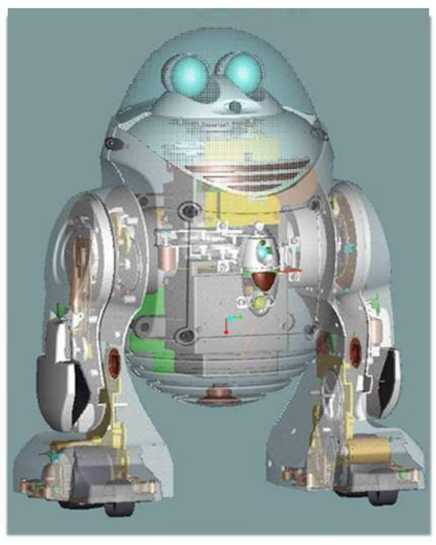
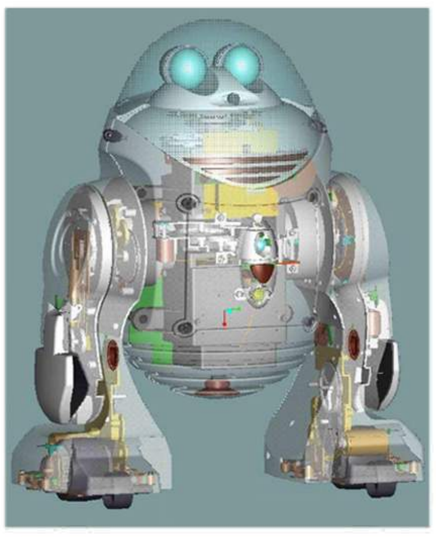


RoboScout Internals

Contents



>> Destination Earth <<

"Greetings Earthling!"

One morning six non-human visitors appeared in my living room.

Where did they come from? What did they want?

"Assistance Please!"

Their journey here was perilous, it seems. One has broken wheels...

...while another could not be revived from his state of suspended animation.

"Assistance Please!"

Who could resist these charming creatures? Alberto and I become acquainted with the visitors before preparing for surgery.

RoboScout Internals: Preliminaries:

Introduction

When I first saw a RoboScout at our San Diego Robotics Society meeting, I *knew* I had to have one. They're delightful, entertaining, and to an aspiring robot builder like me, a wonderful source of inspiration. When we looked inside, I knew I had to have at least *two*: in addition to RoboScout's other charms, he appears to be very hackable!

Before long, I'd talked myself into buying six of them! When two arrived damaged, that was all the excuse I needed to tear one apart to "see how it works." In addition to being a fun thing to do, I wanted to photograph what's inside RoboScout and to encourage other potential hackers by documenting as much as I could along the way. My good friend Alberto Gobbi was just as enthusiastic about the whole project as I, and between the two of us, we managed to take quite a few photos of internal mechanisms along with detailed notes while reducing one RoboScout to a pile of parts.

I hope these pages will be useful to fellow RoboScout hackers. I do add to them from time to time, so check back for updates. Please do tell me about any errors you find. If you have additional information or experiences that would be of value to RoboScout hackers, I would be very glad to include that information in these pages.

Thanks to Alberto for working with me to dismantle a RoboScout and create useful documentation of the process. We had a lot of fun doing this!

Complete Disassembly

The RoboScout we dismantled was Serial Number 4891. We found the Assembly Diagram very helpful. I'll give the page references to it wherever that seems helpful.

If you intend to reassemble the RoboScout, you'll want to keep track of all the electrical connectors that plug into the main board. Fortunately, this was very easy to do. In the unit we dismantled, The board has connector IDs printed on it and the connectors themselves had these IDs written on them in indelible ink. Of course, you can always diagram the board and mark the connectors yourself if these IDs are missing or unreadable.

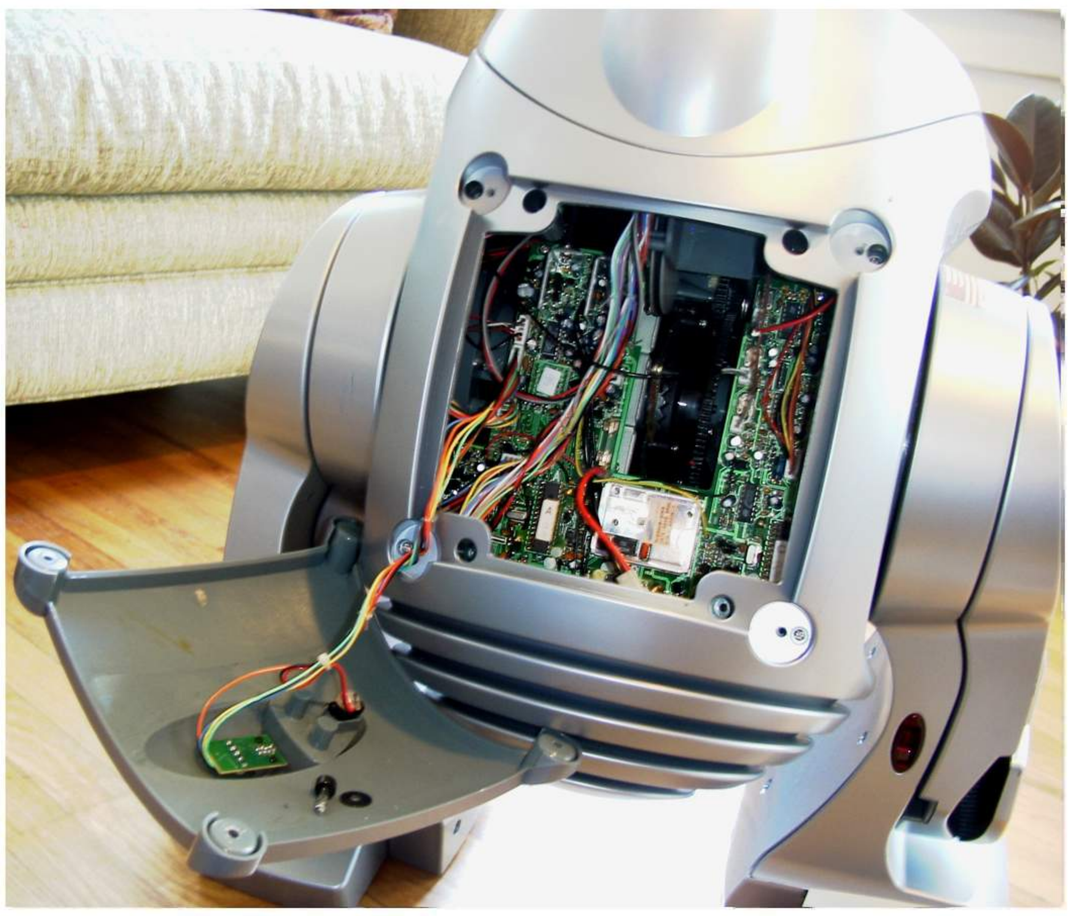
Sections:

Warning: *If you take apart your RoboScout, you may damage it. Also, you'll almost certainly void your warranty. If you decide to take apart your RoboScout anyways, be aware that you do it* ***at your own risk.***

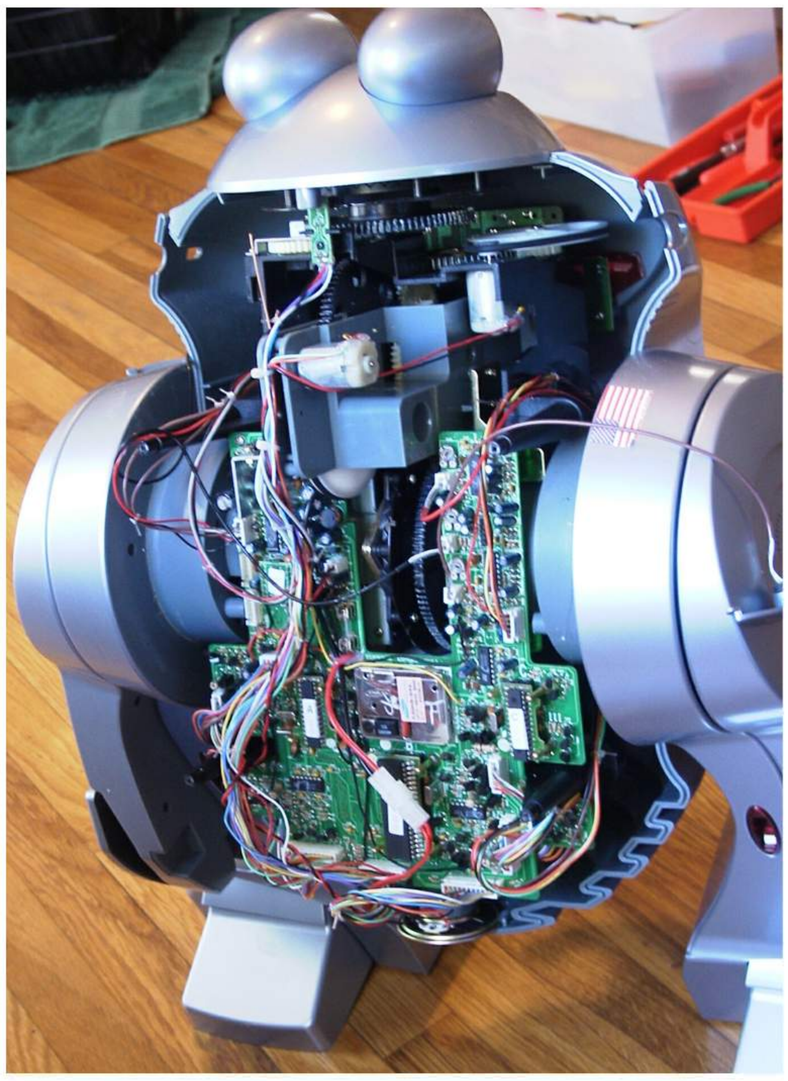
Remove the Back

This part is shown in pages 4 and 5 of the Assembly Diagram. The diagram shows the front PIR housing and chest panel being assembled at the same time as the corresponding rear panels. We found it was easier to completely remove the rear first, and then remove the front.

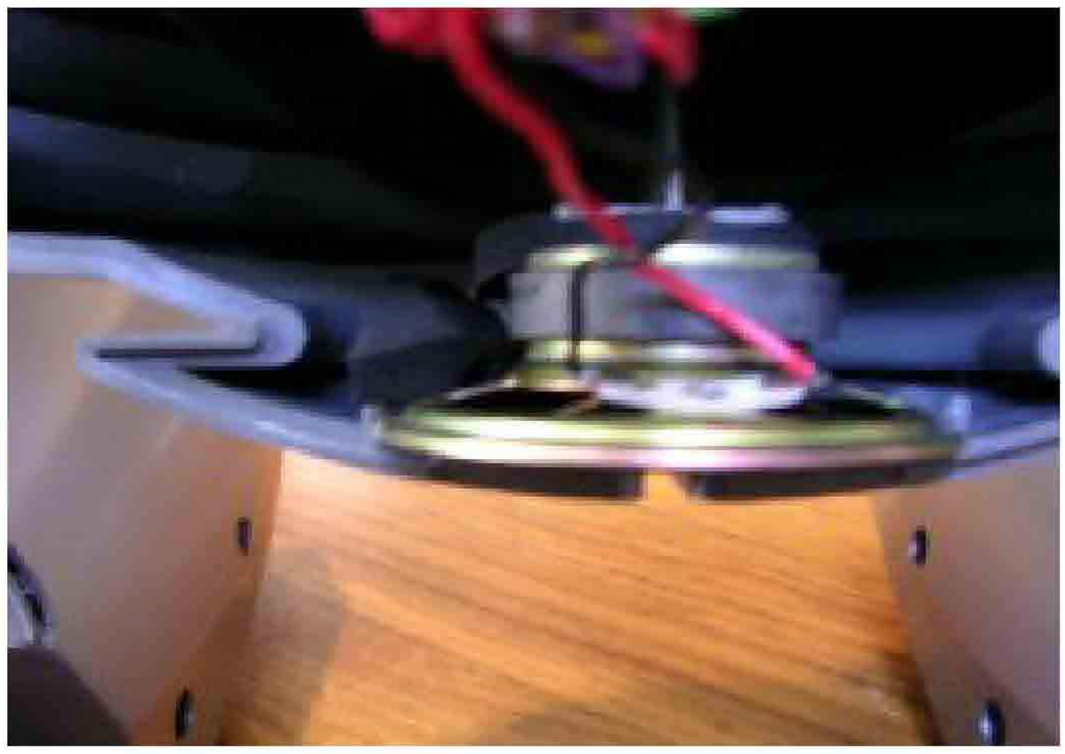
Remove Charge-Display Panel. Unscrew the 4 allen screws (3/16" hex wrench) holding the charge-display panel. Lift the panel free and disconnect the charge-guage's electrical connector at the main board. On our model, the connector was labeled J14 and was located on the left side of the main board.



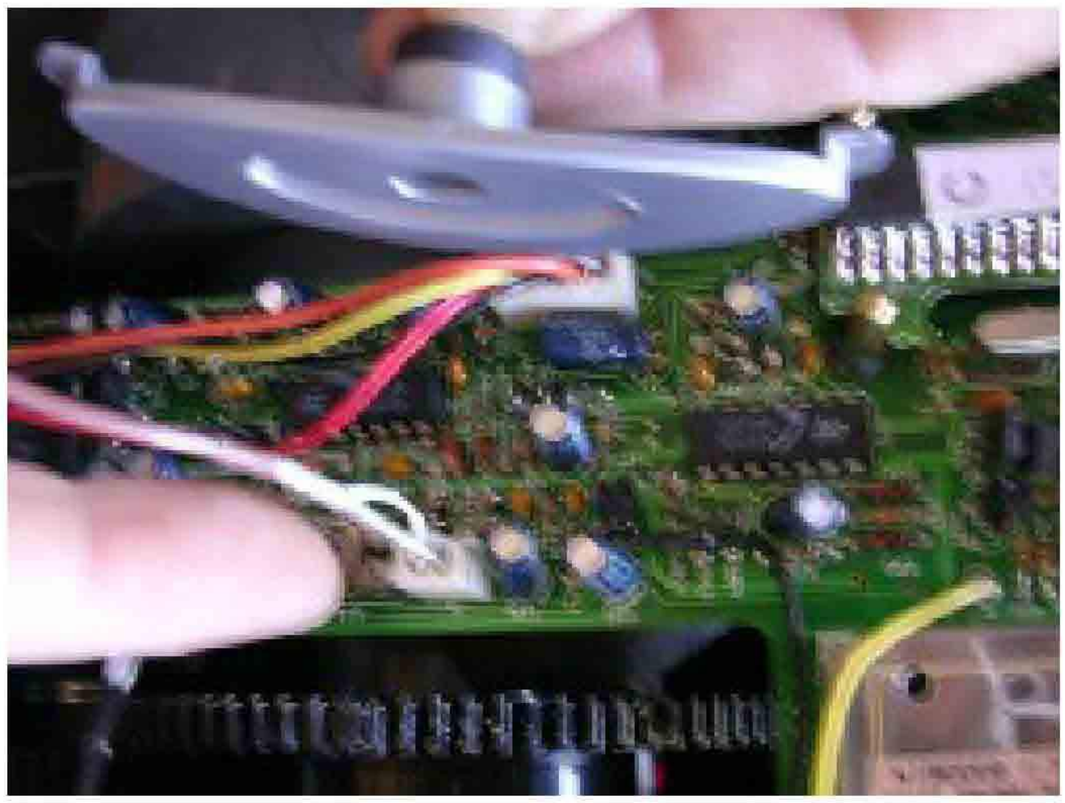
Remove Rear Body Panel. Unscrew the 4 phillips screws securing the rear body panel. Hold the panel up while disconnecting the main power switch from the board. This was connector J1 for us, also on the left side of the board.



Remove Speaker. The voice speaker is at the bottom of the robot. To remove it, disconnect its cables from the main board. For us, that was the connector labeled SPE. Then, push out the foam that's wedged between the speaker and the robot's body to free the speaker.



Remove Side Microphones. Surprisingly, RoboScout has three microphones. Two of these fit between the chest and back panels near the head, looking rather like ears. They fall loose when the back panel is removed. Disconnect them from the main board and put them aside.



Detach the Power Switch

A nice thing to do at this point might be to detach the power switch from the rear body panel. Then you will have a small, convenient switch to use if you want to probe voltages on the main board. Turns out this is possible, though a little difficult.

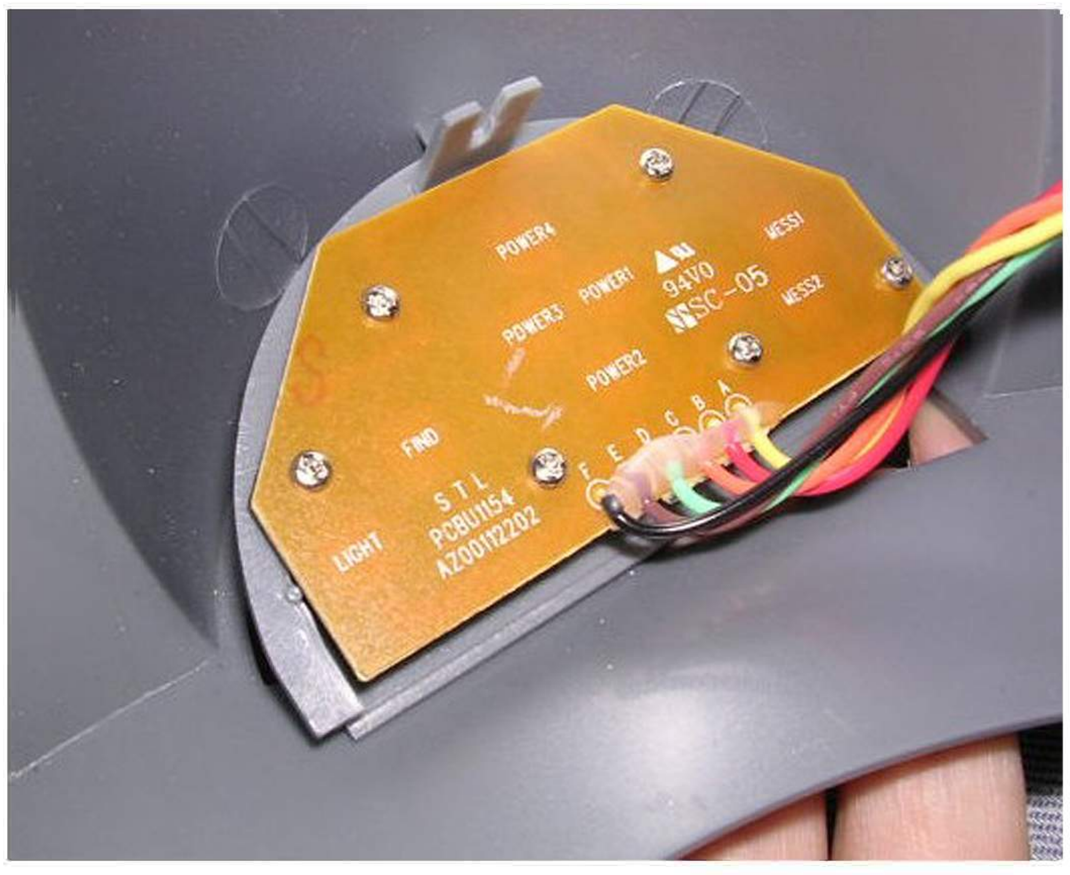
There's also some risk of cracking one or more body panels during this part. If you're concerned about that, you might prefer to rig up your own switch and just skip this section.

This part is shown on page 10 of the Assembly Diagram.

Detach Switch Panel From Rear Body Panel. Sit down and place the rear panel in your lap, concave side toward you. Use a stout screwdriver to pry the right corner of the rear body panel apart from the button panel (CAB707) as shown at right. The goal is to free the right-hand plastic tab shown on page 10 of the Assembly Diagram.

Once the right-hand tab is free, the button panel can be separated by twisting it to free the bottom hook and left-hand tab.

Detach Power-Switch PCB. Remove the phillips screw securing the PCB to CAB707. Press out the PCB unit by pushing up the edge where the retaining screw was. You will then be able to slide out the small PCB with switches and reattach its cabling to the main board at connector J1.

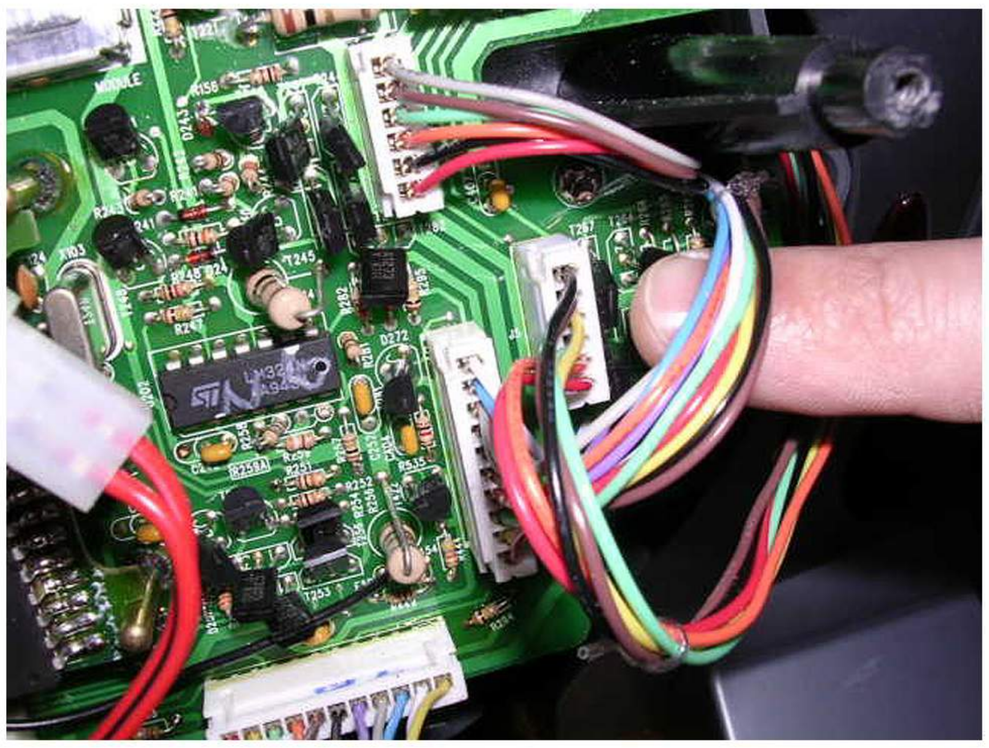


Remove Front Panel and PIR Unit

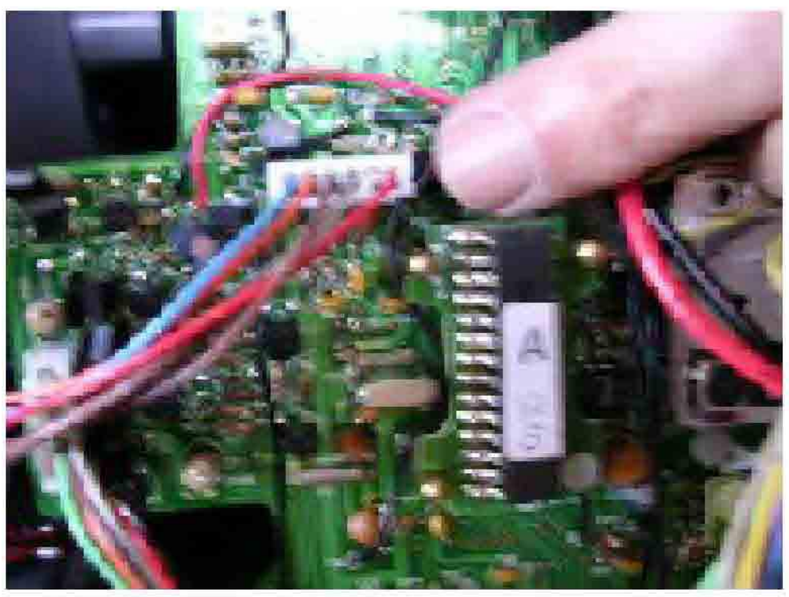
The front panel contains quite a few sensors -- a PIR motion detector for locating warm objects (such as people), a headlight, an infrared proximity detector, and a third microphone. It also contains gearing and a small DC motor for rotating the PIR sensor unit. To remove it, several electrical connectors must be unplugged from the main board and threaded towards the front. If the rear body panel has been removed, as in the previous part, these connectors will be within easy reach.

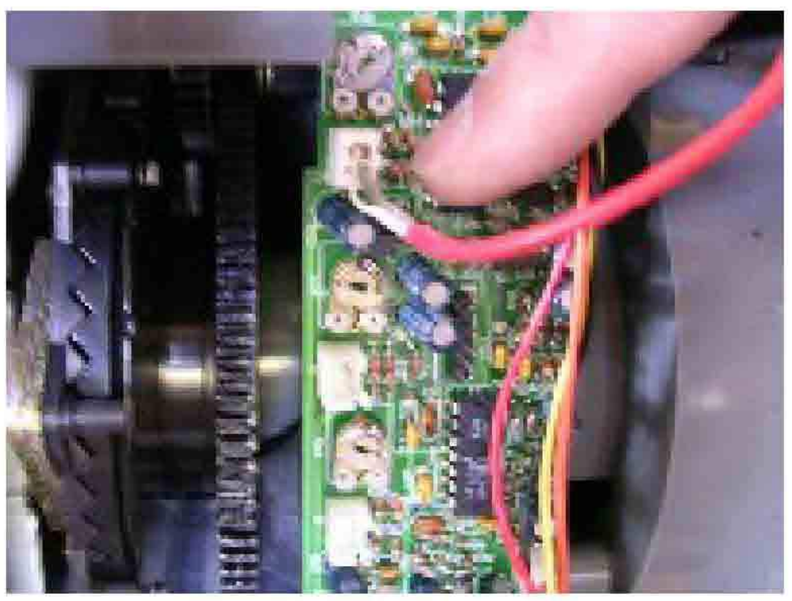
This part is shown in pages 4 and 11 of the Assembly Diagram.

Remove Front Panel. Disconnect the following from the main board: J5 (bottom right - motor and encoder for the PIR Unit), J16 (left side, near chip A - headlamp and infrared sensor), J7 (top right - front microphone), and J9 (midway up on right - PIR sensor). Unscrew the 4 allen screws (3/16" hex wrench) securing the front panel and pull it free, threading the electrical connectors and cables through the body until they are completely removed.

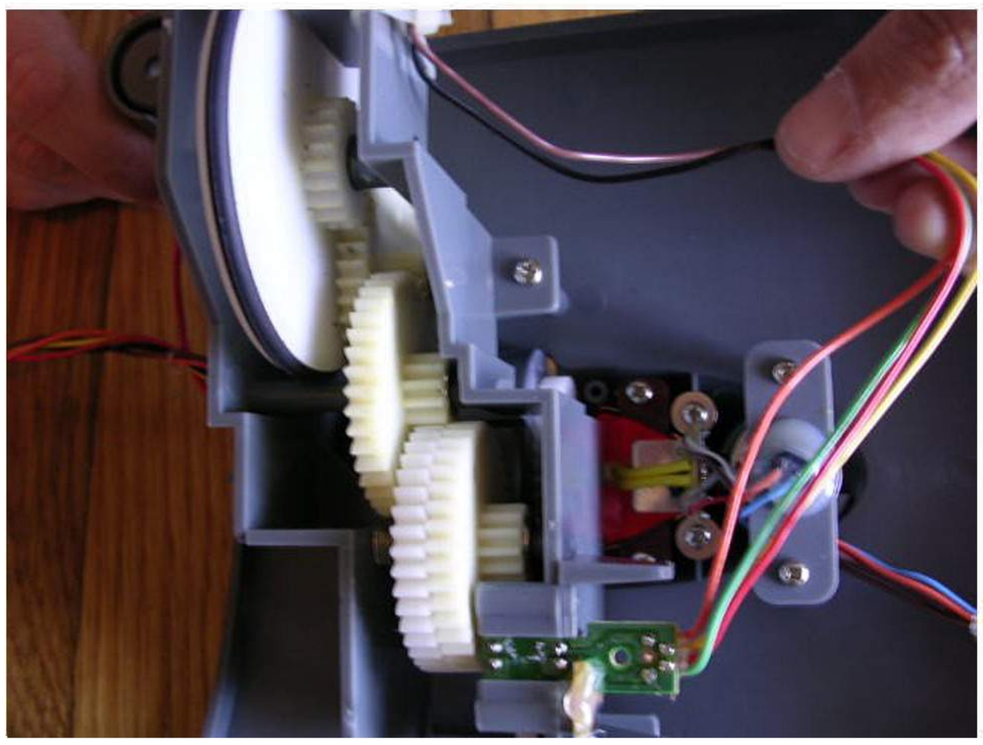


The figure at right shows the location of J5. The locations of J16, J7, and J9 are shown below:

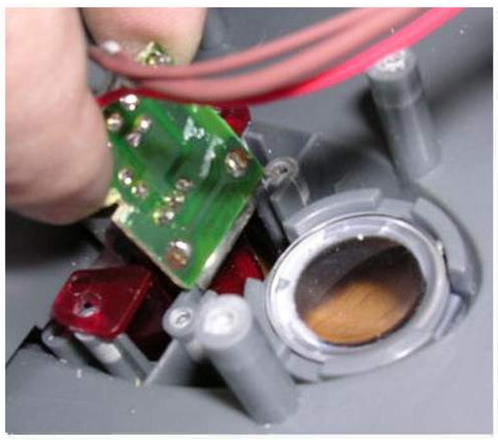




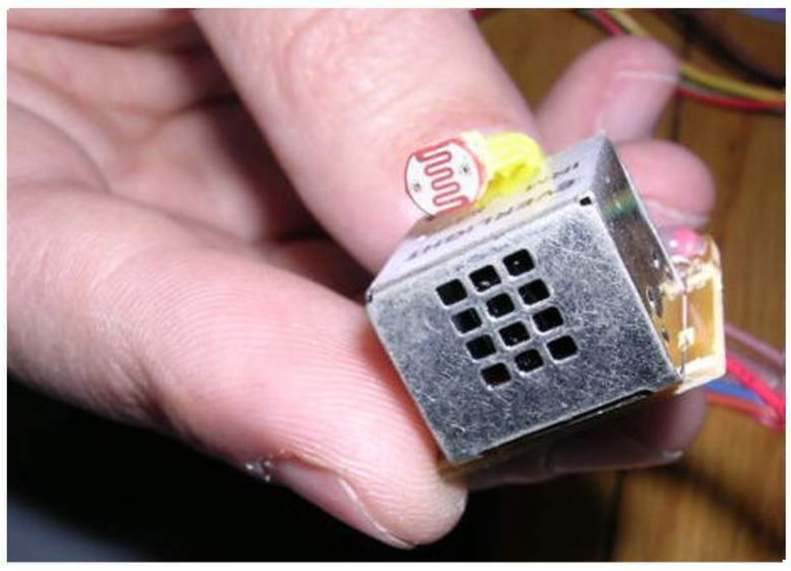
Inside View of Front Panel. Here's the inside of the front panel right after it's been removed.



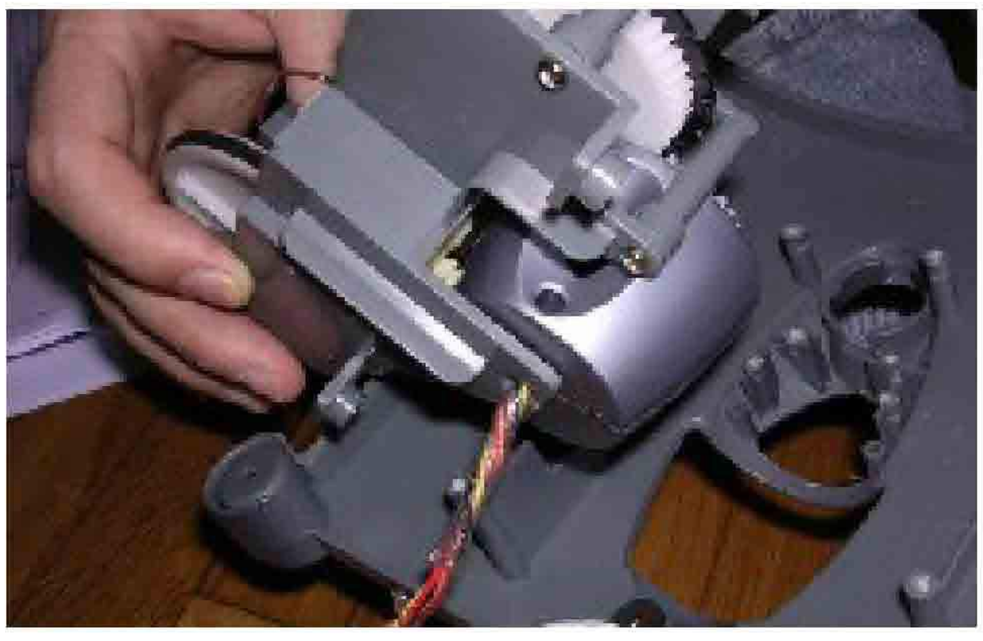
Remove Headlamp and Infrared Proximity Sensor. If you want, you can also dismantle the front panel itself. The headlamp bulb, reflector, and lens lift right out once the gray plastic retaining piece is removed. To remove the infrared proximity detector, unscrew the phillips screws holding it in, then use a screwdriver to carefully pry out the small PCB board with the proximity detector from its slot.



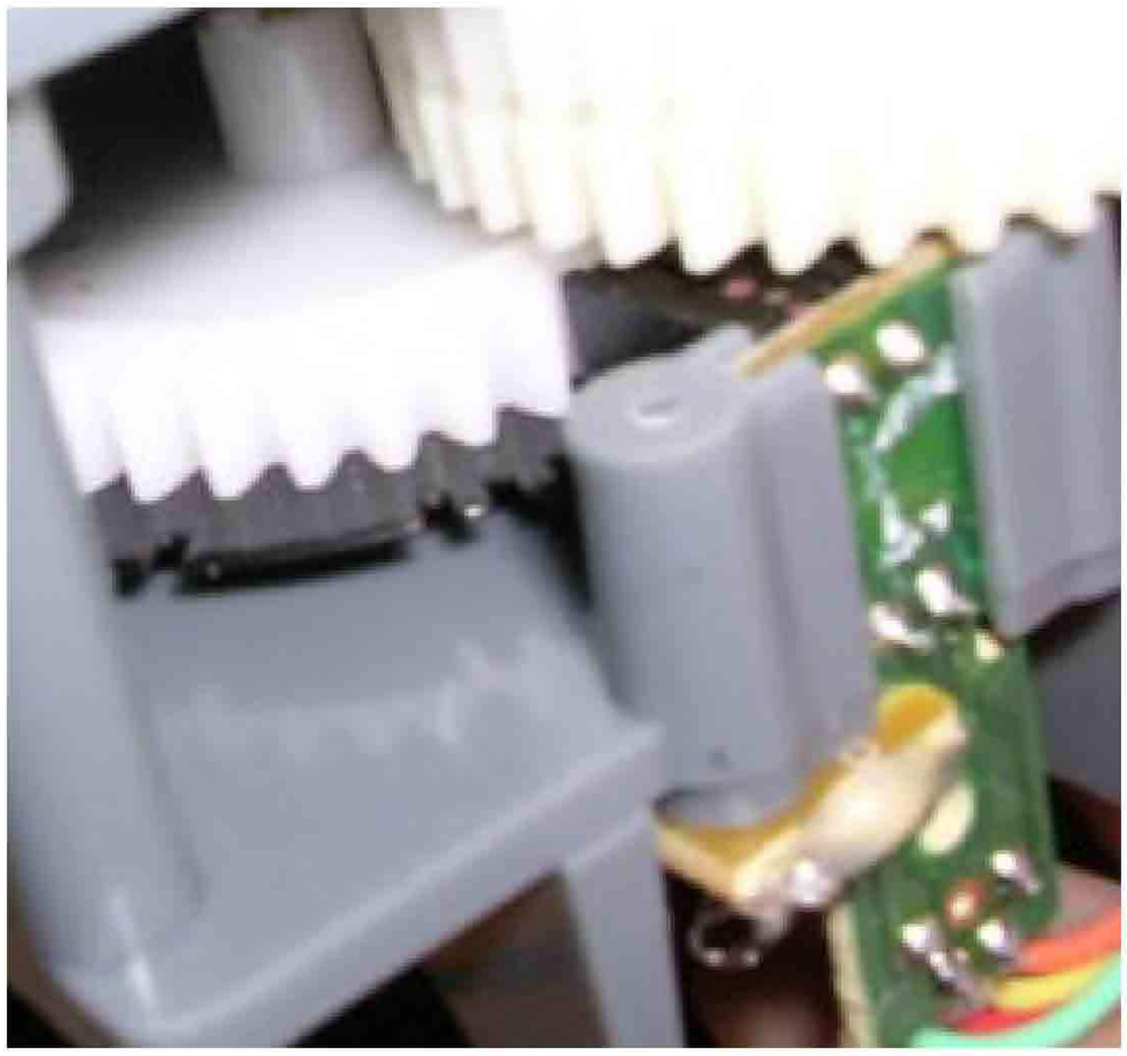
Close-up of the infrared proximity detector:



Remove PIR Unit. Unscrew the 4 phillips screws securing the PIR unit to the front panel. These screws are glued in to hold them despite vibrations, so they're very tight.



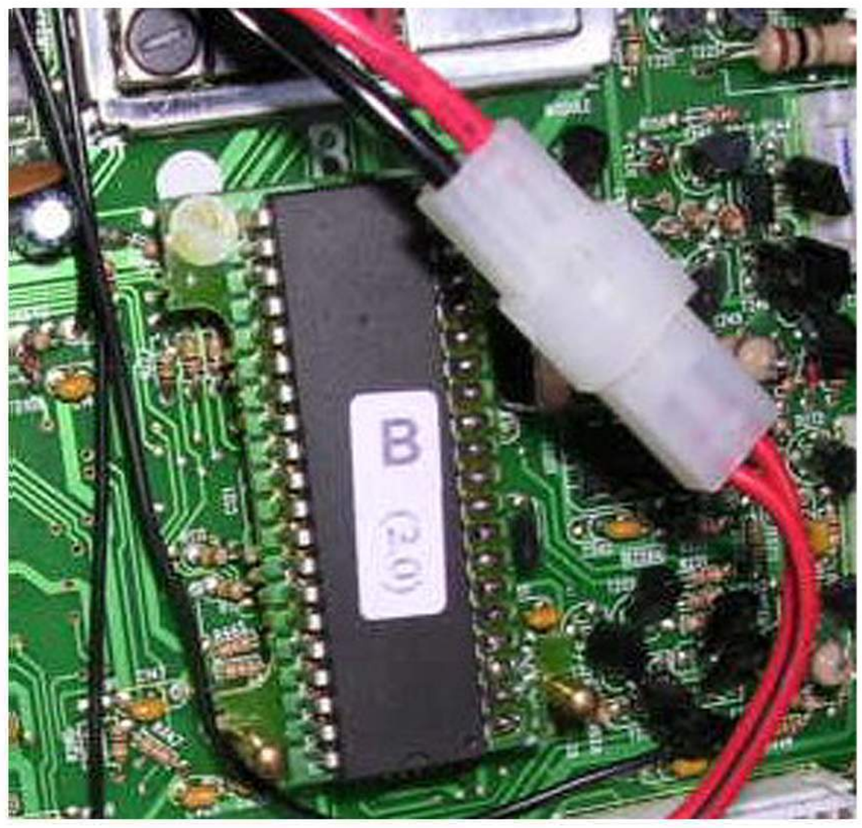
Close-Up of PIR Unit's Encoder.
There are two sets of spacings on the encoder wheel. The gaps to the right in this picture are close together, while those to the left are fairly far apart. The region of the encoder wheel with tight gap spacing should be centered under the interrupter beam when the PIR unit is facing straight forward.



Remove Main Board

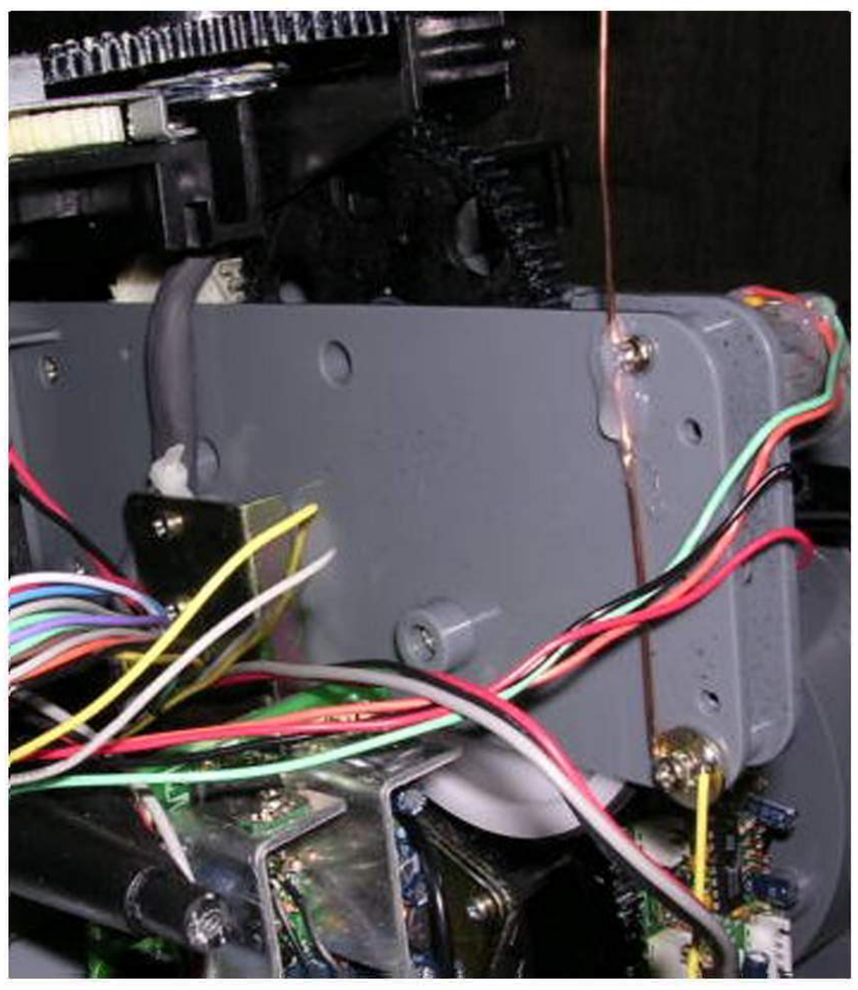
This part is shown on page 6 of the Assembly Diagram.

Disconnect Main Power. Remove the fuse from the main board. It's located just above and to the left of the wireless receiver unit.



Carefully disconnect the main battery power by separating the connector shown at right. It's hard to separate, and you don't want to stick a screwdriver down inside it because you'll short across the battery terminals. A couple pairs of pliers and some patience did the trick for us.

Disconnect All Cables. Use a razor blade to slice the glue holding the receiver antenna to the body's gear box (figure at right) and cut it free. Also, remove the phillips screw securing it to the gear box.

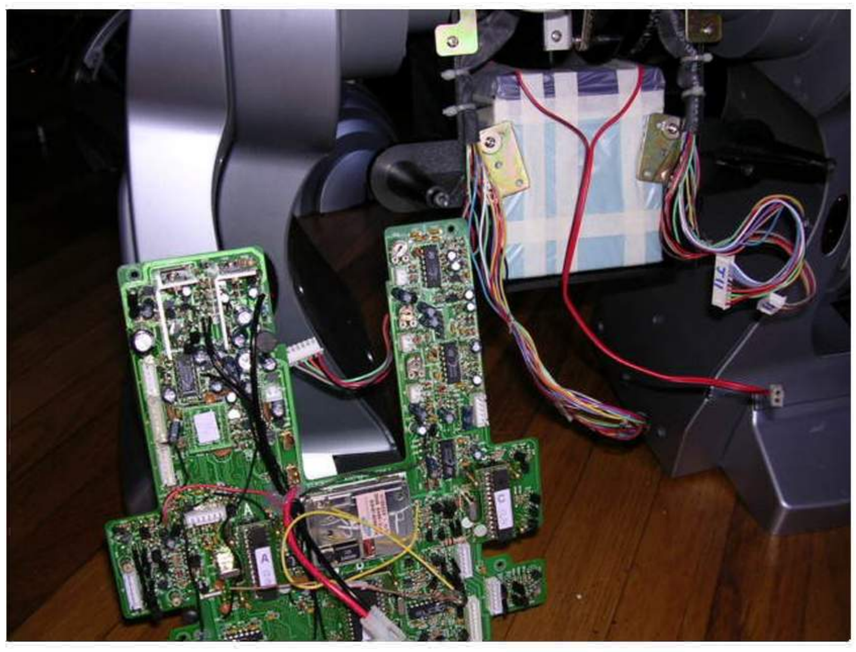


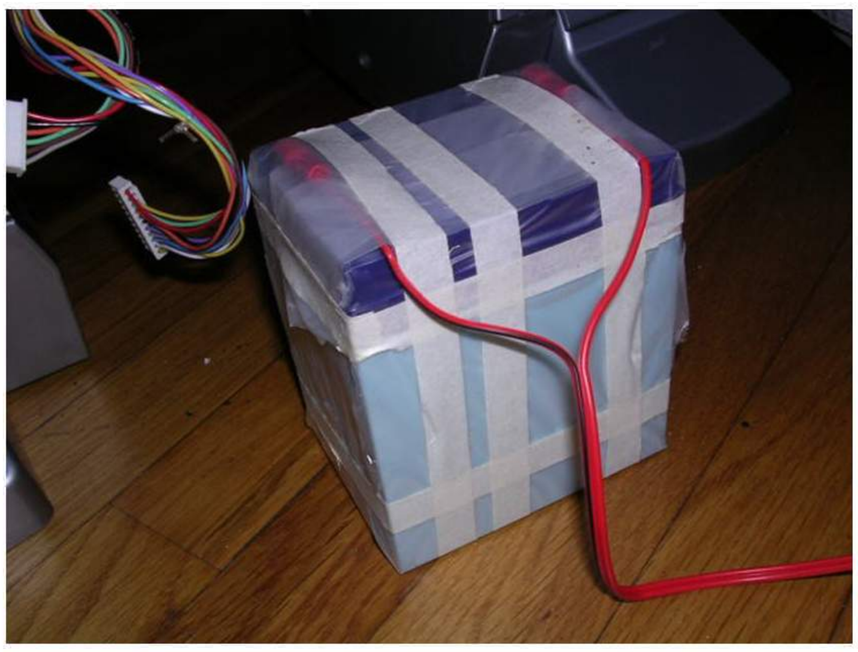
Unplug any remaining connectors that are still plugged into the main board. These will include J2 and J10 for the left arm and leg, J3 and J11 for the right arm and leg, J13 for the mouth LEDs, J15 for the video transmitter, J4 for the head, and any others you see that are still plugged in.

Take Out the Main Board. Unscrew the 4 phillips screws securing the main board to the robot and lift it out. If you want to reuse or test this board, be careful to avoid electrostatic discharge while handling it. Wrap it in anti-static plastic and set it aside.

Remove The Battery

Once the main board is removed, the battery is easily accessible. Unscrew the phillips screws securing the 2 retaining brackets and slide the battery out.

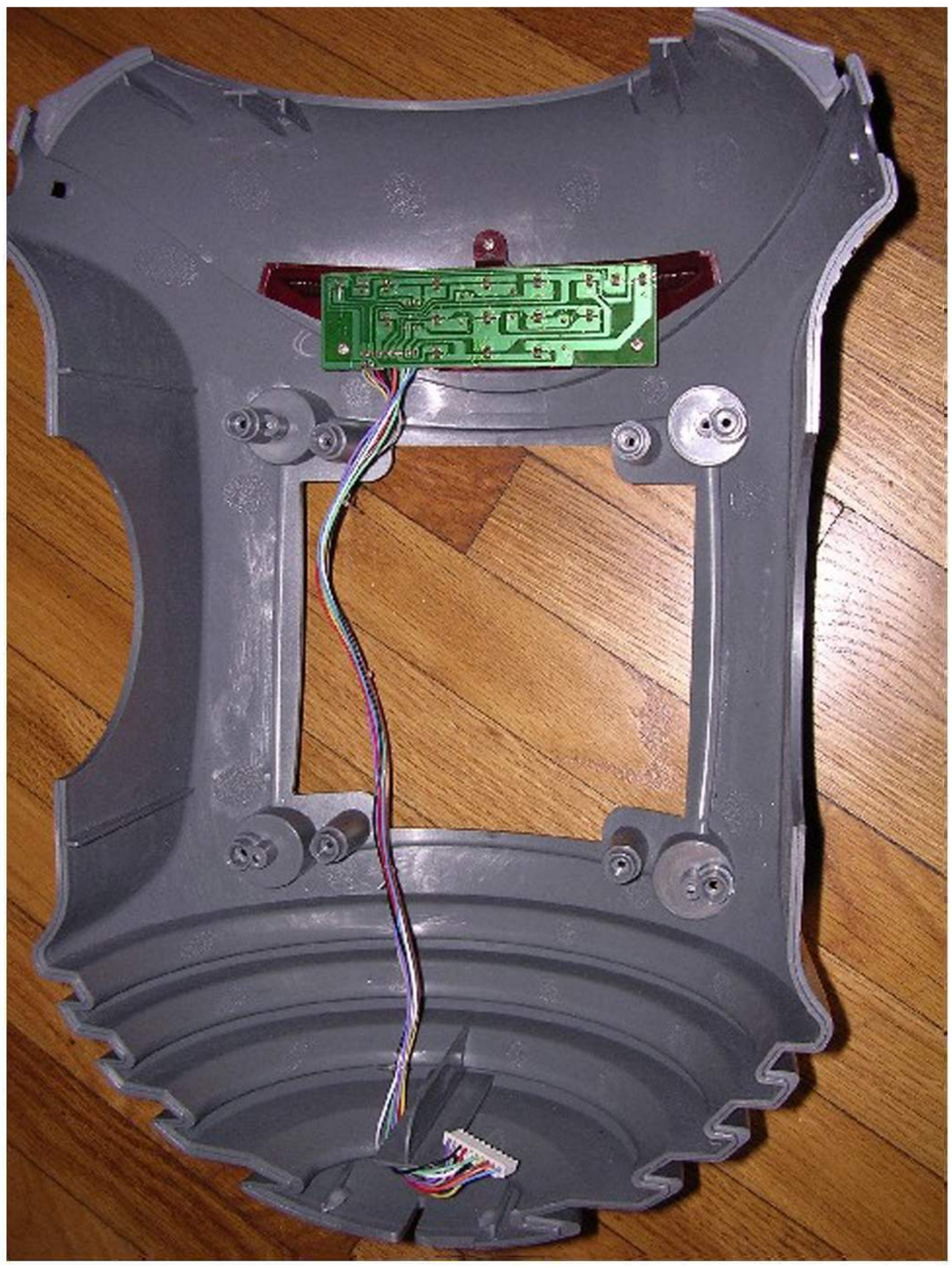




Remove Chest Panel

This part is shown in page 5 of the Assembly Diagram.

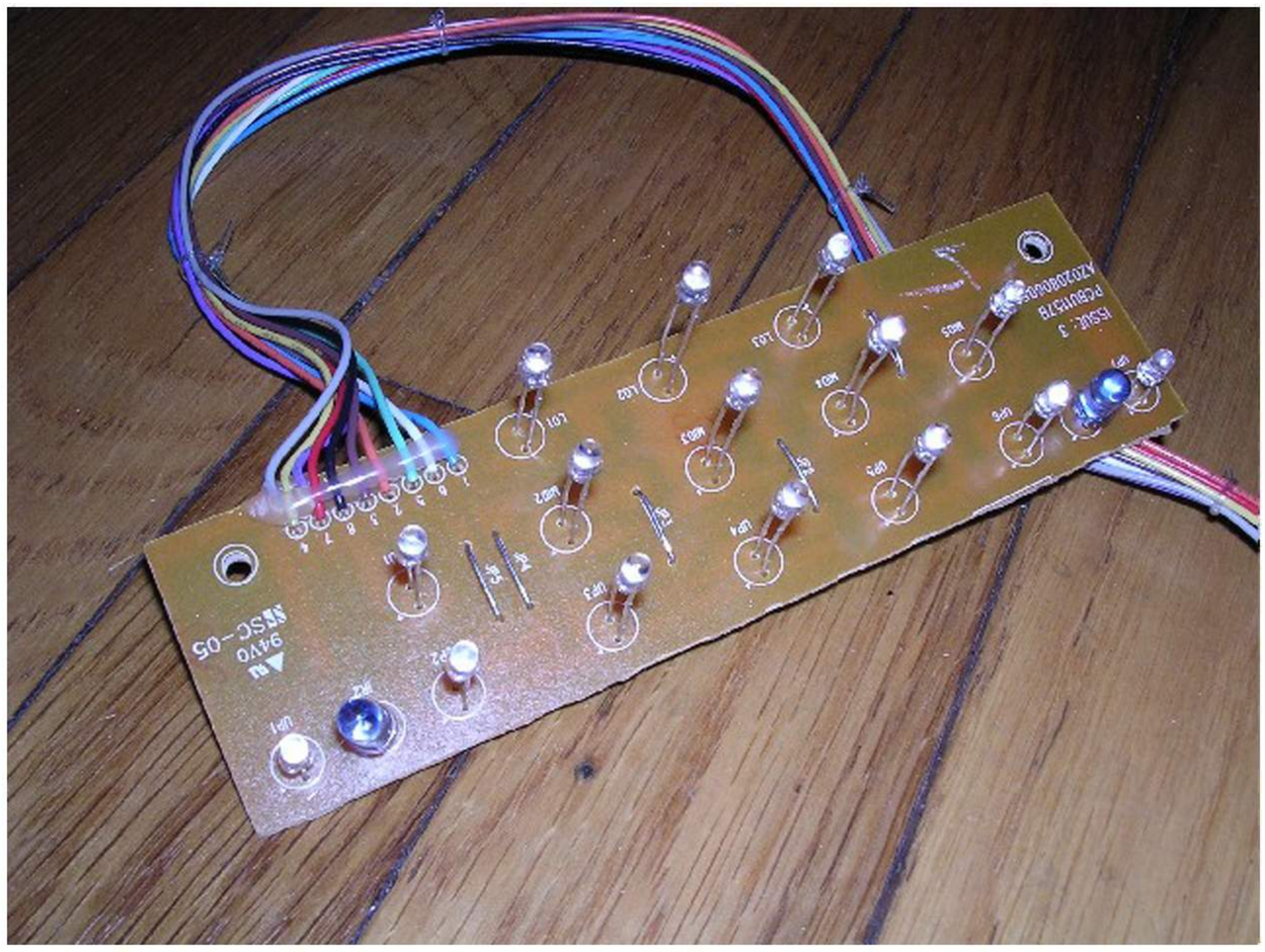
Remove Chest Panel. To remove the chest panel, unscrew the 4 phillips screws securing it. Remove the chest panel while threading connector J13 and wiring forward through the body.



Remove LED Panel For Mouth. The LED panel and wiring for the mouth can be removed from the chest panel by unscrewing the 2 phillips screws securing it.

Interestingly, two of the mouth LEDs are infrared.

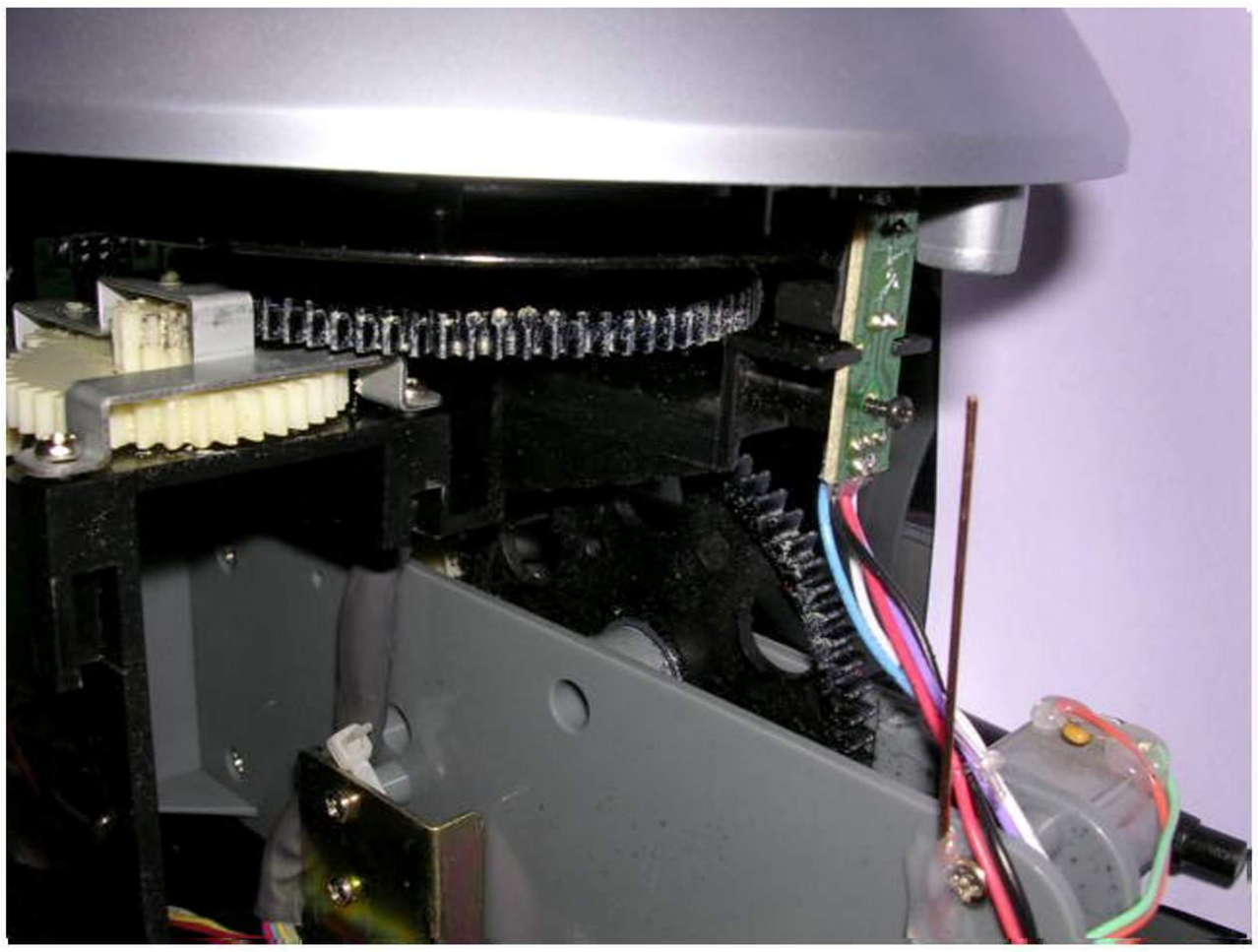
Complete Disassembly



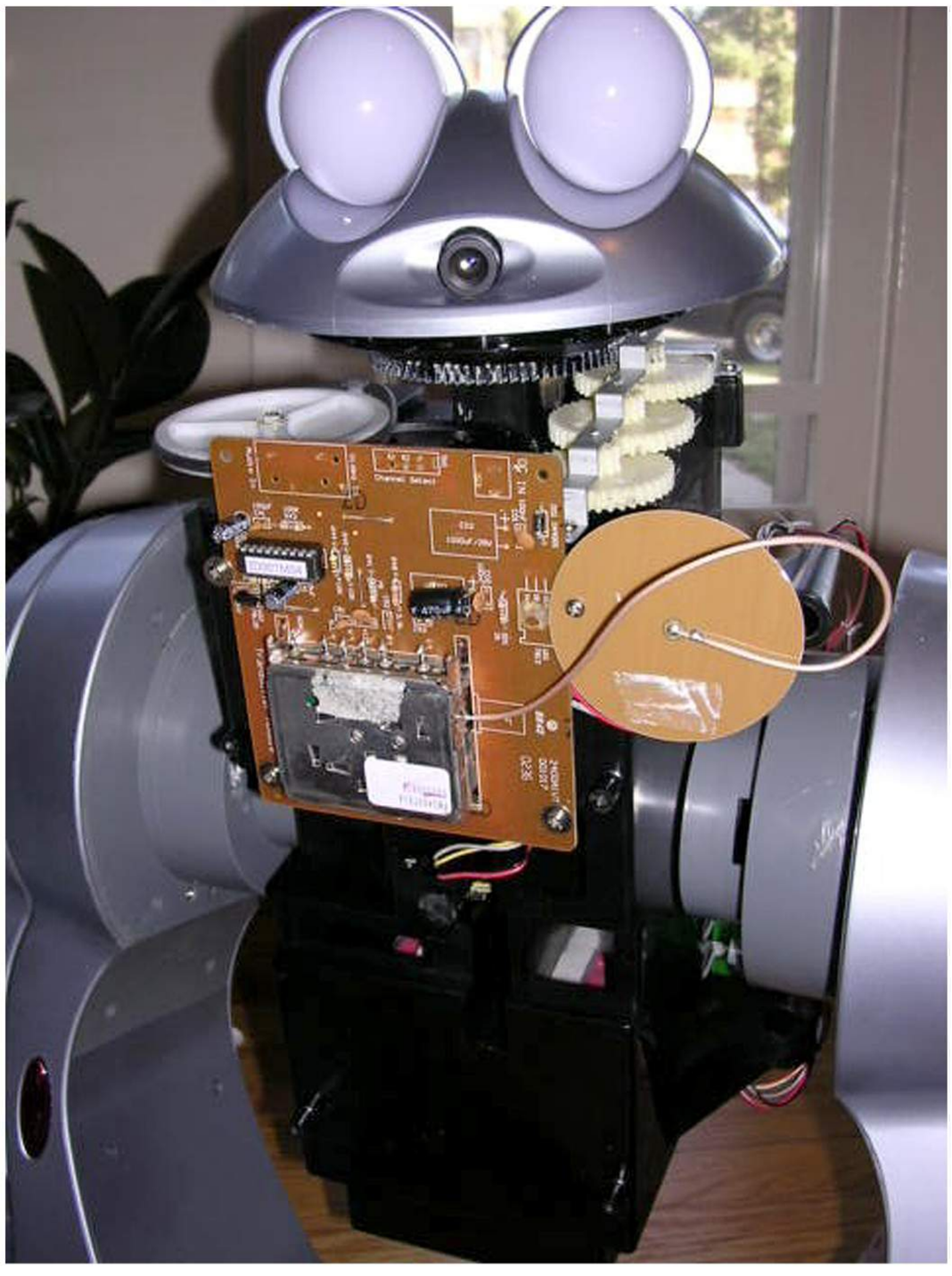
With chest and back panels removed, many of RoboScout's internal mechanisms are visible.

At bottom left, a second wireless unit and antenna can be seen in the front of the robot. These connect to the camera and transmit the video signal to the remote. Gearing for turning the head is also visible in this photo.

The photo at bottom right shows the encoder disk and interrupter for head motion, located in back. A portion of the gear train for pivoting the body is also visible.

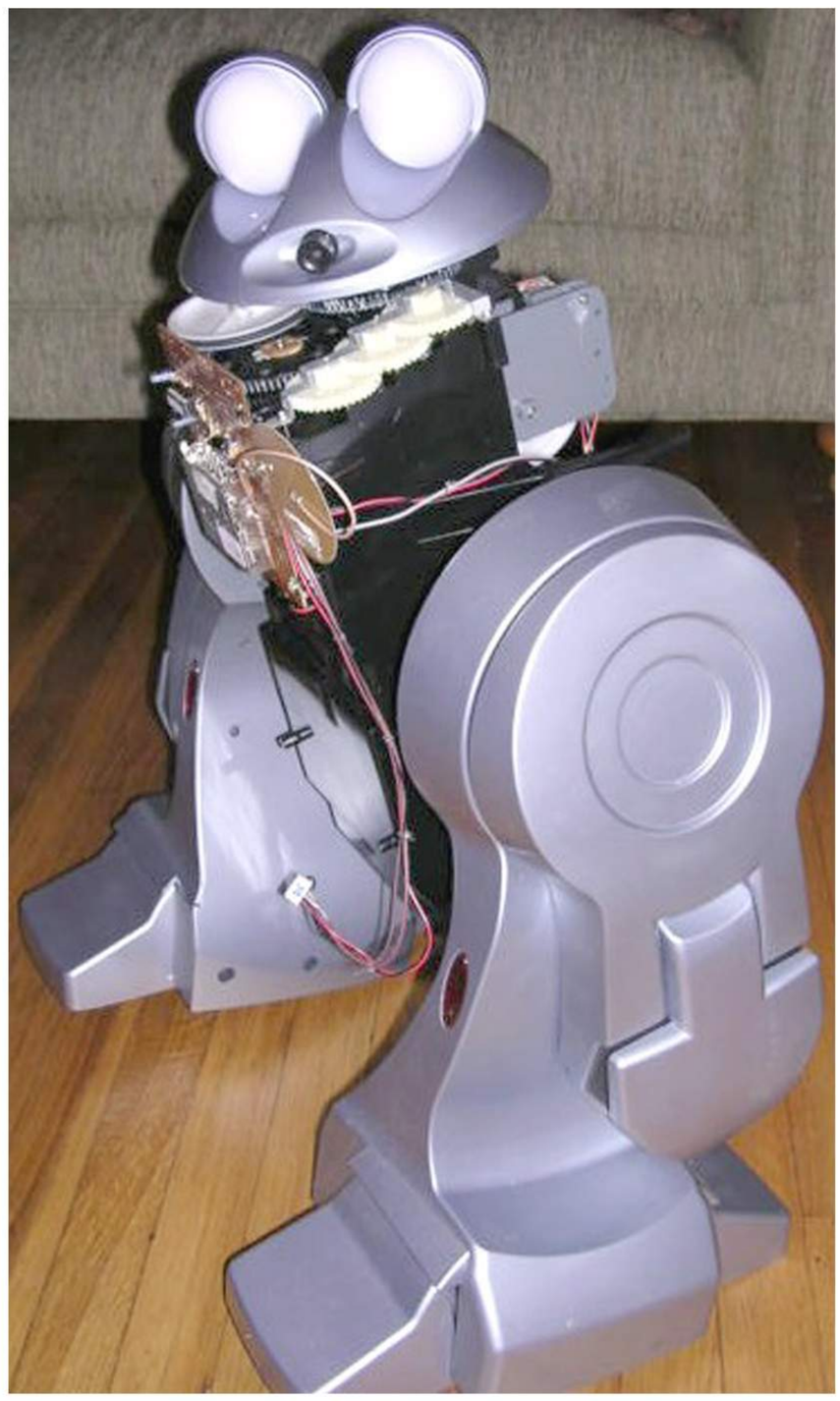


Complete Disassembly



Remove Head and Gearbox Assembly

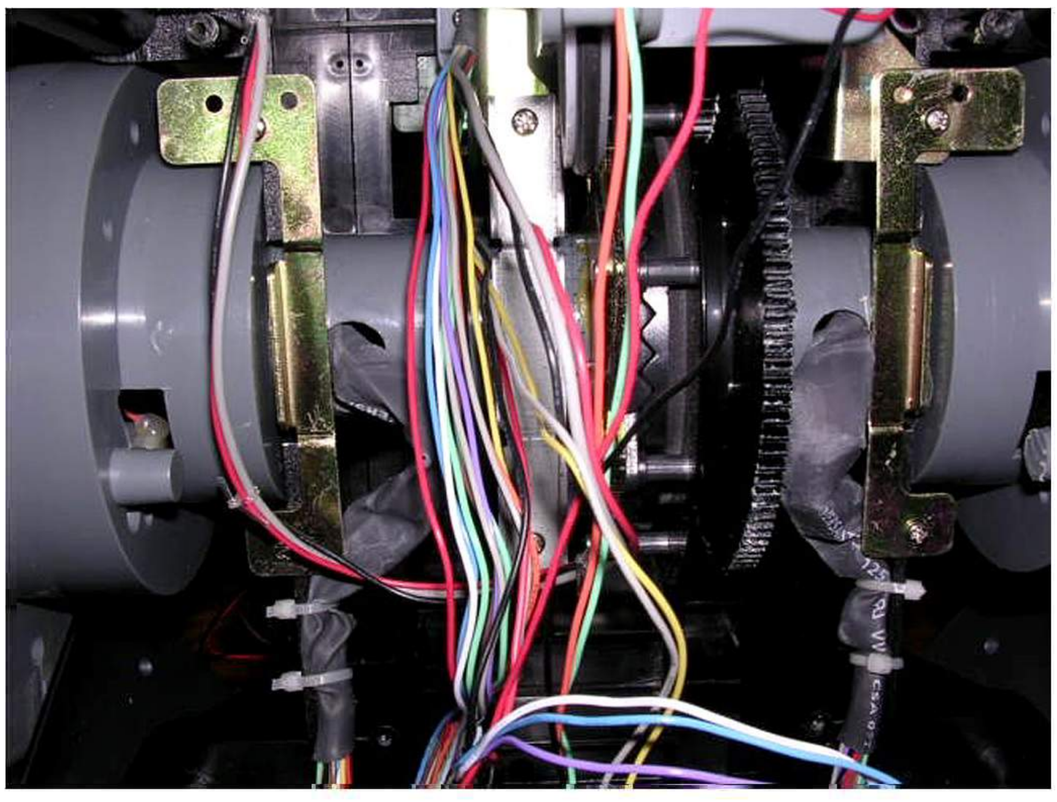
At this point, RoboScout has both chest and back removed, and looks like the picture at right.



The next step is to separate the head and gearbox assembly from the leg frame assembly.

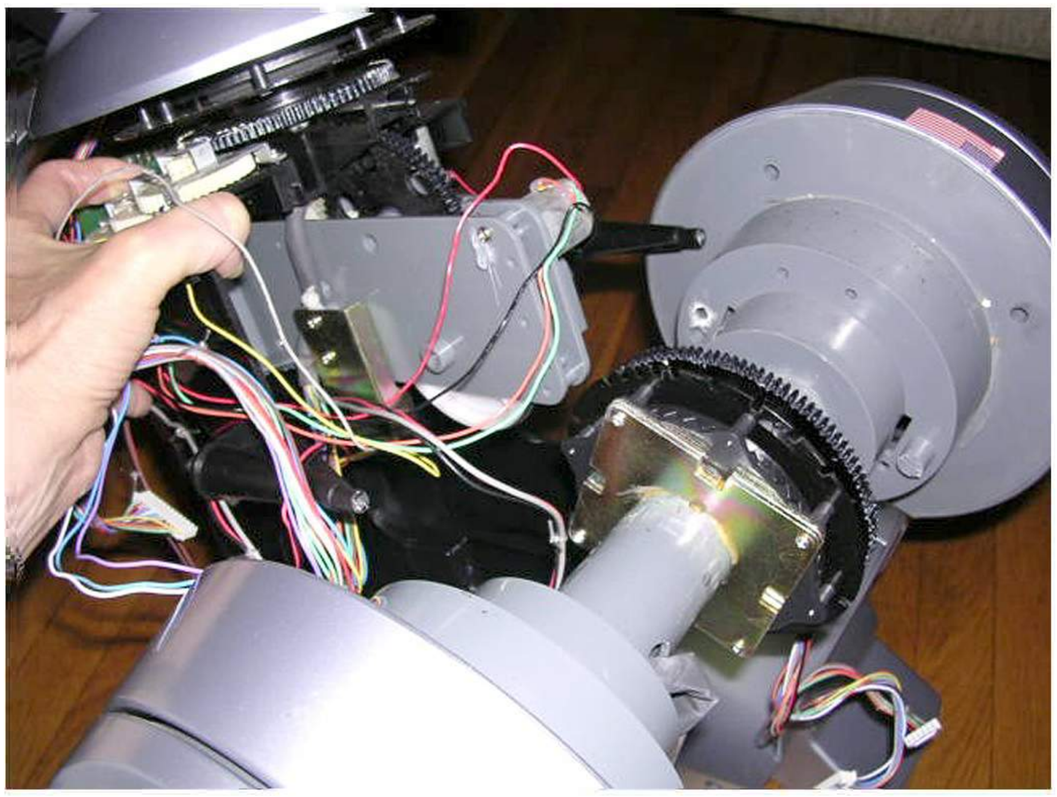
This part is shown on page 6 of the Assembly Diagram.

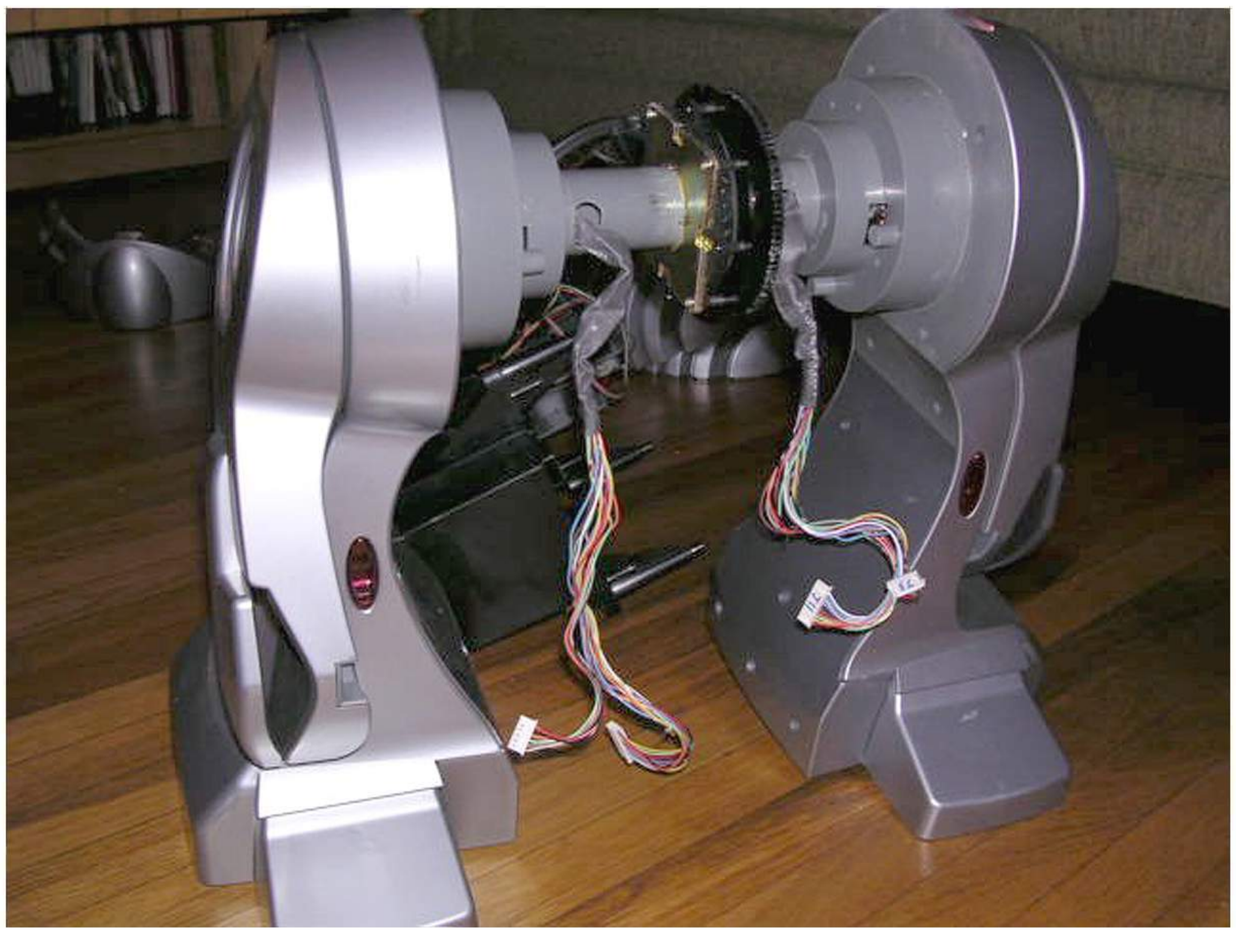
Clip Cable Ties. Clip the four nylon cable ties that secure the wiring harness to the battery box. These are visible in the lower third of the photo at right (click photo for larger view).

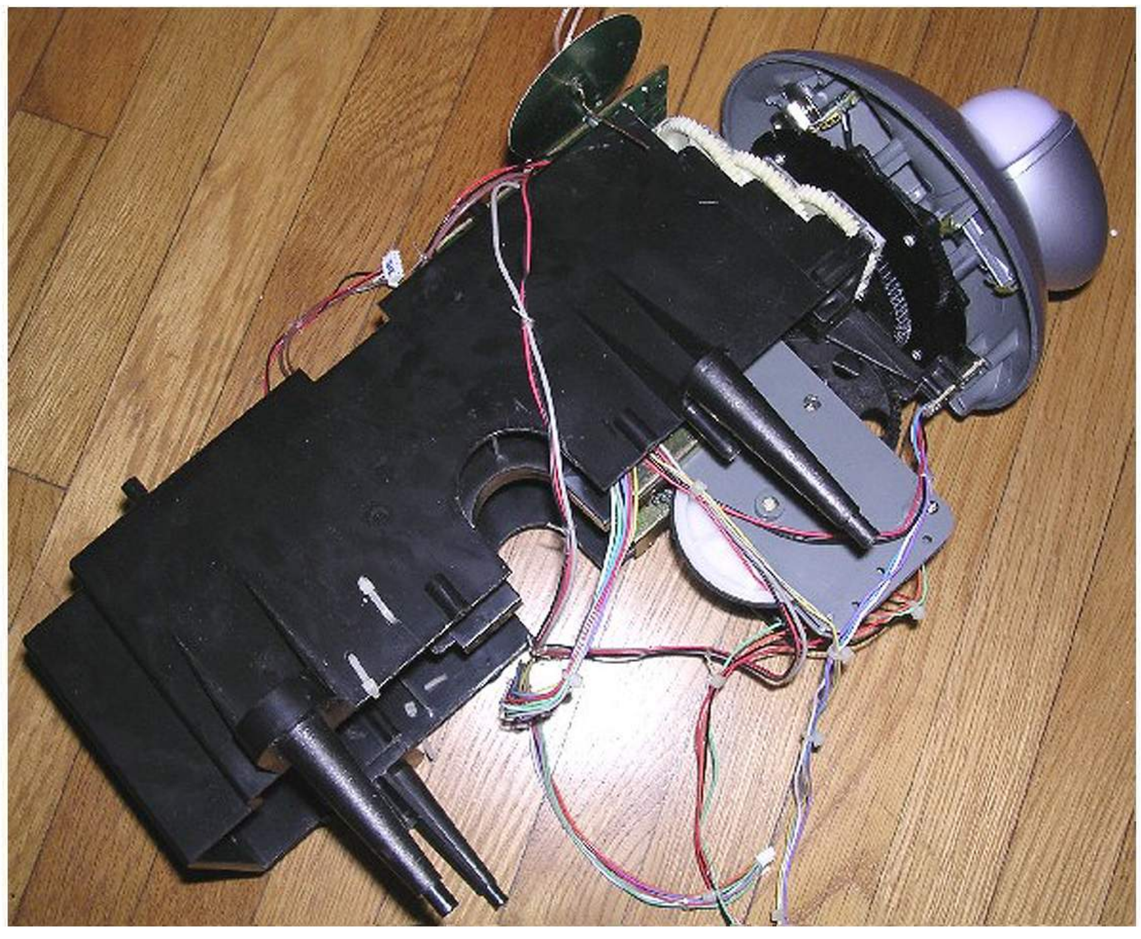


Remove Brackets. Remove the three brackets that secure the head and gearbox assembly to the leg frame. These can also be seen in the photo at right.

Remove Head and Gearbox Assembly.
Once the brackets are removed, the head and gearbox assembly can be pulled away from the leg frame as shown in the picture at right. The Assembly Diagram has an especially clear drawing on page 6 showing how these two pieces fit together.





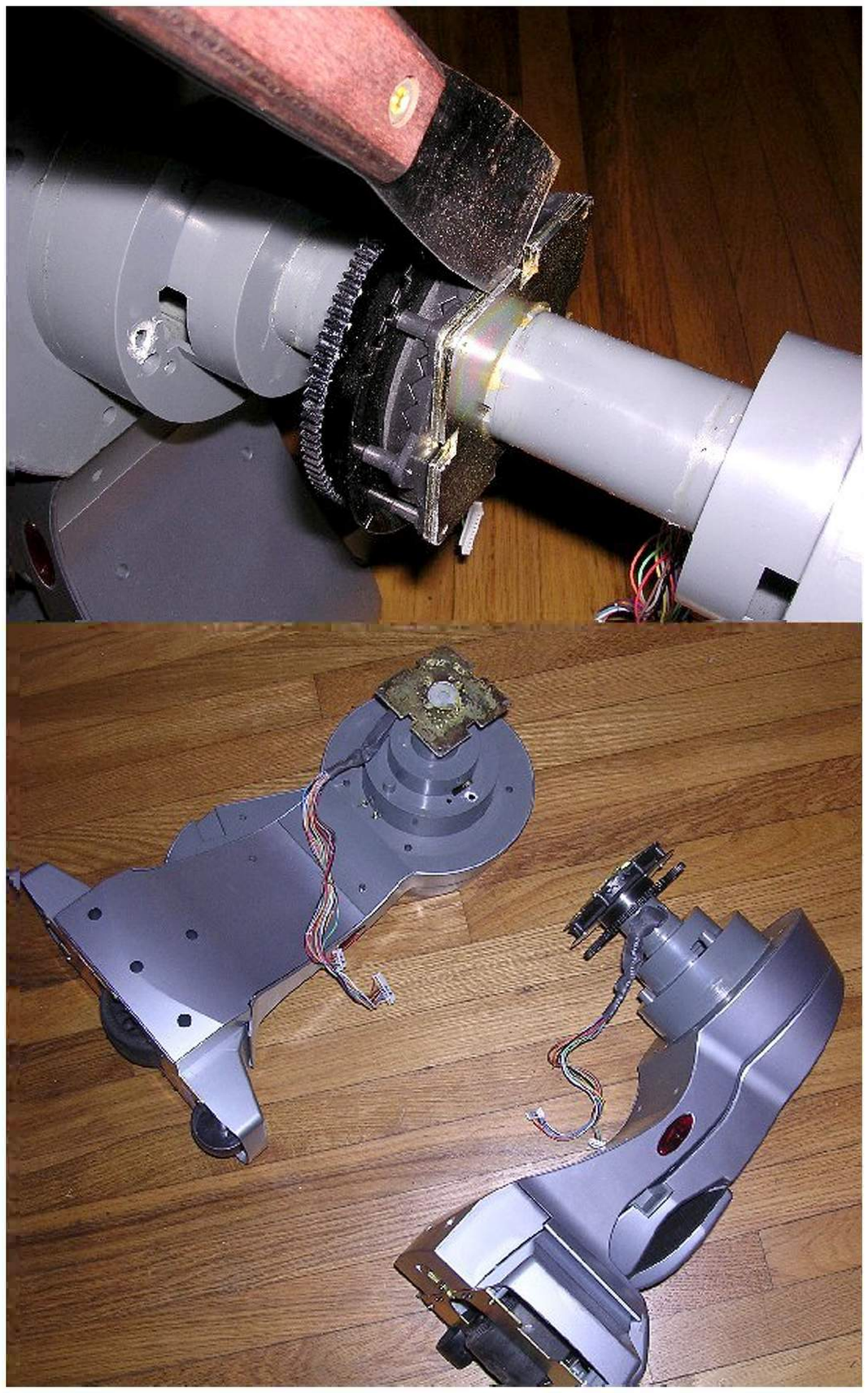


Separate Leg Frame

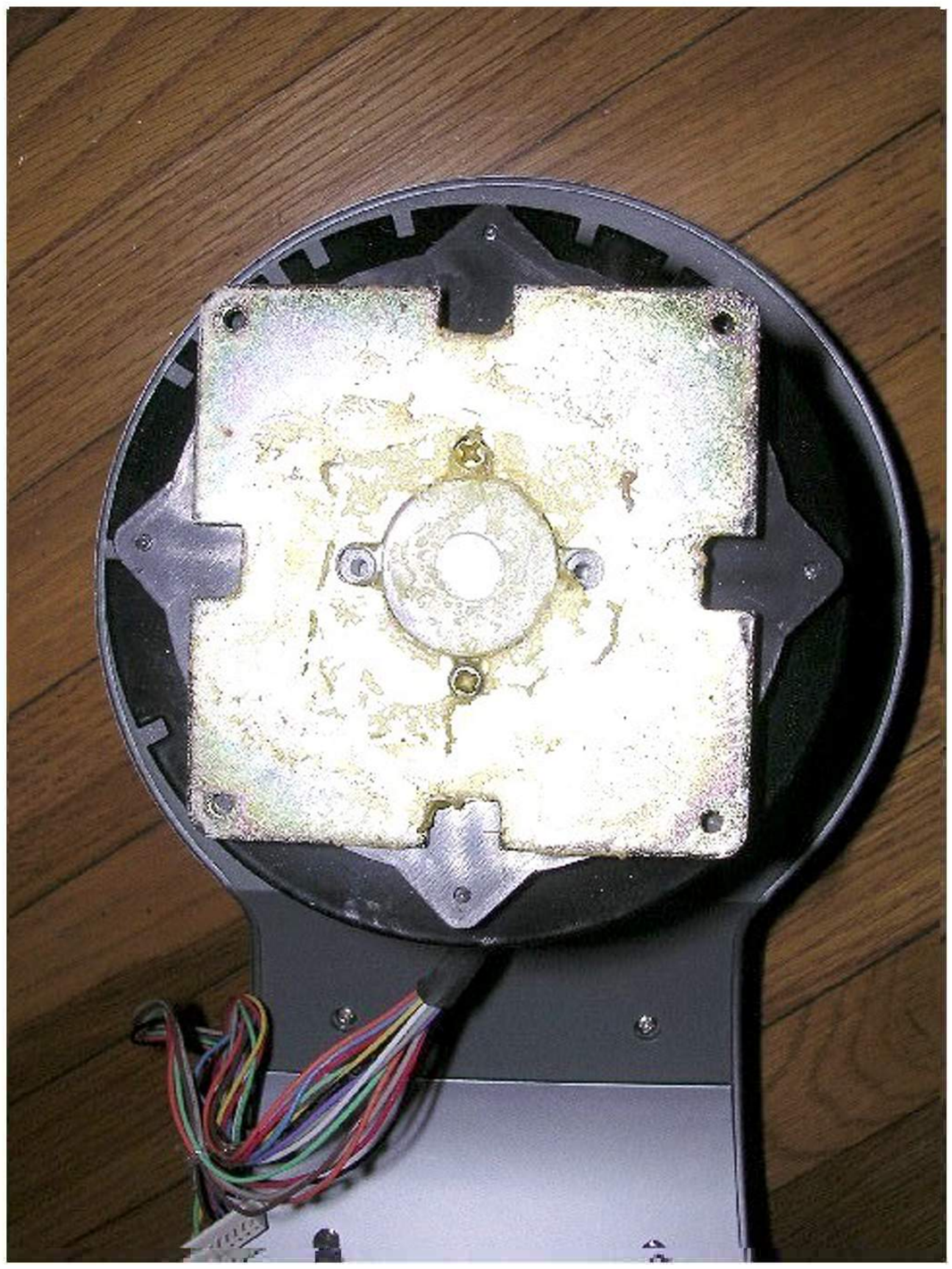
Remove the 4 phillips screws at the corners of the anodized brackets near the center of the leg frame.

Page 7 of the Assembly Diagram shows these screws as the only fasteners holding together the two halves of the leg frame. But in our RoboScout, the brackets were also tightly glued together.

To separate them, we used a stiff putty knife and tapped it in between the brackets with a small hammer. We discovered the brackets are very easily bent, so if you plan to reassemble, work slowly from the outer edges toward the center, rotating the assembly frequently and tapping in evenly from all directions.



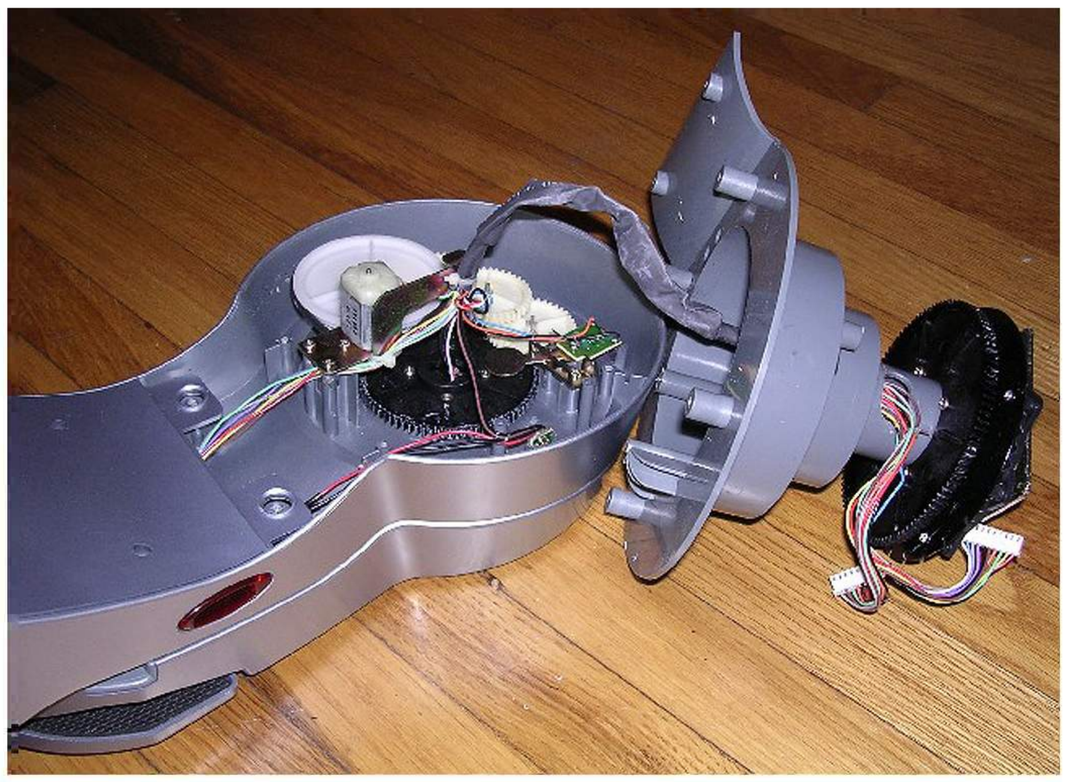
The encoder disk for the body's bending movement is attached to the right arm-and-leg assembly. The gap arrangement is very non-symmetric (click photo for larger image).



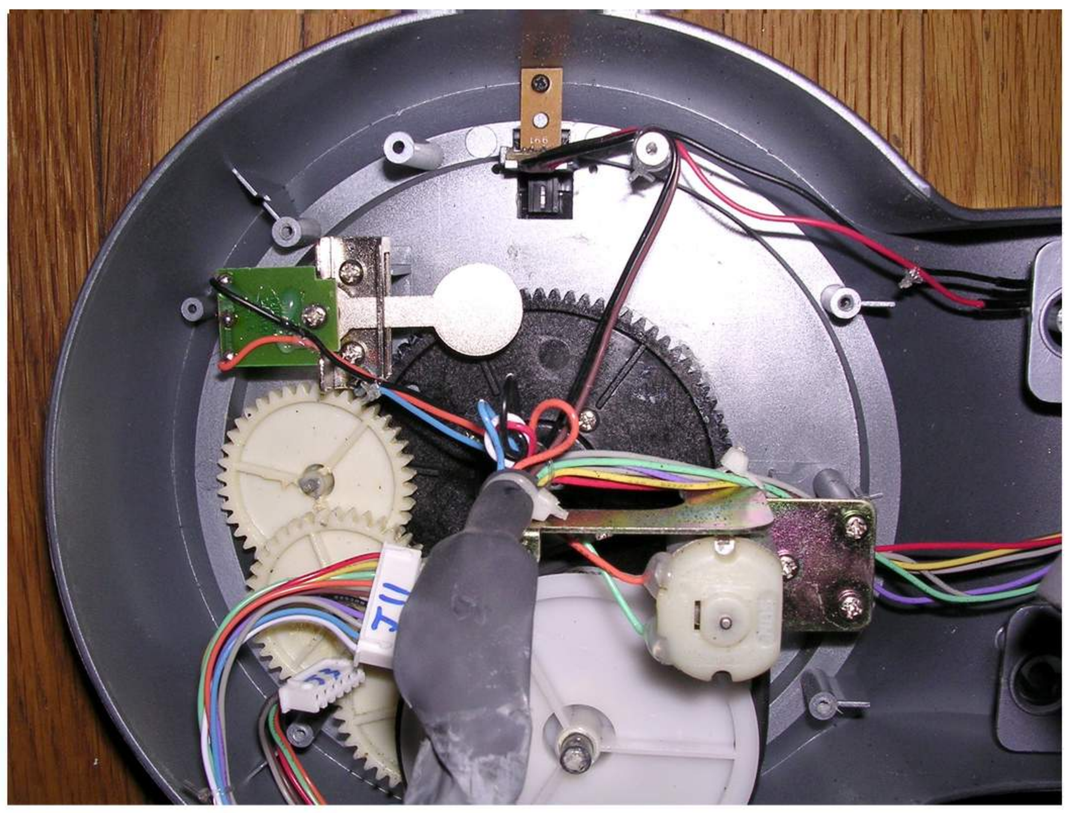
Remove Right Arm

This part is shown in page 8 of the Assembly Diagram.

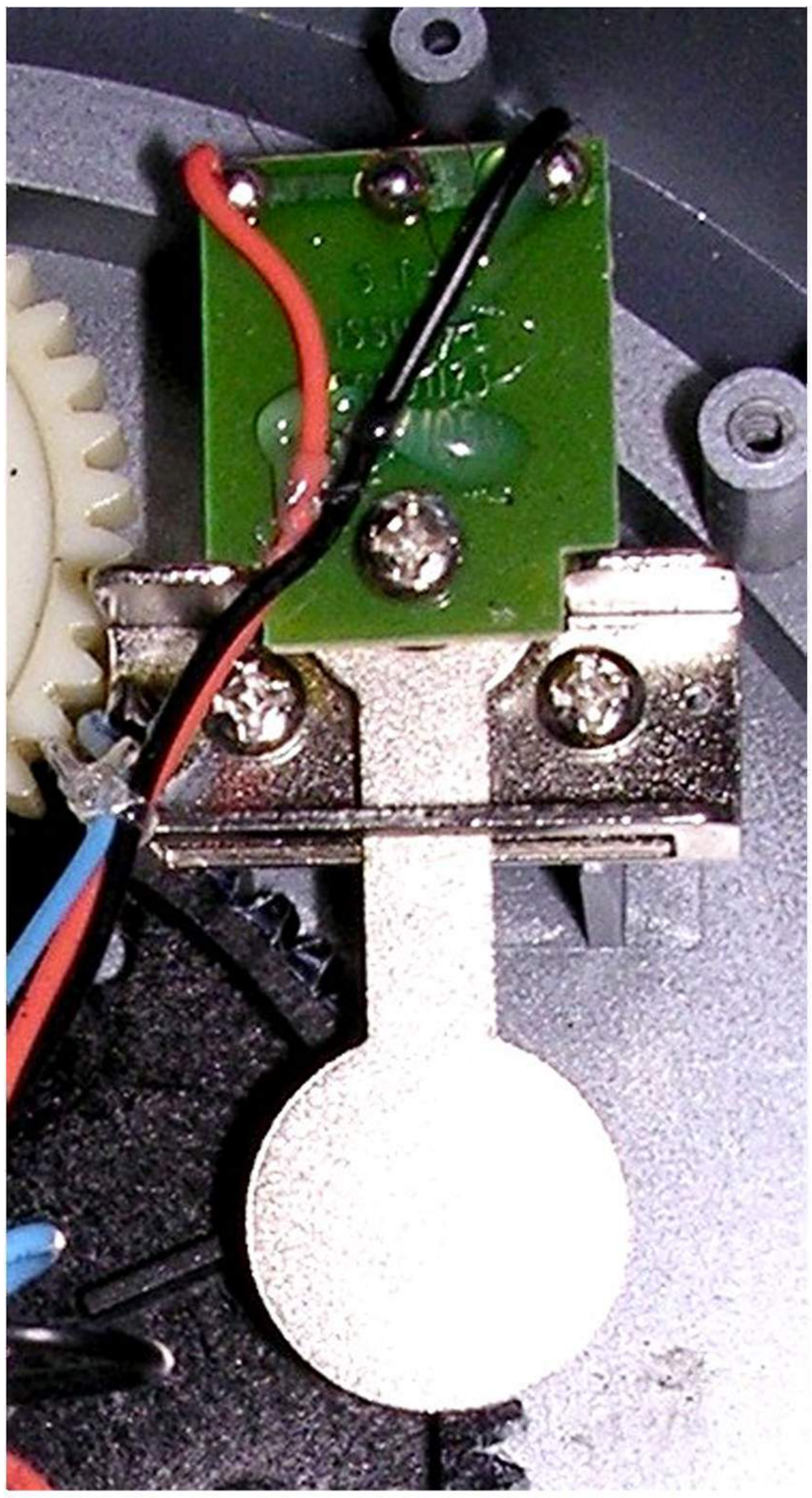
Remove Top Access Panel. Unscrew the 6 phillips screws that attach the topmost panel of the arm-and-leg assembly. Lift off the panel while threading cables through hole. Push the cables completely through the hole and set the panel aside.

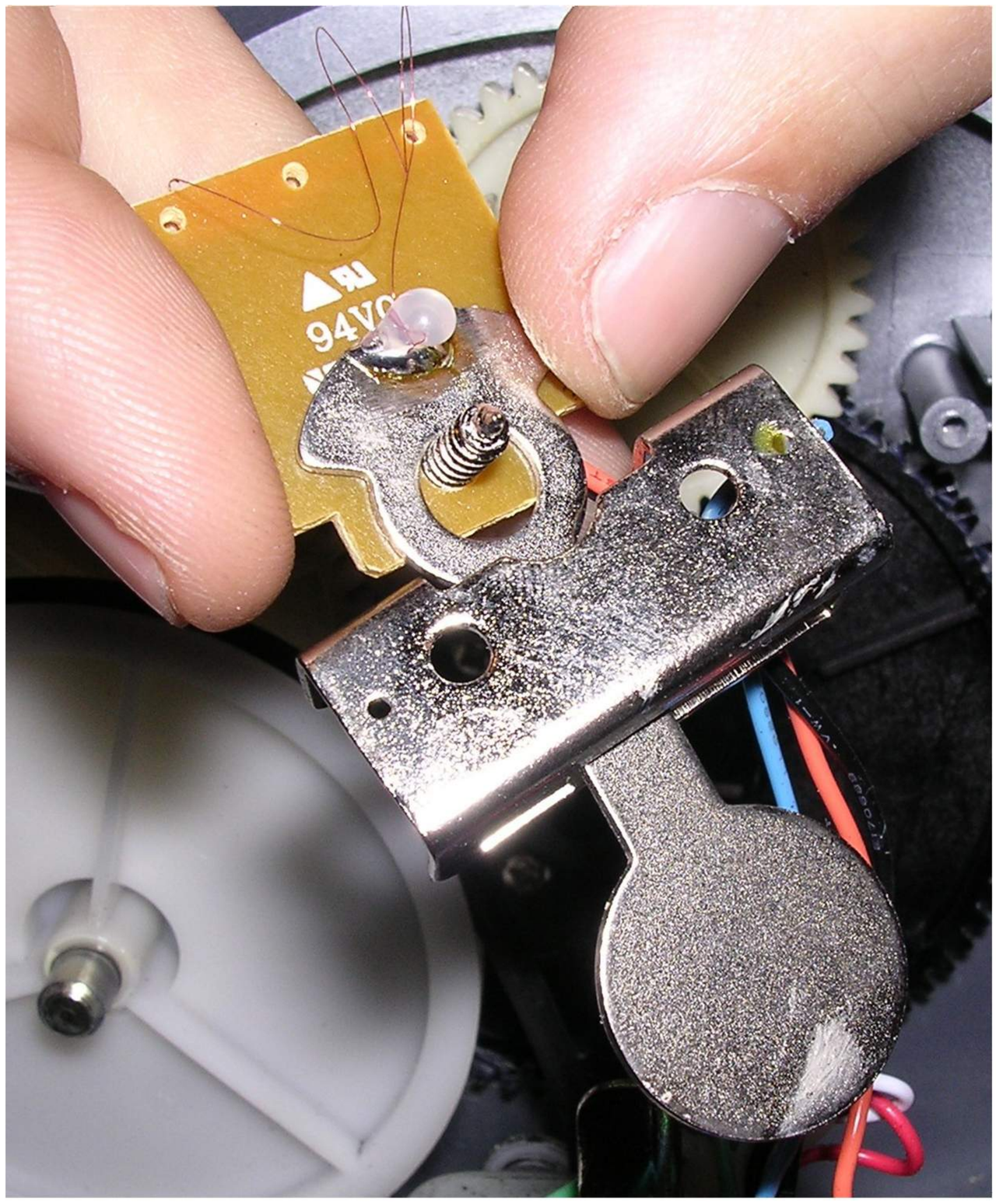


Inside, are the drive gears and motor for raising and lowering the right arm. The interrupter for the encoder disc is visible at the top of the photo at right (tan PCB). The disc itself is behind another panel. At 10 o'clock in this photo, there's a tilt sensor consisting of a pendulum and cage.

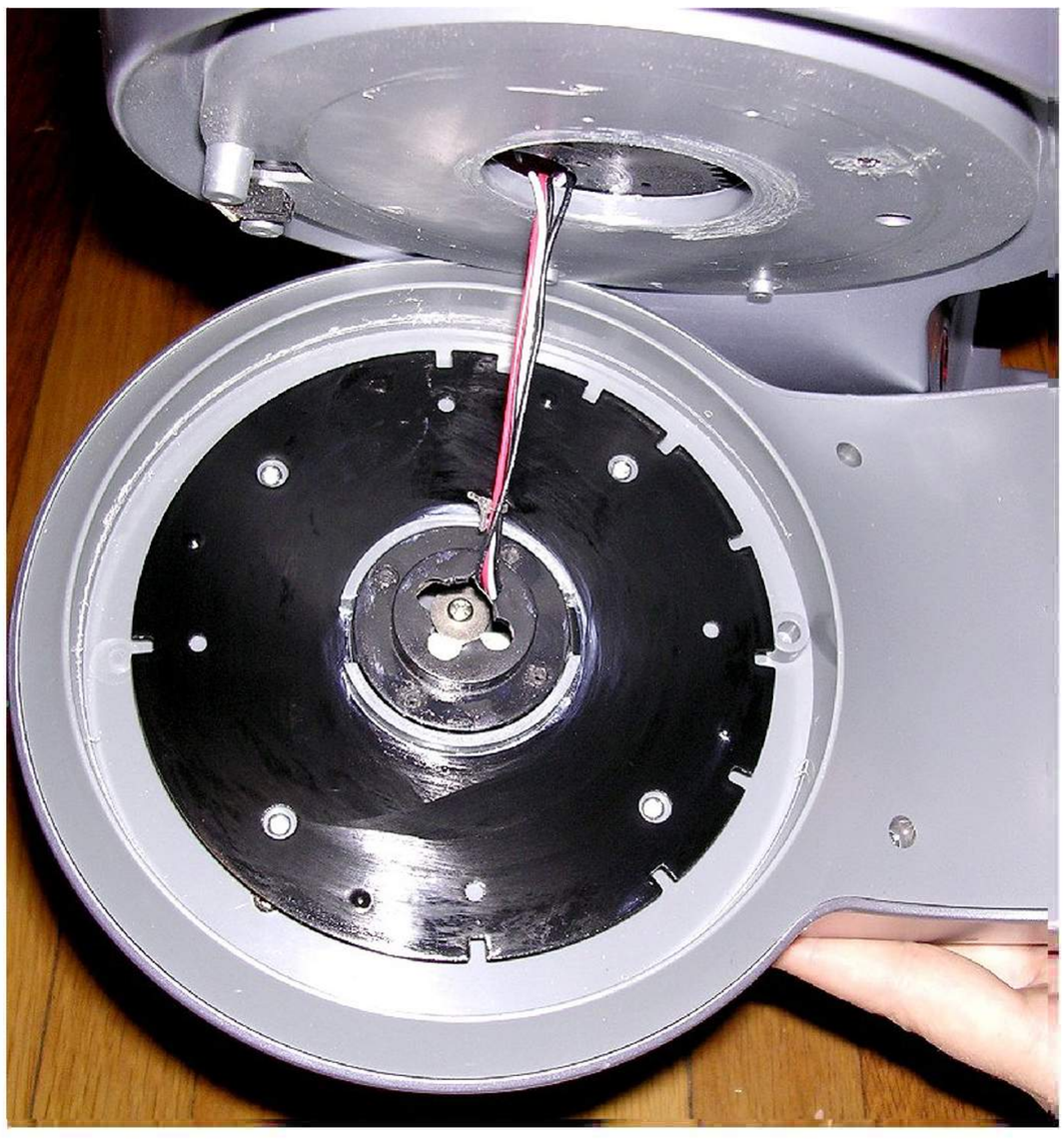


The photos below show close-ups of the front and back of the tilt sensor. A coiled hair wire allows current flow through the pendulum bob without inhibiting its motion.





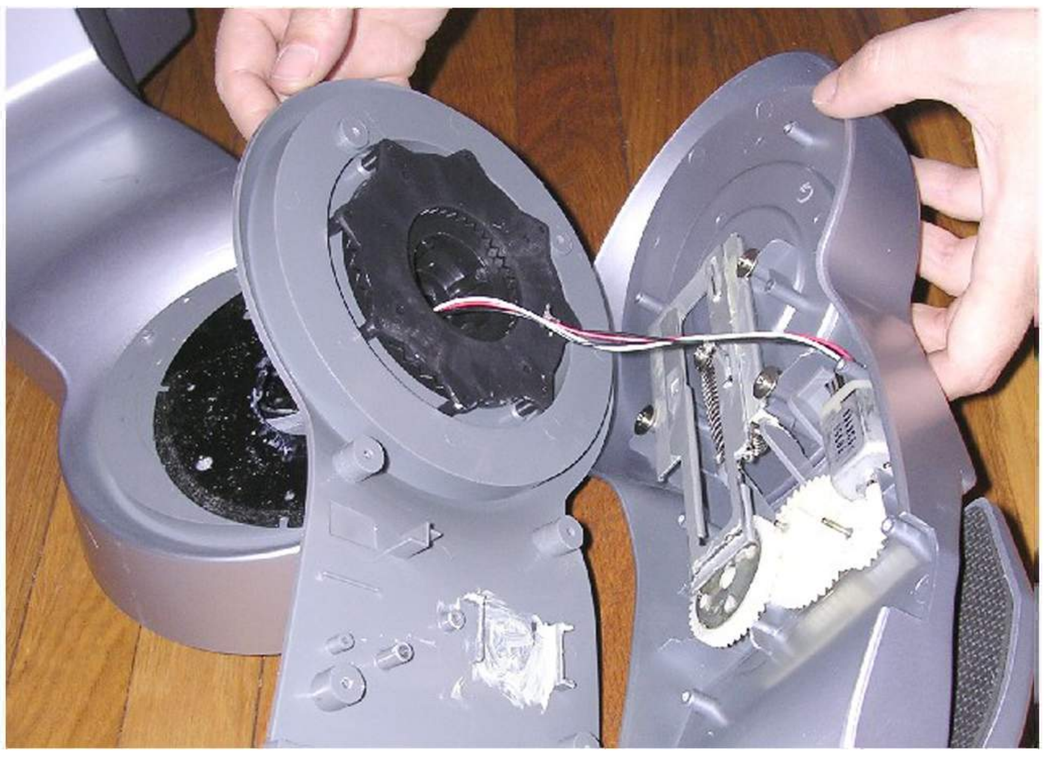
Detach the Arm. Separate the right arm from the right leg by removing 4 phillips screws from the black center gear. Pull the arm away from the leg. The encoder disk for arm rotation is located between the arm and leg assemblies (photo at right).



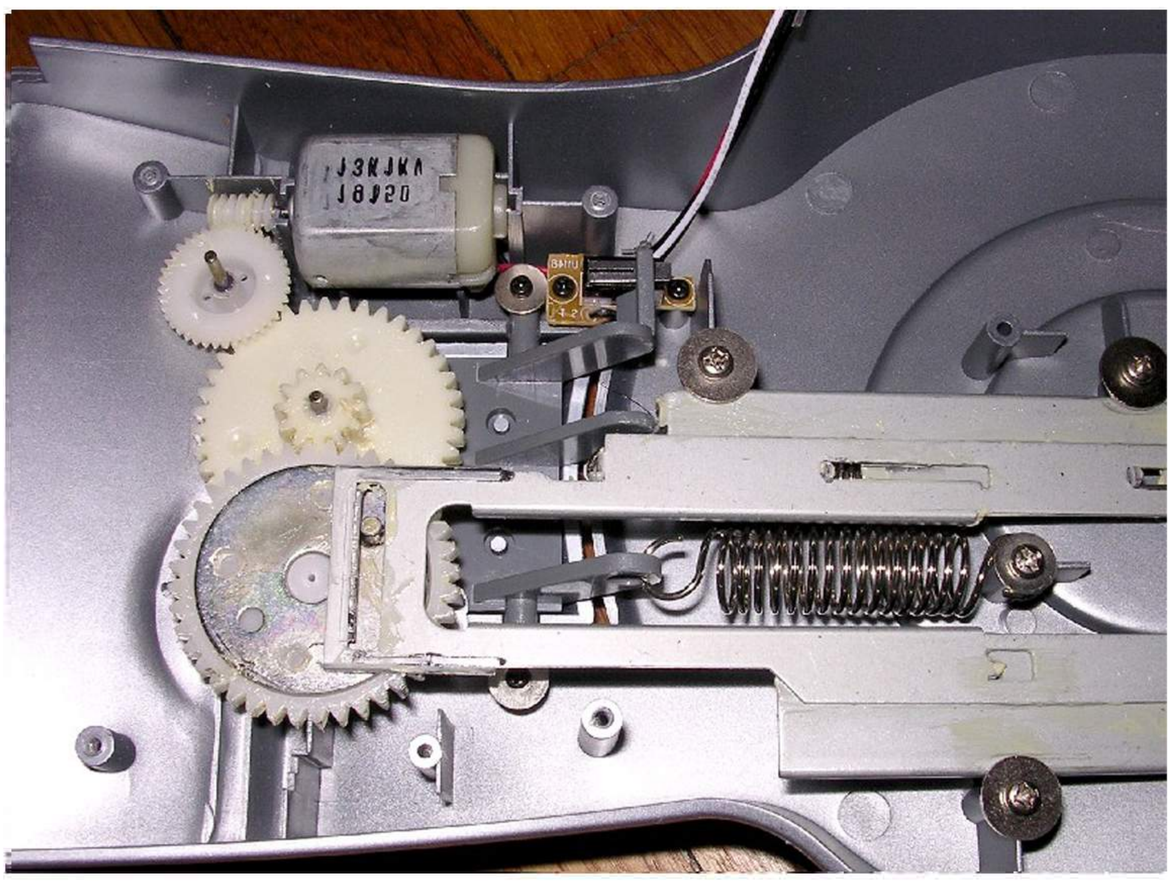
Open Right Arm

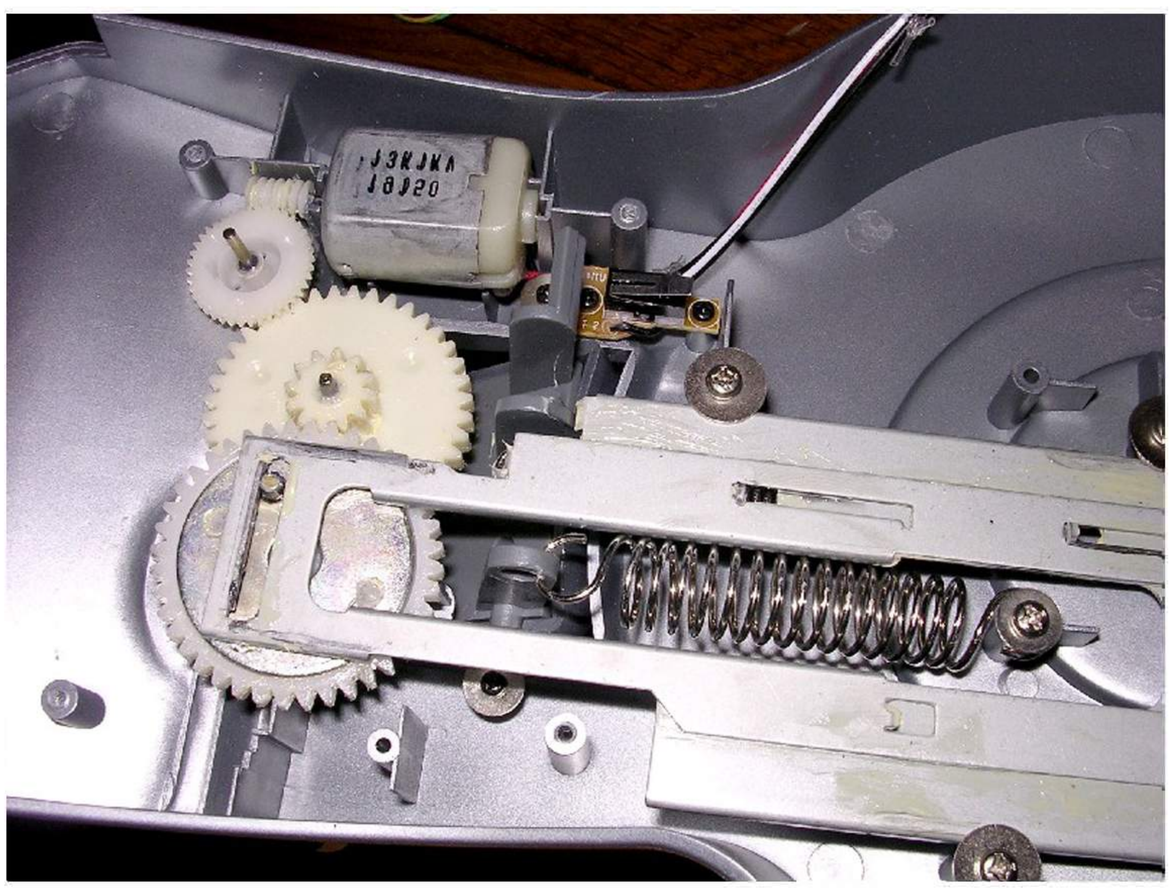
This part is shown in page 8 of the Assembly Diagram.

Open the Arm. Lift the arm's encoder disk and remove the 7 phillips screws that hold together the two arm halves.



The mechanism for opening and closing the hand is shown below. The hand is tensioned towards its closed position by a pair of springs, one of which is visible in the photos. A motor with worm gear rotates the gear train. The final gear has an offset pin that oscillates as the final gear rotates, alternately opening and closing the hand. A microswitch next to the motor detects when the hand is closed.



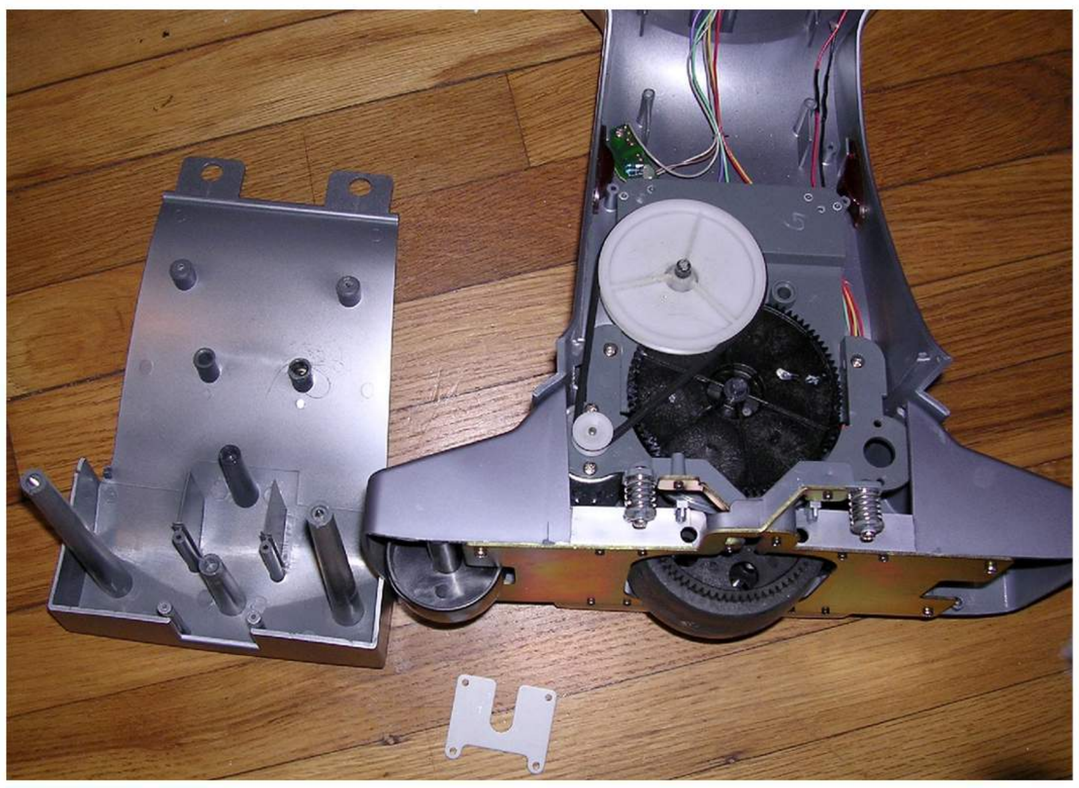


Open Right Leg

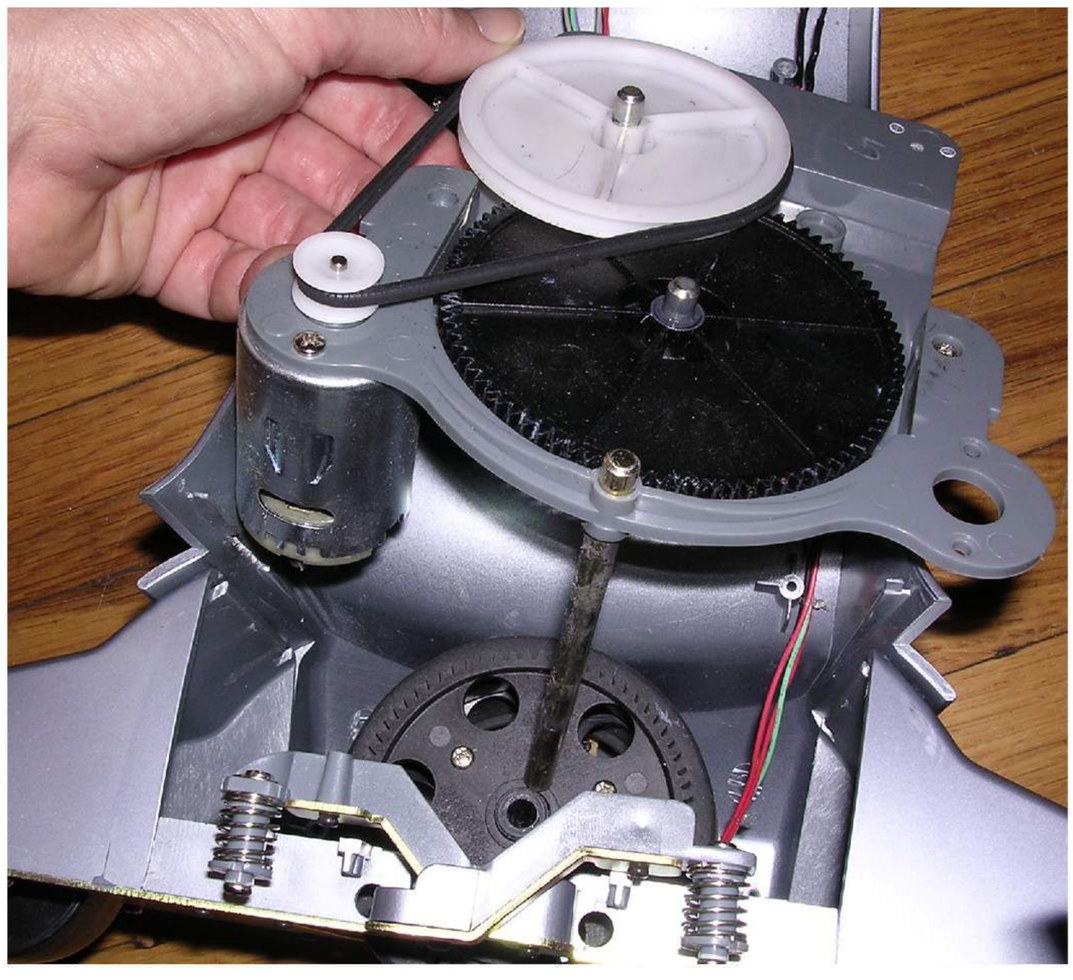
This part is shown in page 8 of the Assembly Diagram.

Remove Bottom Bracket. Remove the inside bracket from the bottom of the right leg. The 2 phillips screws securing this bracket are hard to reach with a screwdriver. A small pair of pliers works well, though.

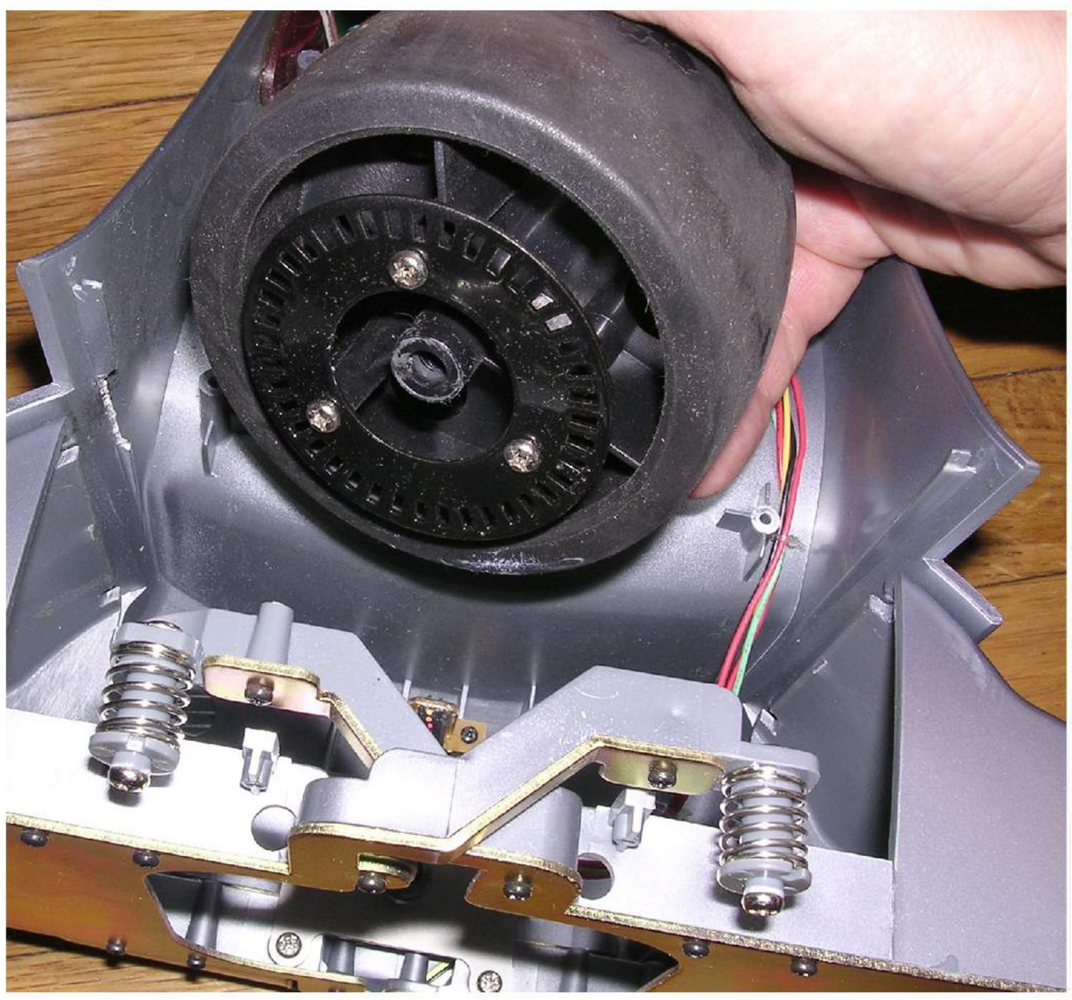
Open the Leg. Remove the 4 phillips screws securing the inside leg panel. Two of these screws are deeply recessed. You'll need a long (6") #1 phillips to unscrew them.

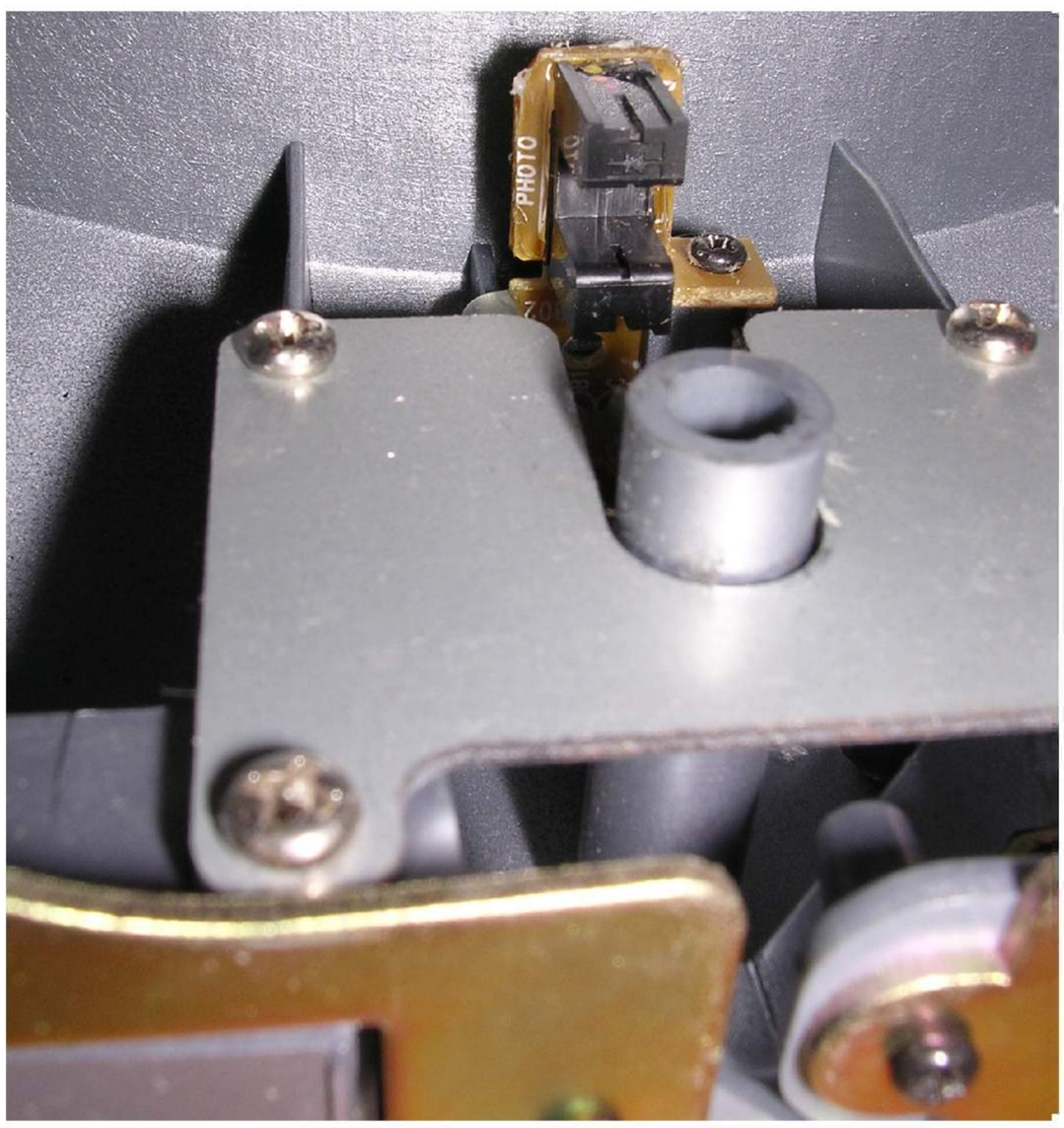


Remove Drive-Wheel Gearing. Remove the 4 phillips screws securing the gearing and motor for the drive wheel and lift out the assembly.

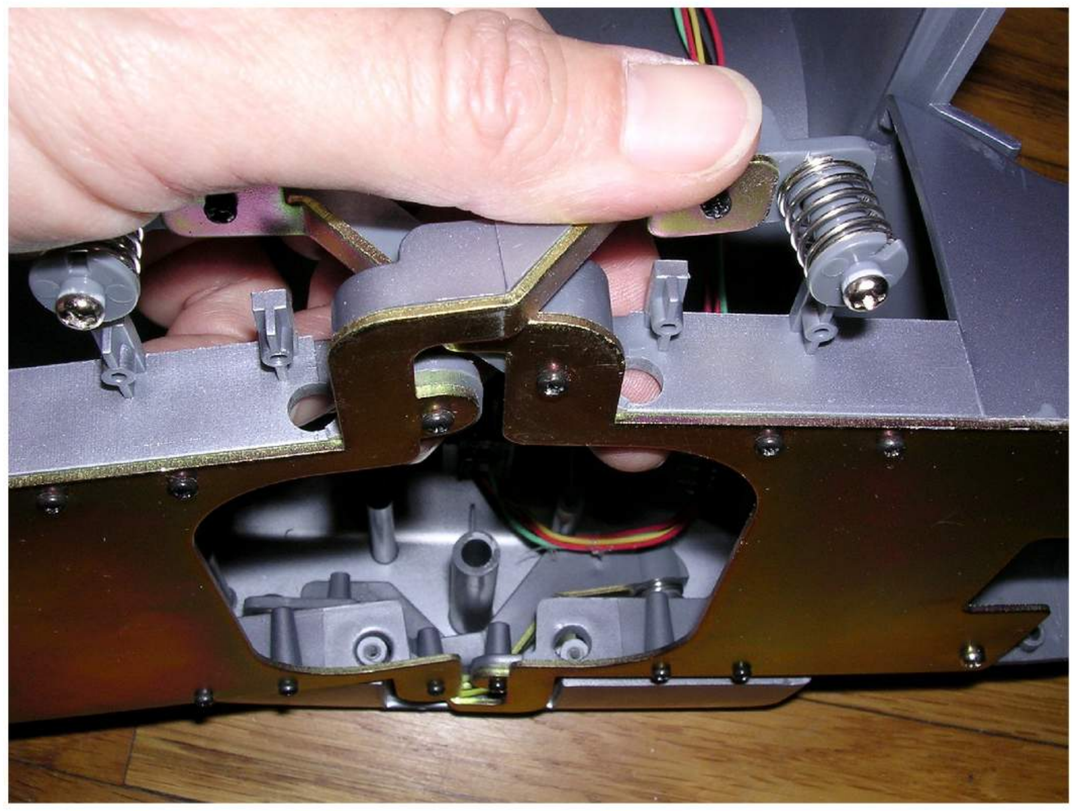


Lift Out Drive Wheel. The drive wheel simply lifts out once the geartrain is removed. It has an attached encoder disk. The encoder's interrupter sensor is located near the outside of the foot. Wheel and sensor are shown below.





Detach Toes. The toes rotate on pivot pins in the inner and outer cover panels. They can be removed by unscrewing the bracket beneath the interrupter sensor and lifting them out.



Microcontrollers

There are three PIC microcontrollers on the main board. Presumably, these control all or most of RoboScout's behavior. Surprisingly, these chips were not code protected, so I used a Warp 13 programmer to extract the program memories as hex files. I found a freeware disassembler at Talking Electronics. The disassembler is specifically for the PIC16F84, but the instruction set should be the same, since all are mid-range chips. With a few manipulations (see Disassembly Notes), I got assembly-language source files that look plausible and are acceptable to MPLAB.

One feasible and useful thing to do with raw assembly code is to see how pins are defined. For multiple-function pins, I searched for control-register settings that specify how these pins are used. For I/O pins, I looked for TRIS settings to determine I/O directions. I also searched for all I/O pins that are explicitly tested or written and for whole-port reads and writes.

I've started to analyze the source code from microcontrollers A and C. I haven't done anything yet with the source code from chip B, except extract and post it.

I'm not a PIC expert, so it's very possible I've made mistakes or have overlooked things.

Sections:

Microcontroller A

Description. Microcontroller A is located on the left side of the main board. On my RoboScout, it's labelled A (2.0). Presumably 2.0 is the firmware version. The chip is a PIC16C76.

Disassembly Files:

- Hex file from Warp 13
- Modified hex file ready for disassembly
- Output from disassembler
- Final product - source file that's acceptable to MPLAB

Source Code Analysis. The table below summarizes the pin usages I found in this chip's source code. Symbols are from Microchip's Mid-Range Reference Manual and the chip's datasheet. I skipped pins like /MCLR and VSS that aren't set in source code. I'm uncertain about UART usage. It appears that the UART is configured, but I didn't see the receive or transmit buffers being used anywhere in the code.

Pin	Function	Description	Used?
2	AN0	Analog Input	Yes
3	RA1	Output	?
4	RA2	Input	No
5	RA3	Output	?
6	RA4	Input	Yes
7	RA5	Output	?
11	RC0	Input	?
12	T1OSI	Counter 1, Input	Yes
13	CCP1	PWM	Yes

14	SCL	I2C Clock	Yes
15	SDA	I2C I/O	Yes
16	RC5	Output	?
17	TX (?)	UART Transmit (?)	?
18	RX (?)	UART Receive (?)	?
21	RB0	Input	?
22	RB1	Input	Yes
23	RB2	Input	?
24	RB3	Input	Yes
25	RB4	Input	Yes
26	RB5	Input	Yes
27	RB6	Input	Yes
28	RB7	Input	Yes

I also traced a little of the code structure. All my analysis notes are in this file: Source Code With Some Added Comments.

Microcontroller B

Description. Microcontroller B is located centrally on the main board. On my RoboScout, it's labelled B (2.0). Presumably 2.0 is the firmware version. The chip is a PIC16C67.

Disassembly Files:

- Hex file from Warp 13
- Modified hex file ready for disassembly
- Output from disassembler
- Final product - source file that's acceptable to MPLAB

Source Code Analysis. The table below summarizes the pin usages I found in the source code. Symbols are from Microchip's Mid-Range Reference Manual and the chip's datasheet. I skipped pins like /MCLR and VSS that aren't set in source code.

Pin	Function	Description	Used?
2	RA0	Input	Yes

I also traced a little of the code structure. All my analysis notes are in this file: Source Code With Some Added Comments.

Microcontroller C

Description. Microcontroller C is located on the right side of the main board. On my RoboScout, it's labelled C (1.5). Presumably 1.5 is the firmware version. The chip is a PIC16C63A.

Disassembly Files:

- Hex file from Warp 13
- Modified hex file ready for disassembly
- Output from disassembler
- Final product - source file that's acceptable to MPLAB

Source Code Analysis. The table below summarizes the pin usages I found in the source code. Symbols are from Microchip's Mid-Range Reference Manual and the chip's datasheet. I skipped pins like /MCLR and VSS that aren't set in source code.

Pin	Function	Description	Used?
2	RA0	Input	Yes
3	RA1	Input	?
4	RA2	Input	?
5	RA3	Output	Yes
6	RA4	Input	Yes
7	RA5	Input	?
11	RC0	Input	No
12	RC1	Output	?
13	CCP1	PWM	Yes
14	SCL	I2C Clock	Yes

15	SDA	I2C I/O	Yes
16	RC5	Output	Yes
17	RC6	Output	Yes
18	RC7	Input	Yes
21	RB0	Input	Yes
22	RB1	Output	Yes
23	RB2	Output	Yes
24	RB3	Input	Yes
25	RB4	Output	?
26	RB5	Output	?
27	RB6	Output	?
28	RB7	Output	?

I also traced a little of the code structure, mostly to validate that the disassembly gave a reasonable result. All my analysis notes are in this file:
Source Code With Some Added Comments.

Disassembly Notes

Disassembling the microcontroller code required some manipulation. First problem was that the hex file from Warp 13 appeared to be in INHX32 format. The disassembler didn't like that. The example hex files for the disassembler seem to be in INHX8M format, so I converted the Warp 13 file by deleting all the control lines. I also deleted all the unprogrammed lines at the end to create a smaller file.

Second problem was that the first program word in my hex file for Microcontroller C is 8A81 (LSB-MSB format). This can't be an instruction, since the pic's program counter is only 14 bits wide. I don't know what it is, possibly the configuration word? The remaining program words were all valid, however, so I zeroed out this word in the hex code for all three microcontrollers to create a NOP as the first instruction.

Third, the disassembler had problems with certain line references. The lines are all labelled as A0###, where ### is the line number. GOTO and CALL instructions reference these labels. That's all fine. But some instructions came out like this: GOTO A0-8##. For a while, I thought this problem happened only with one label pattern, and I used a "Replace All" to get a syntactically-correct source file. I've since discovered, however, that the problem occurs whenever the destination is any label above A01FF. To fix it, I wrote a simple postprocessor program that looks for this problem, pulls out the hex for the opcode, disassembles that one instruction, and inserts it into the assembly-language file.

Useful Links:

- Warp 13 Programmer
- Hex Formats
- Talking Electronics' Disassembler
- Microchip's Mid-Range Reference Manual