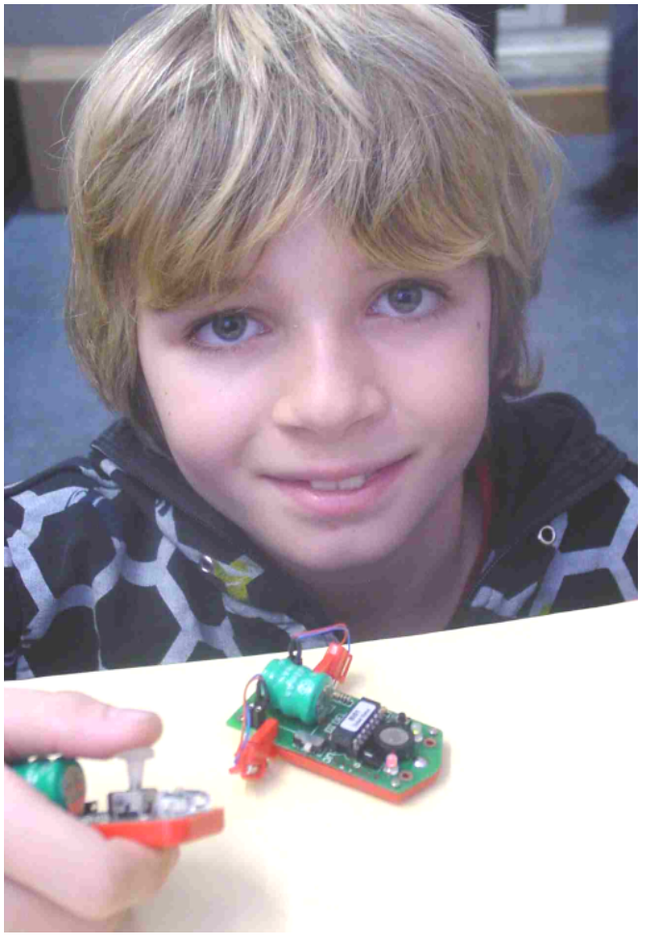
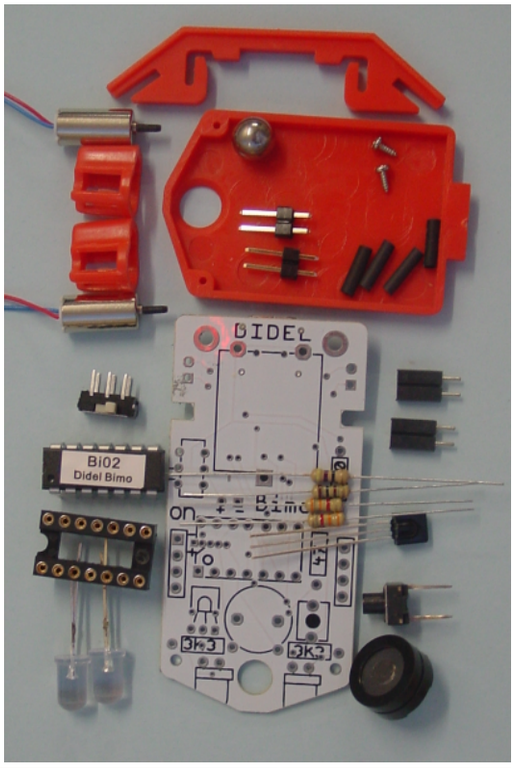
Bimo – a fun Robot kit



The Bimo is a kit with few components to solder. Several SMD components are in place, reducing the work. Kids can play with different motors and LEDs, then solder other kits, learn electronics and microcontrollers.
The objective is to show to young newcomers to robots that it is easy to solder the robot, and then use the board for other applications.
The shaft of the motors carry the «wheels», simple rubber tubes. No gearbox !
No battery to buy, the NiMh is recharged from the USB plug of any PC. The robot runs for about one hour.
Quite fast and nervous, it is interesting to control. Several prerecorded movements can be selected, depressing 1, 2, … 5 times on the push-button. With a PIC programmer, this can be easily modified without knowing about programming (but understanding binary and hexadecimal).
Didel infrared control handle is simple and compact. The power is also provided by a NiMh recharged from a USB port and the battery lasts one hour.

Contents

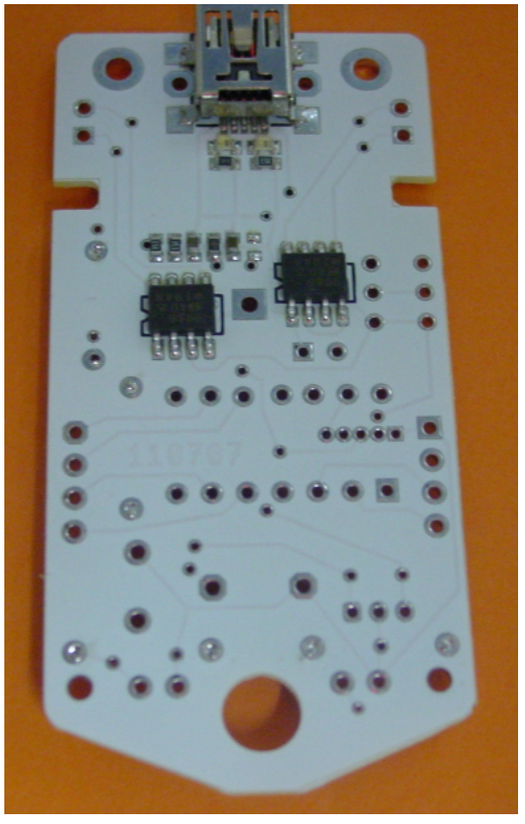
1 board with some soldered components
1 Bag containing components to be soldered on the PCB
1 Bag with the components of the motor-bloc (open later)
1 Accu (to be soldered at last)
1 Infrared control handle (assermbled)
1 Ball
1 Adaptator USB-mini-B



Soldering of the PCB

Some components are already soldered on the board. There are amplis, two LEDs for testing the charge of the accu and some resistors.

Don't solder any component on this side of the board !



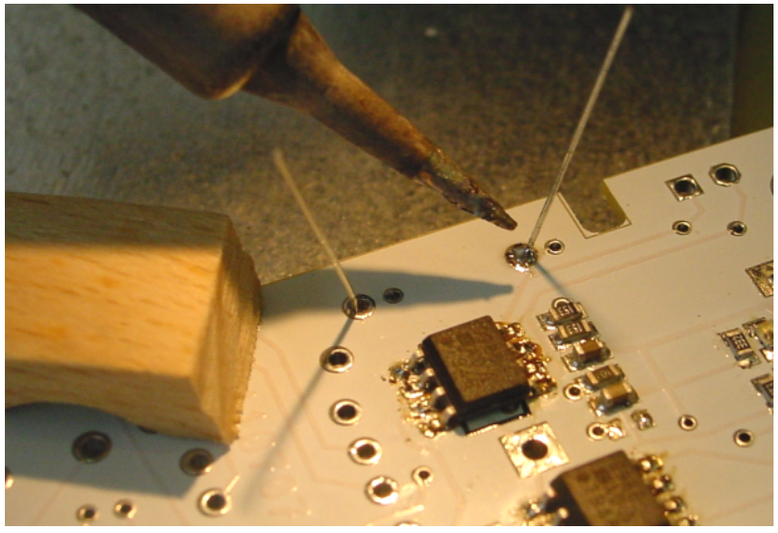
Steps

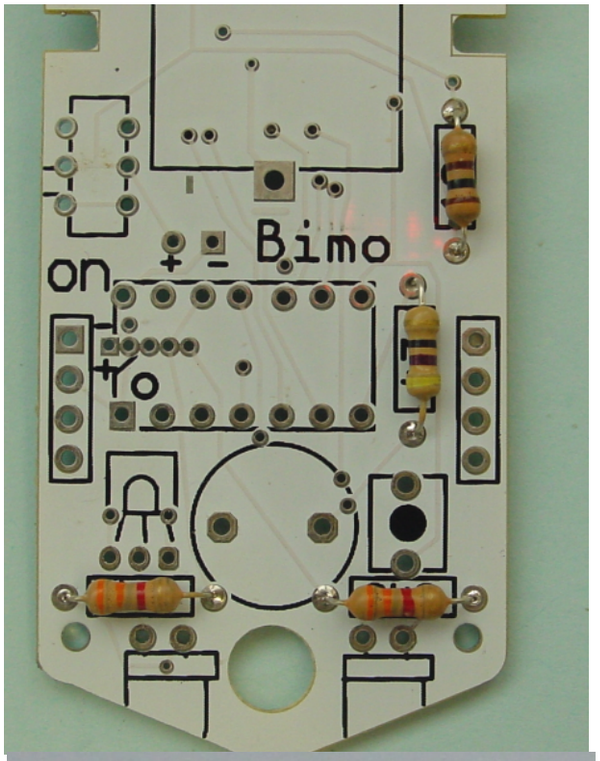
1) Solder the resistors
1x100 Ohm brown-black-brown
1x330 Ohm orange-orange-brown
2x3.3 kOhm orange-orange-red

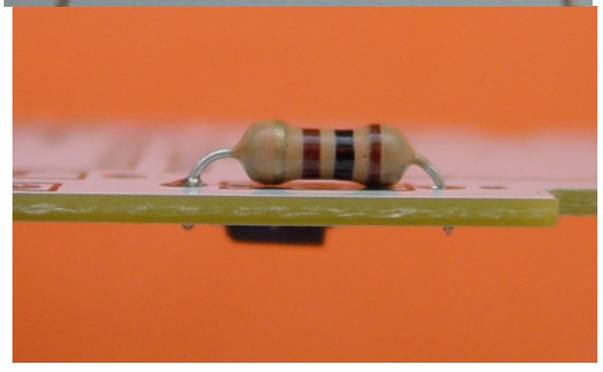
One begins always with the lowest compo-nents (small in hight).

Check colors, fold the wires next to the corps of the resistor, insert, push the wires for maintai-ning the piece. Solder one side, check the re-sistor is. close to the board (not as on the pic-ture) .Push the resistor down and reheat the soldering if necessary. Solder the other side.

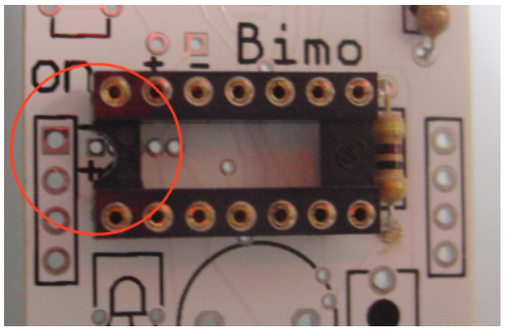
Soldering is easier if the board is fixed. Glue a clothes pin on a wood board or a piece of thick cardboard.

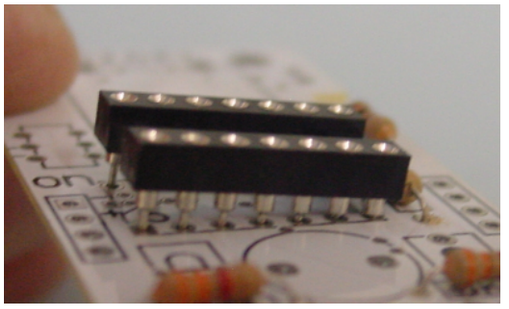




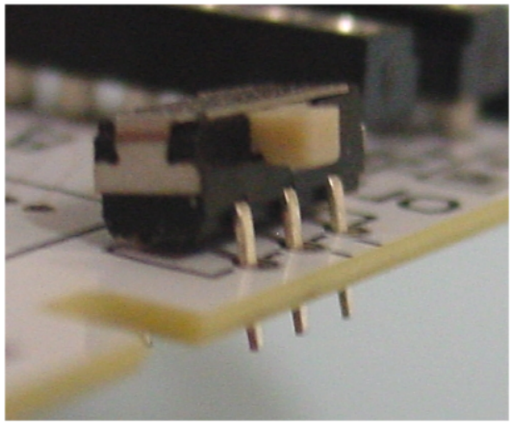


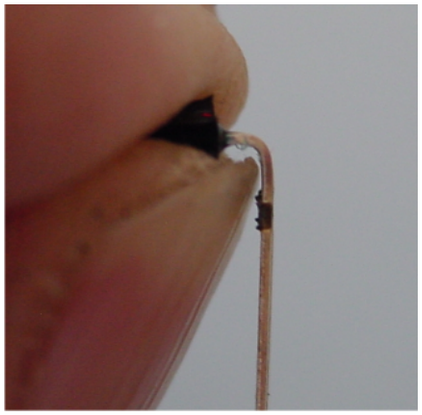
2) Solder the socket
It has 14 pins. Notice the notch on the socket, which has to be at the right. The pins are numbered anti-clockwise around the chip. Pin 1 is marked by a little square on the printed board.
Solder any pin of the middle of the socket. Then check the socket is well inserted, reheat while pressing if necessary, and solder all other pins.

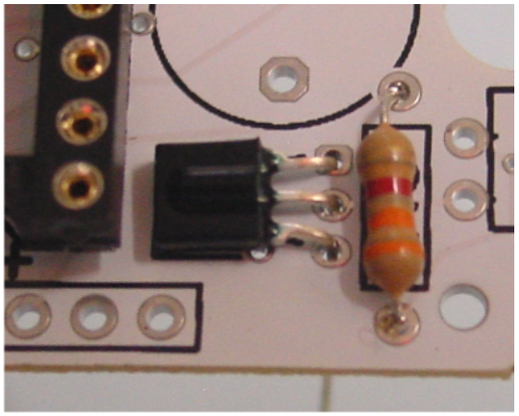




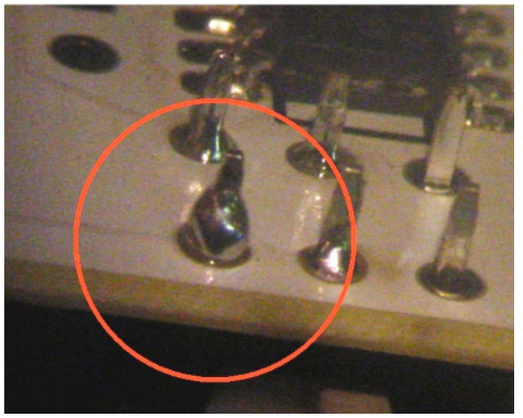
3) Take the 3-pin infrared module.
Bend the pins at 90 degrees, close to the package.
Insert the module and push pins aside in order to maintain the component in place while soldering.





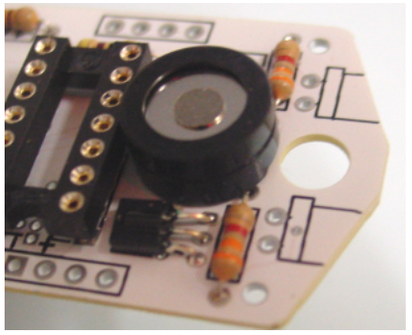


Attention, put very few soldering !
In case of a short-cut, draw the surplus of soldering along the pin.



4) Solder the on-off switch.
Make a red mark to better visualise the «on» position of the switch.

5) Solder the buzzer.
The buzzer is a small speaker.
It has no polarity, despite of the indication +. It is just a coil, which can be inserted in any way.

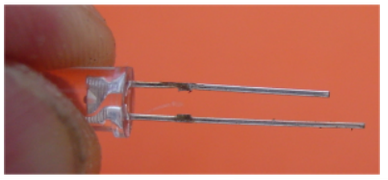


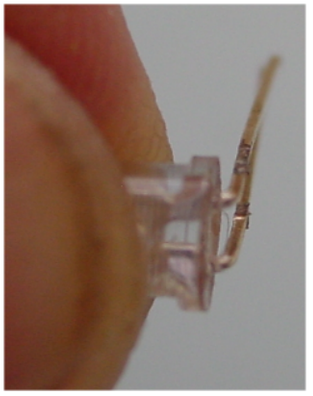
6) Solder the push-button

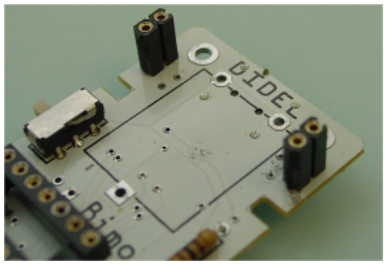
7) Solder the LEDs.
Pay attention to the orientation. The ledge of the case and the short pin are at the left side on the picture.

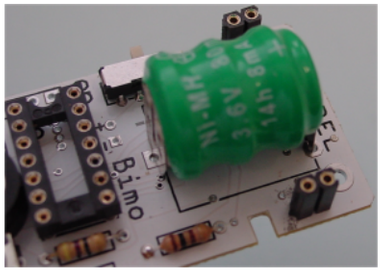
8) Solder connectors for the motors.
The easiest way is to insert both connectors and then return on a piece of wood and solder one pin of each connector. Check for correct insertion, then solder all.

9) Before soldering the accu, be sure the switch is OFF.
Heat enough while soldering the accu.
The two holes above the socket permit to measure the voltage of the accu. They remain empty.
The other empty 6 holes are foreseen for inserting the connector of the Pic programmer.

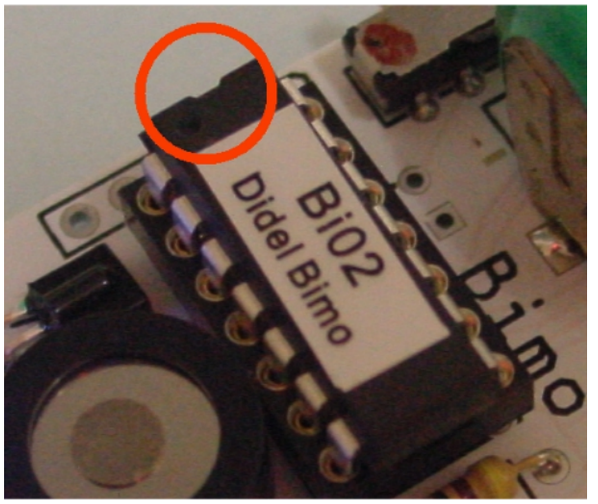






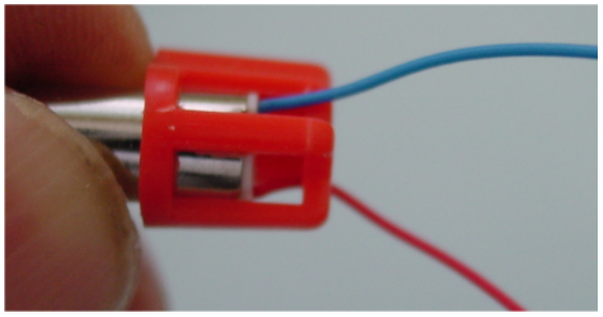


10) Insert the microcontroller.
Pay attention to its orientation. The little mark on the package is pin 1.

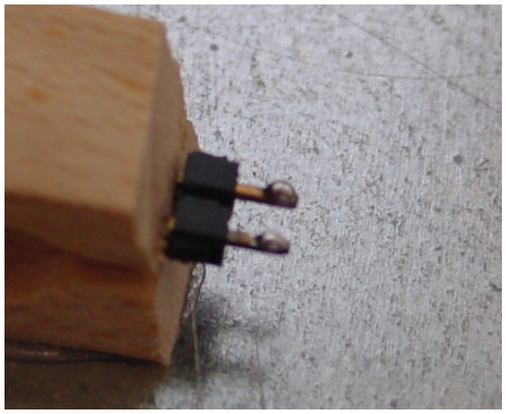


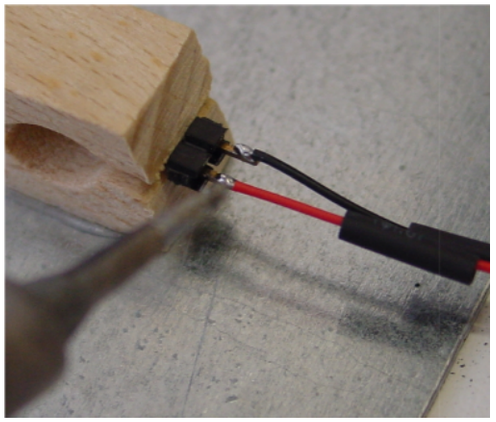
11)Test the board.
Put the switch in position ON.
You should see the LED blinking and hear a sirene.
If not, check the orientation of the micro-controller, reheat the bad solderings, it is probably a problem of bad contact.

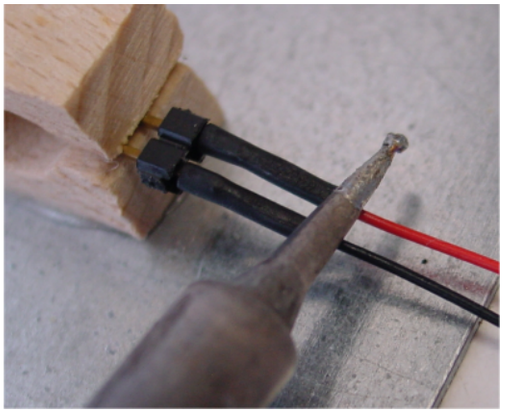
Preparation and assembly of the motors
Solder the connectors of both motors.
Put a little soldering on the edge and dip the wire into that soldering drop. Let cool before moving.



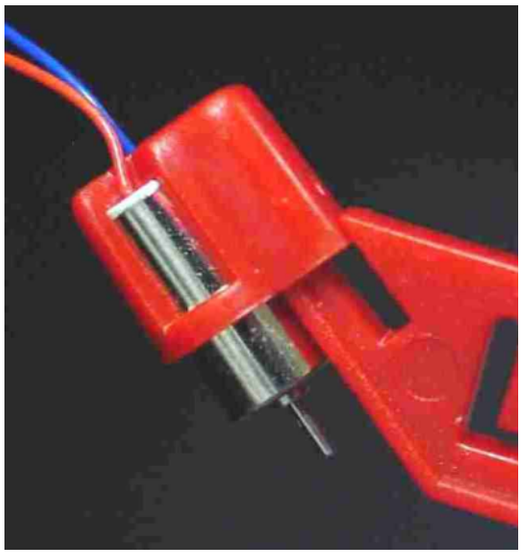
Put the heat-shrinkable tube on the soldering and heat a bit with the cooler part of the iron.





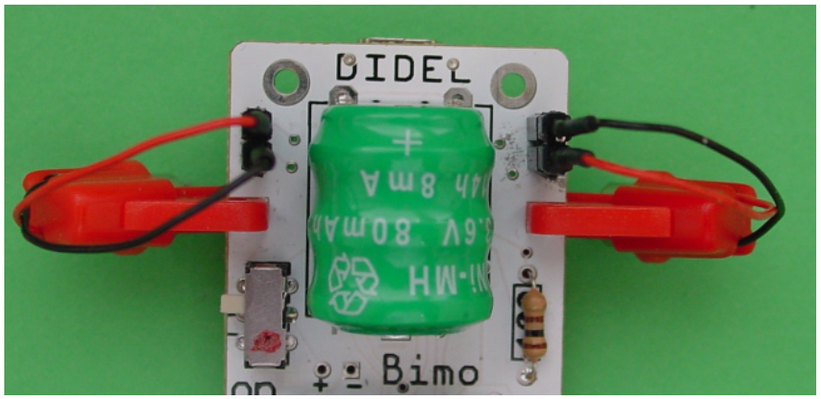


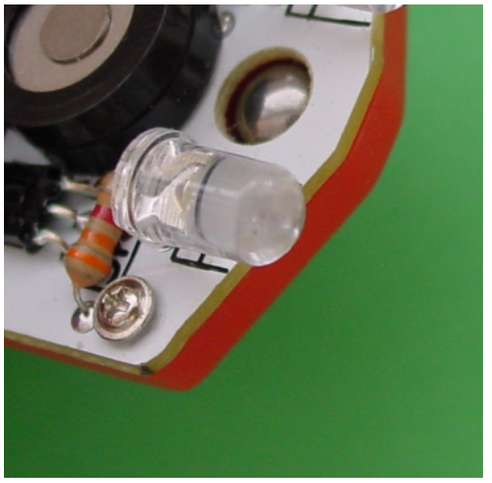
Assembly of the motor bloc



1) Fix the motors on their supports.

2) Clips the bar with the motors and add a little glue if necessary.
   Note the colours of the wires on the right and the left side. If the wires are connected wrong, the control handle will not react correctly.



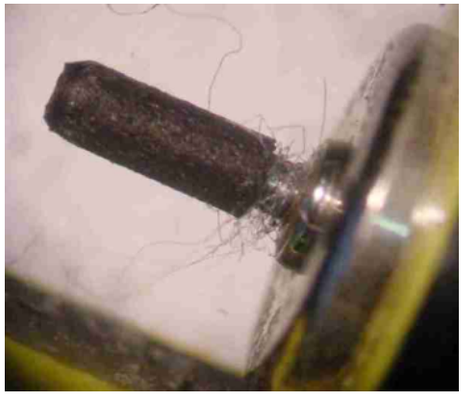


Maintenance

The tubes that serve as tires will be used quickly. Pinch with your nail and pull on the sheath. Pay attention the motor's axe remains always on the edge of the tube.

good

bad



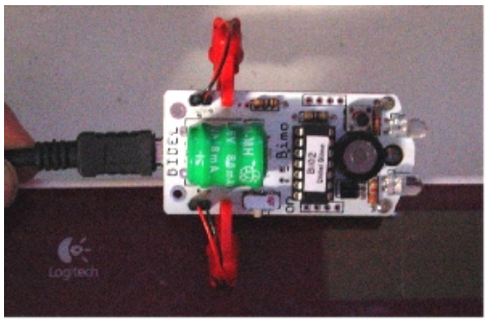
to be cleaned

Recharge of the accu

For recharging the accu, simply connect the Bimo on a USB port of a PC, which has to remain powered. A complete recharge lasts 5-8 hours.
If the PC gets into the sleeping mode, it cuts the current on the USB ports and stops recharging and may even discharge more.

Test and instructions for use

Put the On-Off-switch on position ON. The controller blinks and bips.



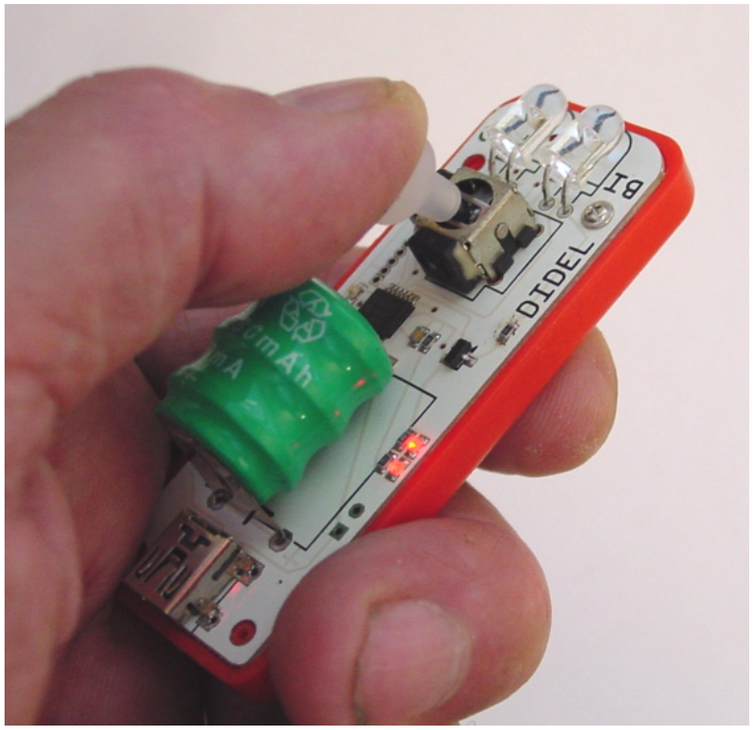
Without control handle

Press 1, 2, 3, 4 or 5 times on the push-button. and lay the Bimo on the floor. After few seconds, the controller plays sirene and starts moving.
1 action : little moves for a demo on a table
2 actions : lines and curves on a larger space.
3 actions : right line during 2 seconds. Who will get the more far ?
4 actions : pirouettes
5 actions or more : sequence which repeats itself in a space of 50 cm at least. Included pauses help saving current.

Control handle

At power-on it is important to leave the joystiock in the middle position (calibrage). The handle waits for an additional action. before emetting.

The control handle emits on 4 different channels. The selection is made by pushing the joystick in one of the four possible directions forward/backward/right/left.
As soon as the joystick returns to the middle position, the emetting starts. The robot synchronizes itself automatically on the selected channel, if it is not already connected on another channel.



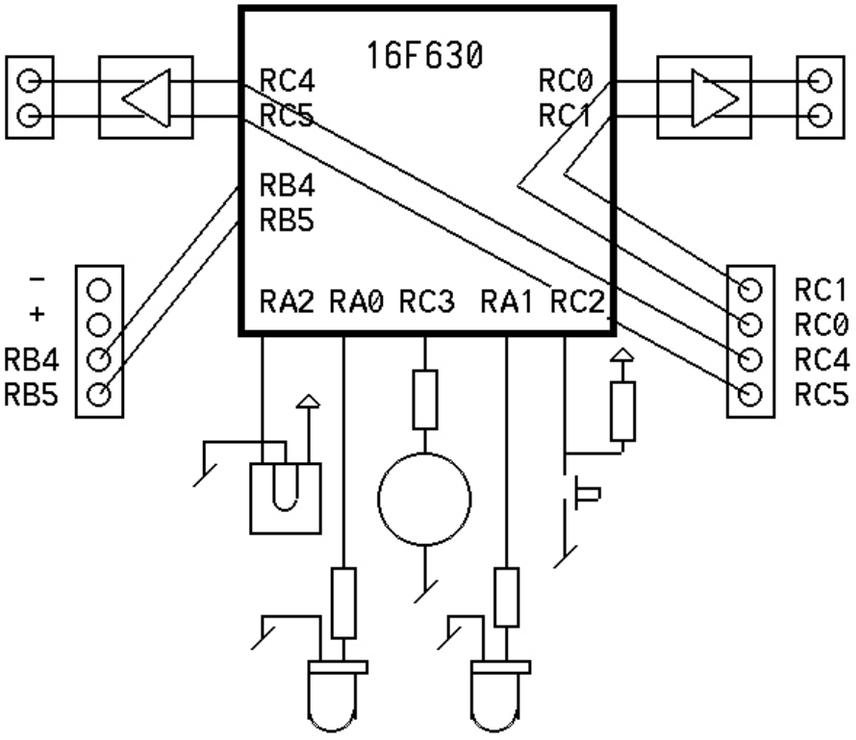
If one presses the joystick more than 2 seconds, till the LED goes off, the control functions are inverted.

You may pilot several Bimos simultaneously, if they use different channels. The first one switches his handle and robot on, and then truns them off. The second one does then the same actions. After the last one also has selected the channel of his Bimo, all handles can be switched on, on the previous selected channels.

The ball

For playing football, one takes the special ball, which is emply and can be opened. When the ball is empty it rolls far away when the Bimo pushes it.Put sugar inside, and the ball will be slower and more funny to play with.
Try various amounts of sugar.
Try also water.

Schematics and explanations



The processor

The microcontroller Microchip 16F630 has a built-in oscillator, which rythmes one million instructions per second. The voltage on the pins 1 et 14 has to be between 2 and 5.5 Volts and the processor consumates less than 1 mA. That is all was is needed, all other is programming, which decides the role of each pin, tests the inputs and assigns the outputs depending on the inputs and the programming sequences.

The LEDs

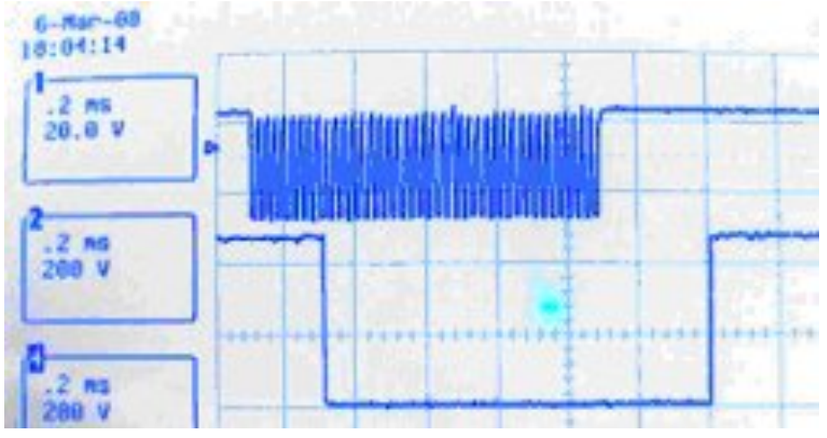
Two bocolor LEDs are connected on the output of the micriocontroller.The current is limitated by 3.3 kOhm resistors. One could use a lower value to be more luminous, but it would consume more. With 3.3k and about 3.3V , the current is 1 mA. If the logic state on the outputs is 00 ou 11, the LED is off. If the state is 10 or 01 the LED is green or red. If one alternes fast, the eye sees the LED yellow. Move the robot while it waits for action on the joystick, and look at the yellow LED on the left.

The speaker
The speaker is a 50 ohm coil, which attracts a membrane. It is connected to pin 7, with the 330 ohm resistor R2 which limits the current. One could put 100 Ohm in order to hear better. The current would be 3,6V/(100+50) = 24 mA, but only half of the time, because of the oscillation. Pay attention to programming mistakes which could maintain the signal active; the current would be maximal and one would hear nothing ! To avoid this problem one could have added a decoupling condensator.

How is the sirene programmed ?
A variable counts the time between changes of state of the speaker. that is the period. Every three impulses in the same period, one decrements this variable, the period diminishes, and the frequency augments..

The push-button
The push-button is connected to a « pull-up » resistor, soldered below.
The resistor imposes the "1* state (>2v) and the button forces the state "0" when pushed. (<0.5v).
As with all mechanic switches, there are contact-bounces, which must be filtered by the processor. Indeed, when the switch is closed or opened, rthere is a bad contact, micro openings and closings during 1 to 5 milliseconds.
The processor could be confused by the number of contacts,that is way the state of the push-buton is read only every 20 ms.

IR receiver
The three-pin IR receiver contains an electronic which transforms the received IR impulses (38 kHz) into an enveloppe whose duration is measured by the processor.



One can see on the scope that this circuit takes 0.2 – 0.3 milliseconds before reacting. That sensitive electronic has to be protected against parasites generated by the motors. The high-frequency impulses are filtered by resistor R4 and condensator C2 (on the face below).

Ampli motors

The motors are controlled by amplis soldered under the printed board.
These circuits can control bigger motors up to 400 mA.
With 3.6V, that means the motor must have more than 10 Ohm resistance.
Our motors have 30 Ohm: if they are bloked, the current will be 3.6V/15Ohm = 240 mAh. Fortunately, the current is much lower when the motors turns and they run not always at full power . Estimate average current is 100 mA.

Accumulator

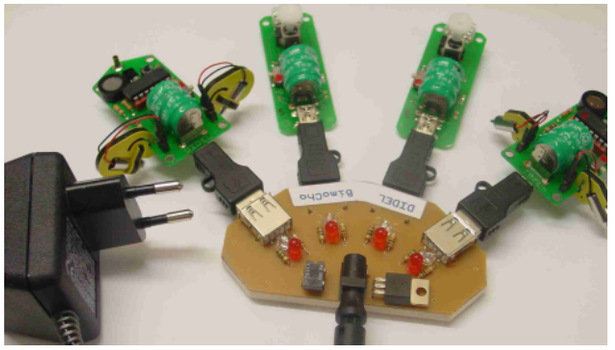
The capacity of the NiMh accumulator is 80 mAh, that means the full loaded accu should last 50 minutes if the current is about 100 mA as evaluated previously.

During discharge, the voltage goes down from 4V to 2,5V, with a long step at 3.7V. On the face below, two LEDs and two resistors indicate the actual voltage; both LEDs are ON of the accu is well loaded. If the voltage in only 3.3 V, one of the LEDs is OFF. The same indicator exists on the control handle.

The solution used by the Bimo for recharging the accu is very simple, but it is rather slow ! A resistor limits the charge current to one tenth of the capacity value, that is to 8 mA. Don't forget to remove the accu when the charge is completed (after 8 hours).
The quick chargers have an electronic which measures the charge and turn off. The USB connector generates 5V, the voltage of the accu at mid charge is 4.2 V, that means you have to add in serial a resistor of 0.8V/8mA = 100 Ohm.

Other motors and "Crane" mode

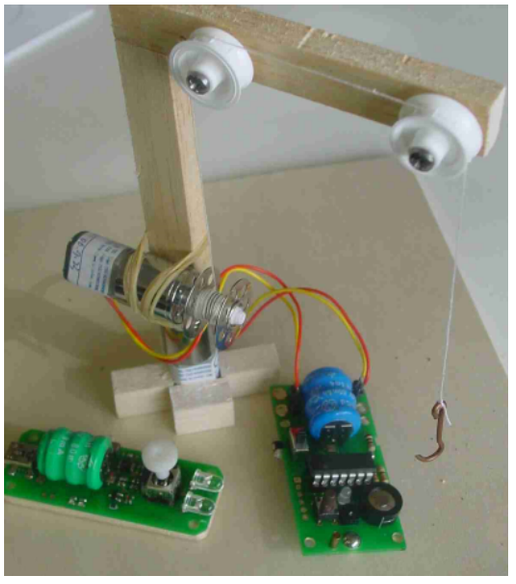
The Bimo board can pilot other robots. or mechanisms.
The amplis can control up to 400mA, but the accus are too small for as much current and LEGO motors, for instance, need 6 V .
The accus of the Bimo are convenient for solar motors (solar-cells), low consuption and good efficiency.



Two control modes have been planned.

In the mode "2-wheels-robot", both motors must run when joystick is pushed.

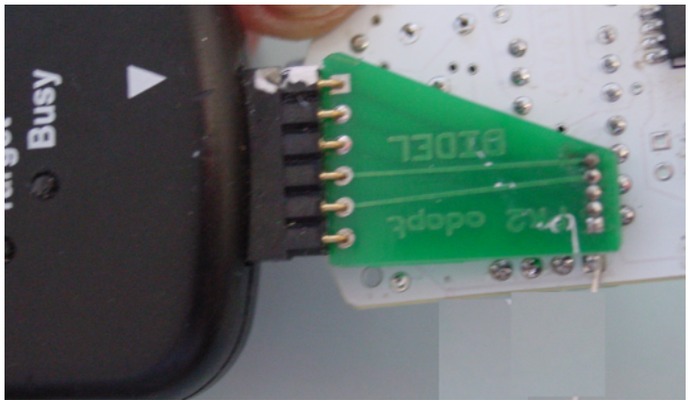
In the mode *crane* one needs independant motors, one controlling the forward backward move, and the other the left-right move.
To select this crane mode, let the joystick pressed for 2 seconds at power on of the handle, till you hear a sound (which is different).



Try to control the Bimo ! You can turn the handle at 45 degrees toward the trajectory, but the speed tables are different: faster and more difficult to pilot.

Turbo mode
This mode allows to go faster (but the Bimo may get slipping). It is selected by maintaining the joystick pushed for about 5 seconds at power on. till one hears a second sirene. The Bimo is then ready to move in turbo mode.

Programming the microcontroller
With a Pickit2 or another Pic programmer, one can modify the program or even rewrite it completely in different programming langages.



The 5 displacement demos are in a file that can be modified without understanding programmation: each move and corresponding duration are written in a table.

Another table defines correspondance between the joystick position abd the speed of the motors. You want your Bimo to be more precise in little moves, or more nervous ? Modify that table, but it is more tricky to understand.

And next...

You have a Bimo and its control handle.

Try to create your own original toy, which can roll, or float.
It may be little or large, amazing, surprising !

Tell us about your experiments and realisations.
We hope you enjoy your Bimo !