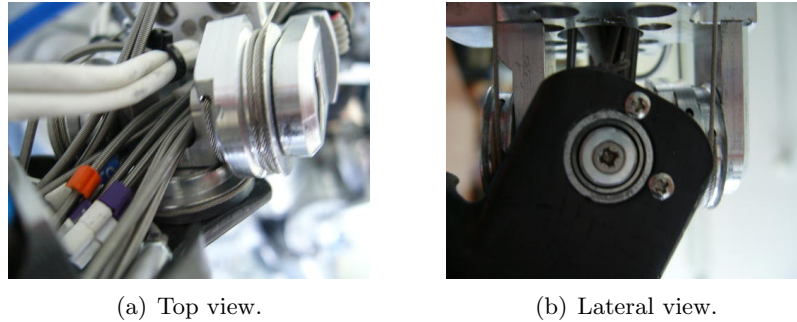


Figure 21: Detailed views of the wrist fully assembled.

2 Electronics

The robot has two independent power supplies: one of 48 V for all brushless motors (although 24 V are used in normal operation) and another one of 12 V for the logic and also for the DC motors. Each motor control board (Brushless or DC) has its own pair of power supply cables coming from the external power supply to reduce the noise.

The brushless motors are controlled by two different boards: the first one with all the logic to read the sensors and close the loop (BLL) and the second one with the power stage for the motors (BLP). The brushless motors have 3 hall effect sensors to measure the position of the rotor in each of the phases and thus be able to actually control the motor, and a hall sensor (AEA board) which reads the absolute position of the shaft of the motor after the gear head to close the position control loop.

The DC motors also require two boards: one to provide all the power supplies needed (MCP) and the second one that provides both the logic to read the sensors and close the loop and the power stage of the motors. All DC motors use incremental encoders which are integrated with the motor except for the abduction motion and the first joint of the thumb. In this two cases, a hall effect sensor is used to measure the absolute position of the joints (the analog signal of the sensor is connected to the MC4 board).

The joints using magnetic encoders do not need any calibration because these sensors provide absolute position. However, the motors using incremental encoders need to be calibrated each time the robot is turned on. In an improved version of the robot, all the joints of the hand will have absolute encoders so it won't be necessary to calibrate the robot each time the power is turned on.

2.1 Electronic boards

All electronic boards used in the iCub platform (except for the PC embedded) are custom made, either because they fulfill a very specific task and no commercial option exist, or because the commercial options did not comply with all the space and functional requirements.

All the boards use small connectors for all cables, which may need to be replaced after some 100 insertions. This is important, because an old connector may cause intermittent errors to occur.

2.1.1 AEA

This board is used to acquire the position information from the magnet placed at the rotor of the brushless motors. This board reads the orientation of the magnetic field generated by the

magnet, digitalizes it and sends it over an SPI protocol to the control board to close the position control loop. This board is shown in Fig. 22.

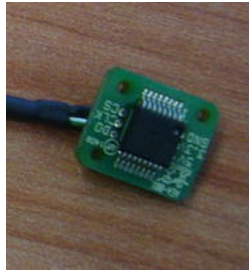


Figure 22: Detail of the board to acquire the position information from the Hall effect sensors.

These boards have three holes on three of its edges. Two of the holes (on opposite sides of the board) are used to screw the board into the structure, and the third one is used to mark the right assembly position.

2.1.2 MCP

This board provides the power supply for up to 4 MC4 boards. These boards use a $2200\mu F$ and $25 V$ capacitor connected to the main power supply in order to reduce noise. Fig. 23 shows a picture of the board.



Figure 23: Detail of the board to provide the power supply up to 4 MC4 boards.

2.1.3 MC4

This board controls up to 4 DC motors, including the acquisition of the encoders and the generation of the control signal to the motor. The main features of this board are:

- Motorola embedded DSP processor.
- External EEPROM memory to store calibration data and the desired CAN bus address.
- JTAG port to download programs (when there is no bootloader)
- CAN bus port to communicate with the board and also to program it (when there is a bootloader).
- Two serial ports formerly used to communicate with the processor but not used any more.

- Two of the incremental encoder inputs are implemented by software.
- Test points and LED's for debugging purposes.
- SPI interface to connect to an absolute Hall effect sensors using an AEA board.
- Current measurement to compute the average power dissipated by the motor.
- Two of the encoders are implemented by software.

These boards have an embedded processor to do all the necessary calculations. In order to develop new control software for these boards, it is necessary to have the C compiler from the manufacturer of the processors (in this case Motorola). Also, there are two possible options to program the processors in these boards: either by using a JTAG emulator port or by use of the CAN bus and a bootloader previously programmed into the processor. By default, all boards come with the bootloader options, which is more flexible.

2.1.4 BLP

This board is the power stage for the brushless motors, and also has the hardware needed for voltage and current monitoring. These boards use a $2200\mu F$ and $63 V$ capacitors connected to the power supply in order to reduce noise. Fig. 24 shows a picture of the board. The main features of this board are:

- Voltage and current measurement.
- Peak overcurrent protection fully implemented by hardware. It disables the motor's power stage without any intervention of the processor, however the processor is notified in such a case.
- Power dissipation protection implemented by hardware and software. The processors monitors both the current and the voltage and estimates the power dissipation, disabling the power stage if necessary.
- Over temperature protection.

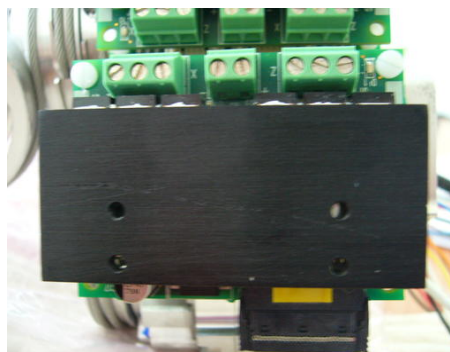


Figure 24: Detail of the power stage board for the brushless motors

2.1.5 BLL

This board controls up to 2 brushless motors, including the acquisition of the encoders and the generation of the control signal to the motor. Fig. 25 shows a picture of the board. The main features of this board are:

- Motorola embedded DSP processor.
- External EEPROM memory to store calibration data and the desired CAN bus address.
- JTAG port to download programs (when there is no bootloader)
- CAN bus port to communicate with the board and also to program it (when there is a bootloader).
- Two serial ports formerly used to communicate with the processor but not used any more.
- It uses both 3.3 V and 5 V power supplies.
- Test points and LED's for debugging purposes.
- SPI interface to connect to an absolute Hall effect sensors using an AEA board.
- It can use either incremental encoders or absolute hall effect encoders.
- Spare analog inputs and general purpose digital input/output (one of each).



Figure 25: Detail of the board to control up to 2 brushless motors.

These boards have an embedded processor to do all the necessary calculations. In order to develop new control software for these boards, it is necessary to have the C compiler from the manufacturer of the processors (in this case Motorola). Also, there are two possible options to program the processors in these boards: either by using a JTAG emulator port or by use of the CAN bus and a bootloader previously programmed into the processor. By default, all boards come with the bootloader options, which is more flexible.

2.1.6 CFW

This board is a standard PCI slave board that is directly connected to the PCI bus of a PC-104 form factor PC embedded commercial board (see the iCub platform documentation for the particular reference of the board and its manufacturer [1]). The main feature of this board are:

- 4 CAN bus controllers handled by the main processor: one for the head and the hip, one for the left arm, one for the right arm and the last one for both legs
- A 2 port firewire controller.
- PCI to local bus interface bridge.

It is important to note that each of the CAN buses have to be terminated with a 120Ω resistor between the differential data signals at both ends to avoid signal bouncing. At the moment, one of the ends is the CFW board, and the other is one of the motor controller boards.

Currently, the 4 CAN buses are almost at the limit of their bandwidth. Also new sensors are expected to be introduced in the near future which have quite high bandwidth requirements. For these reasons, it is being planned to add 6 additional CAN buses, and also to move all the bus handling issues to a dedicated processor to free the main processor.

One important thing about the CAN buses used on the CFW board is that they do not use a standard protocol. The physical layer complies with the CAN specifications, however the upper layer do not in order to use the available bandwidth more efficiently. Therefore, it is not possible to use standard CAN controllers to communicate with the different boards of the robotic platform.

Both the CFW board and the motor control boards are connected to the CAN bus. In general, there is a CAN address for every 2 motors, so each BLL board has a single CAN address, while each MC4 board has two addresses: the one assigned in the configuration file and the next one.

In the wiring spreadsheets there is also information regarding the CAN buses connections. This information includes:

- The number of the CAN bus for each board and also its address on that bus.
- The previous (if any) and next (if any) board on the CAN bus.
- The lengths of the wires for both the previous and next boards.

2.2 Wiring

The robot come completely wired, however due to the use, some wires may break and may need to be replaced. In this section, the detailed procedures to make most of the cables on the robot are presented.

In the iCub platform documentation [1] there is a spreadsheets with a lot of information about all the wiring of the robot. The information for each board and cable provided by this documents includes:

- Board identification. This is the unique identifier of each board.
- Cable identifiers. This is the unique identifier of each cable.
- Type of board. This is one of the names presented in section 2.1.
- The name of the connector where the cable is plugged. This name is the same that is printed in the board.
- The type of cable used and its length.
- Correspondence between each of the pins of the connector and the lead color and function.
- A list of all the materials necessary to build the wire.
- What is connected to the other side of the cable (board, motor or sensor).

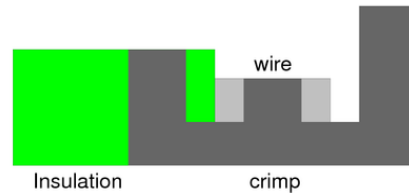
2.2.1 MC4 power supply wiring

These wires connect the MLP board to the MC4 board to provide the necessary power supplies. The MCP boards can have up to 4 such cables connected at any time, and the MC4 boards have only one. These cables provide 12 V, 5 V and 3.3 V power supplies, and use an 8 way connector. The procedure to build such wires is detailed next:

1. Place the wire identifier corresponding to the desired wire from the wiring spreadsheet.
2. Remove around 2 *cm* of the external insulation with a cutter. Pay attention not to cut also the shielding of the cable.
3. Separate the internal wires from the shielding and remove the teflon insulation keeping them together.
4. Only in one side of the wire, completely remove the insulation from the black wire and wrap it with the shielding. This side of the wire will be the one connected to the MCP board.
5. On the other side of the wire, remove the shielding completely. This side of the wire will be the one connected to the MC4 board.
6. Place heat shrinking tube around the black cable and the shielding and apply heat. Leave about 1 *mm* of the wire unprotected to be crimped.
7. Place heat shrinking tube around the whole set of wires at the point where the external insulation was removed and apply heat. This step must be done at both ends of the wire.
8. Remove about 1 *mm* of the insulation of the individual wires to be crimped.
9. Crimp each of the individual wires following these steps:
 - Place the crimp in the crimper as shown in Fig. 26(a).
 - Place the cable inside the crimp as sketched in Fig. 26(b).
 - Press the crimper until it releases to achieve a good fixation. The crimp should look like the one shown in Fig. 26(b) to guarantee an electrical contact and also a good mechanical resistance.
 - For all the individual wires, use small crimps (size 28-32).
 - For the shielding, use a little bigger crimp (size 26-28). Also in this case, due to the size of the wire, it is necessary to put the wire above the crimp before pressing, and let the crimper put it in place.
10. Repeat step 9 for all the individual wires on both side of the cable.
11. Place all the crimped wires inside the connector in the right order. See the corresponding wiring spreadsheet for details.
12. When both connectors are in place, test the electrical connections with a multimeter. If there is any unconnected pin, start over from step 1.
13. If the electrical connections check out, place some glue between the individual wires and the connectors to improve the mechanical resistance of the connector.



(a) Position of the crimp in the crimper.



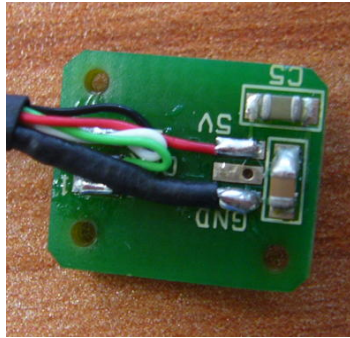
(b) Sketch of the optimal position of the cable inside the crimper.

Figure 26: Detail of the crimp inside the crimper and also the position of the cable inside the crimp.

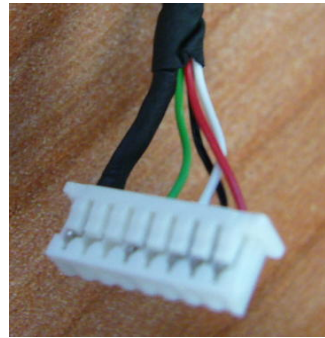
2.2.2 Absolute encoder wiring

These wires connect the AEA sensor board to the motor control board. On one end, these cables are directly soldered on the AEA board and, on the other end, use a 6 way connector to be plugged into the motor control board. The procedure to build such wires is detailed next:

1. Remove around 2 *cm* of the external insulation with a cutter. Pay attention not to cut also the shielding of the cable.
2. Separate the internal wires from the shielding and remove the teflon insulation keeping them together.
3. In this case the shielding is used as a regular wire (GND). Place heat shrinking tube on the shielding such that about 1 – 2 *mm* of the shielding remains visible.
4. Remove about 1 – 2 *mm* of the insulation of each of the individual wires.
5. Put some solder on the exposed parts of the wires and also on the pads of the board with the solder iron.
6. Solder the individual wires in the right pads using the information from the corresponding cabling spreadsheet. All the individual wires must exit the pads from the inner side.
7. Place the wire identifier corresponding to the desired wire from the wiring spreadsheet.
8. Cut all the individual wires to the required length using the information from the wiring spreadsheet.
9. Repeat steps 6 through 10 from section 2.2.1 to crimp all individual wires. Be carefull at this point because there are some pins on the connector that are not used.
10. Test the wire using the multimeter in diode test mode, placing the red lead in contact with the green wire and the black lead in contact with each of the other wires. It should beep momentary.
11. If the electrical connections check out, attach the board to the robot structure. Otherwise check each of the individual wires for a faulty connection.



(a) Detail of the AEA end side of the absolute encoder wire.



(b) Detail of the control board side of the absolute encoder wire.

Figure 27: Details of both ends of the absolute encoder wiring used to get the absolute position of some joints.

If it is necessary to change the exit direction of the cable with respect to the board, be careful with the shielding cable since it is the strongest one and could make the others break when bending. Both ends of the absolute encoder are shown in Fig. 27.

In some places there exist the risk of short circuit due to undesired contacts with the metallic structure of the robot, so it is necessary to put some tape between the board and the mounting point. For example, a joint where it is necessary to put some tape is the one shown in Fig. 16.

2.2.3 Brushless motor wiring

These wires carry the 3 power phases and the 3 hall sensors to measure the position of the rotor, as well as the power supply for the sensors, from the motor to the control boards (BLP and BLL). The phases are directly connected to the BLP board, and the hall sensors use a 6 way connector to be plugged into the BLL board. The procedure to build such wires is detailed next:

1. To avoid interferences of the 3 power phases in the sensor signals and power supply, it is necessary to add a ferrite ring around them.
2. Pass the 3 hall sensor signals wires through the hole of the ferrite ring. The ring should be placed as close as possible to the brushless motor. See Fig. 28 for a detail on the placement of the rings.
3. Pass the 2 sensor power supply wires through the hole of the ferrite ring twice, creating a loop as shown in Fig. 28. The ring should be placed as close as possible to the brushless motor.
4. Attach the ferrite rings to the wires using heat shrinking tube.
5. Put the expandable braided sleeving around both sets of wires, and fix both ends with heat shrinking tube.
6. Put the expandable braided sleeving around the 3 power phase wires of the motor, and fix both ends with heat shrinking tube.

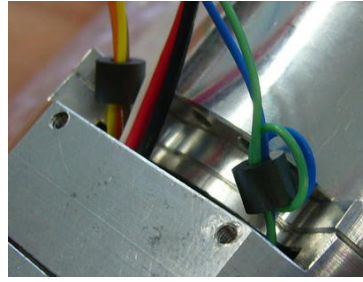


Figure 28: Detail of the placement of the ferrite rings in the hall sensors wires close to the brushless motors.

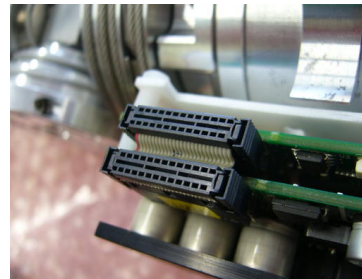
7. Cut all the individual wires to the required length using the information from the wiring spreadsheet.
8. Place the wire identifier corresponding to the desired wire from the wiring spreadsheet.
9. Repeat steps 6 through 10 from section 2.2.1 to crimp all the hall sensor individual wires.

In order to cut the expandable braided sleeving without breaking it, it is necessary to use a heat cutter, which partially melts the edge of the cover. Such a cutter may be build connecting an iron wire into a power supply capable of providing 2 or 3 amperes, thus heating up the wire.

There exist an other cable necessary to connect the BLL and the BLP boards. These cable carries all the necessary control signal from the BLL to the BLP, as well as the necessary feedback signal from the BLP to the BLL boards. A 20 way high density flat wire is used in this case. To crimp the flat cable into the connector it is necessary to use the crimp tool shown in Fig. 29(a).



(a) Crimp tool to build the flat wire between the BLP and the BLL board.



(b) Position of the flat cable between the BLP and the BLL boards.

Figure 29: Details for the assembly and placement of the flat cable between the BLL and BLP boards.

When crimping this flat wire, it is important to pay attention to its right orientation. In some cases, as the one shown in Fig. 29(b), the right orientation of the wire is pretty straight forward. In some other cases, it may be difficult to choose the right orientation because the cable have to be bended several times to go from one board to the other.

2.2.4 External fault wiring

The external fault is used as an emergency stop of all robots. It is composed of a single wire that goes to all the motor control boards and disables their power stages. However, the DSP

remains active and can receive commands and send information back.

A single wire come from the emergency stop button and it is distributed to all boards using connectors like the one shown in Fig. 30 where a single external fault wire is split in several different wires to reach a set of motor control boards in each of the limbs of the robot.



Figure 30: Connector used for the external fault input to split it in several independent wires to reach a set of motor control boards.

3 Software

In this section the main software tools used to develop all the parts of the project are presented. Section 3.1 deals with the mechanical design software used, while section 3.2 deals with the electronic design software. Finally, the tools needed to develop the software and firmware of the platform, as well as a description of its different modules, are presented in section 3.3.

3.1 Mechanical design software

All the mechanical design of the robot has been done using the pro-e CAD program. It is necessary to purchase a license to use this program, but it is highly useful when performing assembly task: it is possible to see the whole robot and how the different parts are placed together, it is also possible to get the unique identifier of each individual part, the type and size of the screws to be used, and many other useful information.

However, this program is only necessary if it is necessary to change the mechanical design by adding, removing or changing some parts of the robot. In most cases, it is only necessary to view the robot assembly, and there exist several free graphical viewers that can be used. Check the iCub platform webpage [1] for a list of these programs.

All the mechanical designs can be downloaded at:

```
cvs -d anonymous@cvs.robotcub.org:/cvsroot/robotcub co iCubPlatform
```

using an anonymous user. In order to get developer access to the repositories, it is necessary to contact the project manager to ask for an account, and then the repository can be accessed using the username and password provided.

3.2 Electronic design tools

All the electronic design have been developed with OrCAD. Although all the schematic designs are available at the iCub project repository, it is necessary to have an OrCAD license in order to edit or modify them.

All the electronic designs can be downloaded at:

```
cvs -d anonymous@cvs.robotcub.org:/cvsroot/robotcub co iCubPlatform
```

using an anonymous user. In order to get developer access to the repositories, it is necessary to contact the project manager to ask for an account, and then the repository can be accessed using the username and password provided.

3.3 Firmware development tools

The low level control software is running on a Motorola DSP. Although the development environment is free, the C-compiler is not, and in order to edit or modify the control software it is necessary to get a license.

The high level software is executed on a linux platform, so all the development tools are free. Both the high and low level software can be downloaded at:

```
cvs -d anonymous@cvs.robotcub.org:/cvsroot/robotcub co iCub
```

using an anonymous user. In order to get developer access to the repositories, it is necessary to contact the project manager to ask for an account, and then the repository can be accessed using the username and password provided.

3.3.1 Low level control loop

The low level control loop is closed in each of the motor control boards, both for the brushless motors and the DC motors. The main tasks performed at this level are listed next:

- **CAN Protocol Interface:** implementation of the protocol layers of the CAN bus which do not comply with CAN-open standard to make a more efficient use of the available bandwidth.
- **Bootloader CAN:** handle the bootloading of new firmware into the board. The processor enters the bootloader state when the power supply is turned on (for 5 to 10 seconds) and also when it is requested by the high level control loop.
- **Position control:** the position control loop is closed at frequency of 1KHz, and it is based on minimum jerk trajectory as shown in Fig. 31.

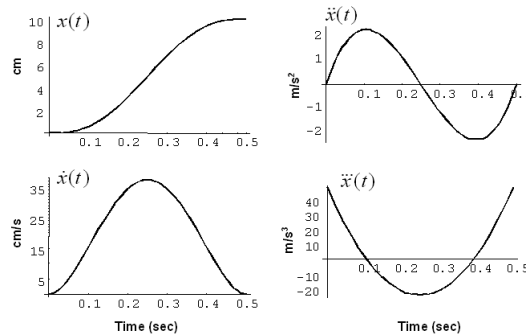


Figure 31: Features of the minimum jerk trajectory used to control the motors.

- **Velocity control:** the velocity control loop is also closed at a frequency of 1KHz.
- **Send feedback information:** it is possible, at configuration time, to choose which kind of information is automatically reported back to the high level control loop. These information can be any combination of the following items:
 - Position feedback.
 - Velocity feedback.
 - PID and errors feedback.

- Status, Faults and debug feedback.

At the low level side of the software, several libraries have been developed to control the peripherals of the DSP (low level CAN, PWM generation, Analog to digital conversions, encoder reading, current monitoring, brushless motor control, etc.) and also some other libraries that handle the high level CAN protocol, the trajectory generation and the PID control.

After a reset, all the motor control boards (both DC and brushless) enter a bootloader stage for a few seconds. During this time it is possible to download new firmware to the boards using the CAN bootloader. Otherwise, the current firmware is executed, configuring first the peripherals and then starting the control loop. It is possible to enter to the bootloader stage at any time sending an special CAN command to download new firmware.

The control loop consists of several well defined steps, listed next:

- **CAN interface:** check for new commands or configuration parameters in the CAN bus.
- **Feedback:** read the encoders and compute the current position and velocity of the motors.
- **compute trajectory:** with the new command and the current state of the motor, compute the next step of the desired trajectory, using a minimum jerk philosophy.
- **compute PID:** with the current error compute the next setup point, taking into account physical limitations of the system.
- **send feedback:** send back to the computer information about the state of the board, the position, velocity or acceleration of the motor, the values of the PID and the PWM, the current value, etc. The set of parameters transmitted to the computer is configured a priori.

3.3.2 High level control loop

The high level control loop is closed in the PC embedded. The main tasks performed at this level are listed next:

- **CAN Protocol Interface:** implementation of the protocol layers of the CAN bus which do not comply with CAN-open standard to make a more efficient use of the available bandwidth.
- **Control Board firmware update:** when the bootloader is present on the motor control boards, this high level software can handle the update of the firmware over the CAN bus.
- **Control Board configuration:** before starting normal operation, the high level software can configure some parameters of the motor control boards.
- **Control commands:** the high level software can send to each of the motor control boards position or velocity set points, but does not have access to the low level control of the motors. The high level control loop is closed at a frequency between 50 *Hz* and 100 *Hz*.
- **Receive feedback information:** it is possible, at configuration time, to choose which kind of information is desired to be automatically reported back. These information can be any combination of the following items:
 - Position feedback.
 - Velocity feedback.

- PID and errors feedback.
- Status, Faults and debug feedback.

This information is reported back to the high level software at the same rate as the high level control loop is closed, which is actually slower than the low level control loop.

The high level control loop is executed on a PC embedded running a Debian distribution. This computer is only responsible of executing the icubinterface, which is the YARP interface of the iCub robot, and all other programs are executed on external computers in the same network.

References

- [1] IIT Genova. Robotcub platform web page, www.robotcub.org/.

Acknowledgements

This work has been partially supported by the EU PACO PLUS project FP6-2004-IST-4-27657 and the Universitat Politècnica de Catalunya (UPC) project Eix-C-2010-”Humanoid Lab”

IRI reports

This report is in the series of IRI technical reports.
All IRI technical reports are available for download at the IRI website
<http://www.iri.upc.edu>.