Hero Robot Frequently Asked Questions

Table of Contents

Copyright

New Updated Unchanged External Link

1.0 General Information

```
Heathkit had about 8 years of sales: 4,000 Hero Jr's, 3,000
Hero 2000's, 14,000 assembled Hero 1's
```

1.1 Some HERO TLAs:

```
ET stands for Educational Trainer (kit form)
ETW stands for Educational Trainer (Wired by Heath)
ETA stands for Educational Trainer (Assembled)
RTA stands for Robot Accessory (Used for HERO JR Options)
RTC stands for Robot Cartridge (Used for HERO JR)
```

1.2 Is there a mail list/news group for the Hero Robots?

1.2.1 How do I subscribe?

```
Send email to:
    hero-owners-request@homerobots.com

with the first line in the body of the message:
   SUBSCRIBE hero-owners 'address'
```

1.2.2 How do I unsubscribe?

```
Send email to:
    hero-owners-request@homerobots.com

with the first line in the body of the message:
    UNSUBSCRIBE hero-owners 'address'
```

1.2.3 How do I send email to it?

```
Send it as normal email with the To or Cc address set to:
        hero-owners@homerobots.com
```

1.3 How many different robots did Heath Co. make?

```
Heathkit made four Robot Models, they are original HERO 1,
the HERO JR, the HERO 2000, and the Arm Trainer.  The HERO 1 was
the first and most popular robot model.  The HERO JR came out
after the HERO 1 and was created by the consumer division as a
personal robot instead of an educational one.  This is one of the
reasons that there is not nearly as much technical information
about the JR.  They borrowed heavily from the HERO 1 robot, but
made some changes so that it would be much less expensive and
simpler to operate.  The HERO 1, HERO 2000, and the Arm Trainer
were all made by the educational division and had much more
documentation and expandability.  Both the HERO 1 and HERO JR are
very much alike, and even share many of the same components.
The HERO 2000 and the Arm Trainer share the same Arm and use the
same BASIC commands for movement that the HERO 2000 uses, but it
is mounted on a stationary base and uses a Z80 microprocessor as
the main CPU instead of the 8088 like the HERO 2000 robot.
```

1.4 Are there any hero robots for sale; How can I find one?

Check the Hero SwapShop or ask on your favorite forsale news group. Hero-1's tend to sell somewhere around $500 (more or less depending on condition and options). Hero-Jrs have been seen going for as little as $100. Hero-2000 tend to start somewhere around $800 and go up in price - again depending on its condition and options.

2.0 Specs for the Heathkit HERO robots.

2.1 ET-18-A HERO 1

2.1.1 Description

An ideal teaching tool, HERO 1 is a completely self-contained robot that interacts with you and its environment. It's excellent for learning the components and circuitry of robots, as well as artificial intelligence.

HERO 1 has incredible sensing capabilities. It detects sound, light, motion, and obstacles and travels over a course that you predetermine. The robot can see and hear through on board light, sound, and motion detectors, plus a sonar ranging system. The sound detector hears over a 200 to 5000 Hz frequency range while the light detector sees over the entire visible spectrum and into the infrared range. HERO 1's motion detector senses movement up to a distance of 15 feet while a sonar system determines direction and ranges between objects and the robot from 4 inches to 8 feet away. An optical encoder provides precise measurements of distance traveled.

To make HERO 1 even more life-like, the head rotates 350 degrees to position sensors and arm. Seven stepper motors control all movements. Plus powerful front wheel DC motors drive and steer HERO 1 with exceptional maneuverability.

On-board HERO 1 is an 8-bit 6808 microprocessor that can guide the robot through complex maneuvers, activate and monitor sensors, and modify actions as a result of sensor or real-time clock inputs. HERO 1 also has 8K of ROM as well as 4K of RAM. A top-mounted breadboard lets you conduct experiments and interface circuits of your own design to the on-board microprocessor. Plus HERO 1 comes with a six 7-segment LED display for viewing memory addresses, data and program steps.

Commanding HERO 1 is easy with four different methods available to you. A top-mounted 17-key hexadecimal keyboard lets you easily enter, verify and modify programs, and select operating modes. An attachable Teaching Pendant lets you manually control all motor and arm movements, or store them for later duplication. Or you can control HERO 1 with an optional remote, radio frequency controlled transmitter available in two models, each operating at a different frequency. You can also guide HERO 1 by directly linking a host computer with the help of the optional Memory Expansion Board and RS-232 Interface that plugs into the top experimental HERO 1 breadboard.

2.1.2 Specs

HERO 1 is powered by four gel cell rechargeable batteries. A 120/240 VAC, 50/60 Hz charger is included. HERO 1 is 20" high, 18" in diameter and weights 39 lbs. with accessories.

The HERO 1 uses Stepper motors for everything except for the main drive motor. It also has separate boards for each function.

- 6808 CPU (8-bit)
- 4K of RAM (2 6116 static RAM chips)
- 8K of ROM (With Robot monitor)
- Experimenter Board on the head
- Sonar ranging on head
- Ultrasonic motion detector on head
- light sensor on head
- sound sensor on head
- Cassette I/O for program storage
- head rotates 350 degrees
- 2 fixed wheels, 1 steerable drive wheel (it's in front)
- 6 Seven segment LED displays
- 17 digit Hex keypad (with Real keys) on head
- Teaching pendant
- Charger

2.1.3 Available Options

ET-18-1	Robot Arm 5 Axis (16 ounces max)
ET-18-2	Speech (SC-01 Based)

ET-18-35	Remote control (75.43Mhz) with RS-232
ET-18-36	Remote control (75.67Mhz) with RS-232
ET-18-4	DEMO ROM Sings songs, shows off
ET-18-5	Monitor ROM Listing
ET-18-6	Memory Expansion board
ET-18-7	Auto Mode ROM Lets HERO 1 navigate a room
ET-18-9	BASIC ROM Requires Memory Expansion
ETW-18-10	RS-232 serial interface (Plugs in Experimenter board on head)
ET-18-11	DEMO cassette (More Demos)

2.2 RT-1 HERO JR

2.2.1 Description

Meet the first affordable, personal robot with a dynamic personality. HERO JR. A friendly robot, HERO JR will fit right in with your family and into your home. It sings songs, plays games, tells nursery rhymes, recites poems, guards your home and can even wake you in the morning and guide you through an exercise routine. HERO JR explores its surroundings and seeks to remain near human companions. HERO JR's personality is preprogrammed and doesn't require computer programming skills to operate. If you do wish to program HERO JR, you can with a home computer and optional RS-232 Accessory and BASIC Cartridge.

The traits comprising HERO JR's personality include: singing songs like "Daisy" and "America"; saying preprogrammed phrases; exploring and moving about, using sensors to avoid obstacles and seek out humans; playing games; telling nursery rhymes; and gabbing in "Roblish." All routines are stored in 32k ROM. Add more functions with optional plug-in cartridges. In addition to these six traits, HERO JR has an internal clock with a 100-year calendar that can even compensate for Daylight Savings Time. With this time-keeping ability, HERO JR can act as an alarm clock and awaken you at a specified time. After your wake-up call, it listens to be sure you are awake or permits two ten minute snoozes if you desire. It can also be made to announce the week day, date and time at set intervals or just at one important time and date. With 2k of RAM, HERO JR can store and remind you of events such as birthdays and anniversaries - up to 16 dates per year.

A DEMO feature activates a built-in "Robot Variety Show" which demonstrates the Robot's ability to see, hear, speak, tell time and move. You can even participate in the show. To see, hear, speak, and to move about, HERO JR uses a light sensor, ultrasonic sonar, sound detector, speech synthesizer and an internal clock which are controlled by an on-board computer using a Motorola 6808 microprocessor. The sonar is accurate from 4.5" to 13'. An optional infrared sensor adds superior heat/motion detection. HERO JR duplicates all English sounds using its Votrax SC-01 phoneme synthesizer. This permits the Robot to say just about anything. Volume and pitch adjustments tailor the Robot's voice to your liking. All of HERO JR's vocabulary is preprogrammed.

HERO JR can guard your home against intruders and can be used with the GDA-2800 Security System. The Robot give a verbal warning and asks for a password when its sensors detect a presence. The wrong or no response causes HERO JR to activate the security system. HERO JR normally speaks, sings and performs tasks between moving randomly about, however, an optional wireless remote unit allows the Robot to be manually driven from place to place. It will speak while under remote control. Remote operates and 75Mhz up to 100'.

HERO JR is powered by two six-volt rechargeable batteries and plug in wall charger (included). HERO JR operates about 4 hours, with a normal amount of exploring. The Robot will randomly enter a sleep mode which conserves battery power by keeping only critical circuits energized. You can also make HERO JR enter or leave this mode. HERO JR can carry your favorite beverage (up to 10 lbs.) in a 94 cubic inch compartment. HERO JR has a 17-key keypad which lets you modify its personality to initiate a task. The keypad has clearly marked function keys including Sing, Play, Poet, Gab, Alarm, Guard, Help, Plan, Setup, and Enter. Eight LED's flash in time with speech. HERO JR id 19" tall and is easily assembled in about 20 hours.

Every thing is built on two boards (Main CPU board and Power sense board)

One side note: HERO JR drives backwards to HERO 1. The drive platform is pretty much identical, but forward is the other way. I have heard the Heathkit reversed the platform because it have better control and the steering was more accurate.

2.2.2 Specs

```
6808 CPU (8-bit)
2K of RAM (6116 static RAM chip) Expandable to 24k
32K of ROM (With Robot monitor)
Speech built in (SC-01 based)
Sonar ranging on head (Polaroid)
light sensor on head
sound sensor on head
head does not rotate
2 fixed wheels, 1 steerable drive wheel (it's in back)
9 LED's for a display
17 digit Hex keypad (with Rubber keys) on head
Charger
```

2.2.3 Available Options

```
RTA-1-1         Infra-red motion detector on head range about
                35' long x 20' wide
RTA-1-2         Remote control accessory offers manual, wireless
                control of HERO JR's movements up to 100' away,
                operates at 75Mhz.
RTA-1-3         RS232 interface allows use of the BASIC cartridge
RTA-1-4         Two extra batteries (doubles the capacity)
RTA-1-5         Cartridge Adapter, permits use of plug in
                cartridges, with 8k RAM
Plus RF Transmitter to set off Home security system
```

2.2.4 Plug in cartridges

```
RTC-1-1         Trivia Quotes (Trivia game)
RTC-1-2         Songs, phrases and Rhymes #1 (Adds to personality)
RTC-1-3         Animals, Blackjack, and Tic Tac Toe
RTC-1-4         Special Occasions (Jungle Bells, Auld Lang Syne)
RTC-1-5         Math Master a timed math game
RTC-1-6         Riddle Robot/Tongue Twister
RTC-1-7         Philosopher
RTC-1-8         HERO JR BASIC (program the HERO JR with simple
                BASIC commands)
RTC-1-9         HEROBICS (10 Exercises, 4 levels of difficulty)
RTC-1-10        HERO JR Program Language (Program HERO JR with the
                keypad)
RTC-1-11        Musical Chairs with Acey-Ducey and Robot mind
                reader
```

2.3 ET-19 HERO 2000

2.3.1 Description

The first of the next generation of training robots. HERO 2000 will help you explore the related technologies of robot automation programming, electronics for automation, intelligent machines and robotics. And it does it better than any other training aid available.

HERO 2000 features advanced programmability. This incredible robot has a 16-bit 8088 master microprocessor that runs user programs and eleven 8-bit peripheral microprocessors. This lets you simultaneously operate HERO 2000's sensory and manipulative functions. There's also HERO 2000 BASIC stored in the 64K ROM, while other built-in programs are for demonstrations, diagnostics and sensor adjustment routines. What's more, HERO 2000 can even write its own BASIC program.

HERO 2000 has unbelievable expandability. Besides having amazing computational abilities, HERO 2000 features 24K of RAM expandable to 576K - with the use of three optional memory boards. Each memory board contains 64K of RAM and will accept two additional 64K RAM chip sets.

The new disk drive accessory (optional) expands Hero's capabilities by allowing you to store and retrieve programs and data on a 5 1/4" floppy disk. The drive includes the MS-DOS operating system and requires the ETW-19-15 Expansion Board with a minimum of 128K RAM installed for use with your HERO 2000.

Other expansion features include 12 circuit card slots, an optional Experimenter Board for conducting experiments and testing your own circuits, two RS-232 DCE ports configured for a terminal and printer, and a cassette port for storing your programs on standard cassette tapes.

HERO possesses unsurpassed flexibility. Over a dozen programs

are available on the optional demonstration ROM to show you the versatility of the electronically synthesized voice and built-in sensors. The synthesized voice lets HERO talk, play music and even emit sound effects. And for easy programming, HERO 2000 has direct text-to-speech conversion. The built-in sensors include: a 360 degree sonar with a range of 4.5 inches to 10.5 feet; a light scanner with 255-level coverage at 24 bearings, 15 degrees apart; a temperature sensor that covers from +60 degrees F to +90 degrees F; and a sound sensor that picks up 255 audio levels.

And for maneuverability and manual dexterity, HERO 2000 is second to none. The optional robot arm is multi-jointed with a gripper that has a sense of touch. The arm lifts one pound in any direction, plus has full wrist action and accurate repeatability. Additionally, a powerful two-wheel dual servo-motor drive system pulls up to 26 pounds, while an optional auto docking accessory enables HERO 2000 to automatically dock with its charger when a low battery is detected. With all these abilities, you'll think this robot is almost human!

Everything's totally under your control. You'll be able to access ROM routines and perform movements from the top-mounted hexadecimal keypad or the optional wireless remote control. The remote control console features a typewriter-type ASCII keyboard, 80-character LCD display and teaching pendant controls. This incredible console permits wireless control and programming of all functions up to 100 feet away. There's even a two-way radio data link between console and robot. HERO 2000 requires the Remote Console or a computer terminal for programming and checkout routines.

Other sensational features include sixteen head-mounted LED status indicators (eight are user definable) to show you what function HERO is performing. And with three power modes, HERO 2000 is able to operate up to six days on its single 24 amp-hour battery. A 120 VAC charger is also included.

HERO 2000 - by far the smartest, most versatile and easy to use robot trainer around.

2.3.2 Specs

Main processor Intel 8088 CPU (16-bit)
6 slave processors (11 with optional Arm)
24K RAM (expandable to 576k with optional Memory cards)
64K ROM with built in BASIC (Full version of BASIC)
Speech SPA-256
360 degree sonar (24 bearings) and light sensor (255 levels)
Stationary sonar range finder in the base
Two RS-232 ports (one for a Terminal and the other for a printer)
Passive backplane supports up to 12 Cards

2.3.3 Available Options

ET-19-3	Robot carrying cart (Has anyone seen this????)
ET-19-35	Two-way Remote control 75.43Mhz, full keyboard, teaching pendant, 80-character LCD display, and RS-232 port
ET-19-36	Same as above except 75.67Mhz
ET-19-5	Auto-Dock (Find charger on low battery)
ET-19-51	Demo ROM (Some cool demos)
ET-19-14	Experimenter card dual breadboard and buffered I/O
ETW-19-15	Static memory card 192k max on each
ETW-19-6	360K Floppy disk drive and controller

2.4 ETS-19-32 The Arm Trainer

2.4.1 Description

The Robot Arm Trainer is designed to simulate the operation of full scale industrial robots. It's the most effective and complete learning aid available. And it's simple to understand and operate, so you can spend your time learning by doing.

The Trainer features five axes of motion, including wrist pitch and roll for precise pick-and-place movements. The Arm can lift 1 lb., Gripper force is 0 to 3 3/4 lbs., while the gripper opening is 4 3/8". There's also a patented sense-of-touch gripper that is programmable to exert specific amounts of force. Range of motion for the torso is -180 to 170 degrees; for the arm, 0 to 140 degrees; for the elbow, 0 to 180 degrees; for the wrist pitch, -90 to 180 degrees; for wrist roll, -180 to 180 degrees; and for the gripper, 0 to 8 arbitrary units. The axes speed of motion for the torso is 8 to 33 degrees per second; for the Arm and elbow, 3 to

15 degrees per second; for the wrist pitch and roll 15 to 70
degrees per second; and for the gripper, 1.8 degrees per second.

An on-board 8-bit CPU and six slave microprocessors control
simultaneous operation of all the Trainers functions. No external
computer is required for operation.

Six closed-loop DC servo-motors provide smooth motion and
accurately move the Arm through all its axes. The closed-loop
system monitors both speed and position of each servo-motor and
reports to the controlling microprocessor.

In addition, the Trainer has menu-driven software that's
programmed to provide you with command options. It leads you
through the Trainers operations and eliminates the need for
constant reference to manuals. You can also program the Trainer
to do tasks consisting of up to 50 steps and even link tasks to
create larger application programs. And it's easy to simulate
real industrial applications with the Trainers exact
repeatability and accuracy.

The Robot Arm Trainer also has protective features that make it
safe for beginners or trainees. The top-mounted Emergency Stop
Switch interrupts the Trainer instantly without aborting the task
in progress. Reset the switch and the Trainer resumes the task at
the point of interruption. Also, built-in software limits ensure
that the Trainer won't be damaged by exceeding its physical
limitations, even if commanded to do so. Protective shrouds keep
you away from the Trainers moving parts.

Also included in this practical and comprehensive trainer package
is 48KB RAM, 32KB ROM, and a Z80A control CPU operating at 4 Mhz.
The Robot Arm Trainer also comes with an industrial-type teaching
pendant with a 17-key membrane pad and 16-character LCD readout,
and a built-in RS-232 port to interface with a terminal or an
external computer, allowing you to create, edit, and store
programmed tasks at a computer terminal.

An optional cassette interface accessory permits you to store
programs on standard cassette tapes. In addition, an 8-channel
status port can send and receive data signals, permitting the
Robot Arm Trainer to communicate with external sensors and other
robotic devices. An internal set of diagnostic routines aid in
alignment procedures and provide system checks.

Now you can learn the fundamentals of robotics in industry and
turn your understanding of robotics theory into hands-on
experience with the Robot Arm Trainer. Trainer includes arm and
base - both built to last.

2.4.2 Specs

```
Main processor Z80A @ 4Mhz
6 slave processors (one for each servo motor)
48K RAM
32K ROM (16K Each ROM, it is bank switched as needed)
17 key membrane teaching pendant with 16-character LCD display
RS-232 serial port
8 bit parallel I/O port
```

2.4.3 Available Options

```
ETW-19-31       Cassette interface
```

2.5 Batteries

2.5.1 Original Batteries

```
Hero-1:         4 6V 4.0Ah rechargeable batteries
Hero-Jr:        2 6V 4.0Ah rechargeable batteries
Hero-2000:      1 12V 24Ah rechargeable battery
```

2.5.2 Optional 12V usage

The batteries I got with the Hero were pretty much cooked. I couldn't
find a good source of cheap 6V 4AH gel cells, so I changed them for
12v 4AH cells. These are used heavily in alarm system, and can be bought
from most security companies.

The motor power is ONLY 12v, so to install this battery, all you do is
remove the jumper wire, and connect it directly to the remaining + and
- connections.

```
The logic power supply is a bit more trouble. First, the negative
connection is way up in the head. I just added another connection to
the frame of the robot (Be sure to leave the negative connection in the
head connected to ground to insure that the head and frame are
electrically bonded together. The yellow wire jumpering the two original
batteries also had a tap drawing 6V from it. This is only used for the
CMOS clock, which draws about 25 microamps. You can either install a
small 6V battery in the head to power this lead, place a 6V zener between
the robot side of the 12v battery fuse (inside the frame) and the yellow
wire going to the I/O board.

The lower battery compartment is large enough to hold two of the 12V
batteries (with room for the switch/power/pendant wires between them),
but you can't get them in and out past the switch holder. I solved this
by replacing one of the bolts holding the switch holder with a nurled
bolt which can be easily removed, allowing the switch holder to swing
out on one end. While I was at it, I replaced the bolts holding the
door to the insides of the robot with nurled ones as well.
```

3.0 In times of need

3.1 Where can I get parts for my Hero?

Currently there is only one sources for manuals parts for the Hero family of robots (that I know of):

- Robots Wanted: Dead or Alive, Whole or Parts.

3.2 My Hero is broke and I cant fix it. What now?

Just as with the parts; there are only two source for repairs/rework:

- Robots Wanted: Dead or Alive, Whole or Parts.

3.3 My Serial I/O adapter is generating incorrect baud rates

```
I believe the only reason for this is because it has a 4.00Mhz
crystal. The original hero CPU runs at 3.59Mhz, and therefore
will not work at the correct baud rates, which are generated
by delay loops within the ROM.

So far, the only timing that I am sure is dependent on clock speed is
the serial I/O, however I would not be surprised if there were other
"delay loops" in the firmware, and would highly recommend that you
insure your robot has the right crystal for the ROM even if you don't
use the serial port.
```

3.4 RF remote control unit with a RS-232 interface

```
I don't have the book for it, but I've figured out that it runs at 300bps,
8 bits no parity, and accepts the commands '0'-'F' to press the corresponding
keypad key, and 'R' to perform a reset. It does not transmit anything on the
RS-232 (in fact, it doesn't even drive the line).

I have not figured out any way to activate the motor control functions of
the RF remote from the RS-232 port. Do you know if this is possible, and
how it is done?
```

3.5 My Hero-1's head binds. What is wrong?

```
I had a problem with my hero's head when I first got it. It would move
reliably in one direction, but seemed to have no power, and the motor
would sometimes just vibrate when moving in the other direction. I
thought it was a mechanical problem, and removed/greased the head
rotation spindle gear etc. No improvement! Poking around with a scope
I discovered the waveform to one stepper winding was severely distorted,
and traced it to a cold solder joint in the motor driver board. Since I
fixed it I have had no problems, the head rotates about 340 degrees
(goes right to the stops in either direction).
```

3.6 The front wheel spins a lot.

```
Battery placement is almost as important as weight. If you can
move the battery even   a little toward the drive wheel you
will increase its traction and decrease  the resistance of the
other wheels. You might check that the other wheels  are not
bound up. I found that mine had picked up a lot of junk and a
good cleaning helped.
```

3.7 My robot tips over.

There is a 'tipping' problem without the batteries installed,
or by with moving weight (such as batteries) forward to much.
The Hero's front wheel appears to 'shorter' than the rear
wheels giving the unit a forward tilt. When the weight is moved
forward, the front wheel has more traction, however in sharp turns and
stop can result in a fall.

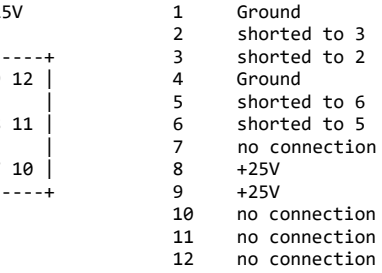
Tipping is not too bad with the original battery configuration
(3 in bottom and 1 in top), and got even better when I went to
two 12v batteries in the bottom (none in the top). I've never
tried to run it without the batteries installed, and it has
never tipped over.

3.8 Does any one have a schematic of the charger for the Hero 1?

The external power supply for the Hero-1 consist of a
transformer which steps the 125VAC down to aprox 20VAC. Its
output is then through R1, R2 to limit the current. This then
goes to a full-wave bridge and is filtered by C1 producing
approximately 25-28Vdc unregulated.

C1 3300uf
R1 2.5ohm 10W
R2 2.5ohm 10W

P101 (the plug) is wired as follows:



1	Ground
2	shorted to 3
3	shorted to 2
4	Ground
5	shorted to 6
6	shorted to 5
7	no connection
8	+25V
9	+25V
10	no connection
11	no connection
12	no connection

3.9 Does any one have a schematic of the charger for the Hero Jr?

The majority of the charger is on board for the Hero-Jr. The
wall part is simply a wall wart transformer with the
appropriate plug.

input output
120 VAC 12VDC @780MA
60 HZ
15 W

Plug wiring
Jump pins 1 to 4
Pin 2 +
Pin 3 -

4.0 Known tech manual errata

Page 24 - Interpreter table
Opcode 55 is defined as "Disable Motion Detector" and 5B as "Disable
Ultrasonic ranging", however these are reversed. Ie: 55 is really
"Disable Ultrasonic", and 5B is "Disable Motion".

Page 25 - Motor control commands
The 'D' bit is defined as 0=Increase 1=Decrease, when in fact these are
also reversed. ie: 0=Increase, and 1=Decrease.

The "abs" MOVE commands indicate "the direction bit 'D' is ignored, since
the motor is given an exact destination". This is not entirely true... the
Drive motor uses 'D' to determine of the robot should move forward or
reverse. Note that the drive motor current position is always zeroed before
a MOVE command, so the "abs" functions really behave like relative ones
when used with the drive motor. Ie: D indicates the direction, and
'ddddddddddd' indicates the distance.

Page 115 - Programmer reference
"READING CLOCK DATA" tells you do STAA $C200 to select the clock register,
when the correct address is actually $C2C0. Also note that leaving a register
selected will disable the "sleep" function... You must restore this port from
its mirror ($0F02) when you are finished reading the time. Also, to avoid
problems with background motor movement etc. You should use the high nibble
of the mirror value when you select a register.

4.1 Helpful modifications

4.1.1 Battery compartment:

The rear cover on the battery compartment is a bit of a pain to remove and re-install. I tapped the holes in the HERO chassis on the right side and installed short bolts that stick out about 1/8 inch on the inside. Then, I enlarged the holes of the battery plate on that side to just slip over the ends of the bolts. A couple of "banana" plugs nicely fit the remaining holes on the left to hold the cover in place.

4.1.2 Wrist (ARM):

The wrist on my Hero's ARM was quite loose. The set screw in it was worn or an odd size which none of my allen wrenches fit. I replaced it with a small bolt with a standard phillips head, and tightened it up. That made it better, but not perfect!

The metal band that slides over the wrist stepper motor is held in the plastic wrist cover by a simple slot. This (on my robot) was quite loose. I dropped hot glue into each side of the slot, and held the metal piece in place while it hardened. Then, after re-assembling the wrist, it was MUCH better than before.

NOTE: If you do these modifications, be sure to position the band so that the bolt stick out toward the claw. Otherwise it will hit the stop (normally hit by bars molded into the wrist cover).

4.1.3 Body Panels

One of the things that impressed me most on first playing with the Hero was the difficulty in getting the body panels onto the machine. You could just leave it open, however if you want Hero you have his clothes on when he plays so you could try this modification.

I removed the balls from one side panel, removed the points from the matching ends of the end panels. Then I screwed the ends onto the side panel, keeping the rubber grommets to give it some flexibility. then I put two of the smaller points into the top two holes of the same side panel. This gives me a 'C' shaped section of panels. Leave the points on the other side (I ground them down slightly to make them easier to insert).

To install the panels, place the remaining side panel in place, than carefully (watch the POT on the power board) slip the C shaped contraption around the robot. Sitting on the side away from the separate panel, lift up the body, and position the two metal points (now in the side panel) to stick into the top grommets on the robot (This gets easy with a bit of practice). Then, reach around (hug) the robot and shove the other ends into the separate side panel.

5.0 Hero-1 ROM Versions

5.1 Upgrade options

If you are upgrading ROMS, Note that according to the serial I/O docs, you should use the 1.3 ROM if you are upgrading from version 1.1, and the 1.U ROM if you are upgrading from version 1.0.

You can get the updated srecord files at http://hero.dsavage.com
Source files can be viewed at http://hero.dsavage.com

(Note: I have obtained a letter releasing the Hero ROMS into
not for profit public domain for Hero Robot Owners.

5.2 What is different between 1.3 and 1.U?

The 1.3 and 1.U ROM require a 4.00Mhz clock. The original HERO CPU is 3.59 Mhz, therefore the robot timing will be slow if you use the 1.3 or 1.U ROMS without also changing the crystal. The memory expansion card has a 4.00Mhz crystal on it, and works properly with these ROMS

Likewise, if you install the memory expansion card, and do not upgrade the system ROM, your robot timing will be too fast.

As for the contents of the proms, they are ALMOST exactly the same! There's only 3 bytes different:

```
F490: 1.3=79    U=3E     This is '3' and 'U'
E32C: 1.3=40,60 U=67,77
```

```
This appears to be in a data table used by a subroutine which is
part of the main drive motor control....
```

5.3 Demo ROM

```
Here is some demo ROM info:
  1: Tick-Tock clock
  2: alarm clock
  3: sing a portion of "Laura's Theme" with different voices
  4: Seem to carry a conversation based on sensing noise in room
  5: fall asleep when left quietly alone
  6: create Baroque-style music
  7: count hand claps
  8: sneeze and seem to have cold
  9: sing alphabet song
  a: demonstrate light sensor by announcing direction of brightest light in room
  b: "try to make you feel cooler on a hot day"...?

  To invoke:  press reset, press execute (9), 0 1 6.  This will copy
  the contents of the ROM into Basic's RAM and begin executing.
```

6.0 I need more RAM, what are my options?
6.1 Memory board for Hero-1
6.1.1 What are the jumper settings on the memory board?

```
  Memory Expansion Card/Map
  +---------------------------------------------+
  |   [:::::::::::::::]    [[[[[6808 CPU]]]](   | Sys ROM: E000-FFFF
  |  +----+ +----+ +----+ +----+ +----+ +----+  | Sys RAM: 0000-0FFF
  |  |A000| |8000| |6000| |4000| |2000| |1000|  |
  |  | to | | to | | to | | to | | to | | to |  | Monitor: 0000-003F
  |  |BFFF| |9FFF| |7FFF| |5FFF| |3FFF| |1FFF|  | Uses   : 0EFC-0FFF
  |  +----+ +----+ +----+ +----+ +----+ +----+  |
  +---------------------------------------------+
    Location  $   |  %   |  #   |  &   | <= Select socket
    U102    | J102 | J101 | J104 | J103 | U102 is only 4K in map
    U103    | J106 | J105 | J108 | J107 | Util ROM ET-18-4 must -> U103
    U104    | J110 | J109 | J112 | J111 | BASIC ROM must -> U107
    U105    | J114 | J113 | J116 | J115 |
    U106    | J118 | J117 | J120 | J119 |
    U107    | J122 | J121 | J124 | J123 |
    --------+------+------+------+------+
    6116*   |   B  |A or B|   B  |   B  | <= Select device
    6264    |   C  |A or B|   B  |   B  | %B powers from VCC
    2716*   |   A  |   B  |   B  |   B  | %A powers from standby for
    2532*   |   A  |   B  |   A  |   A  | RAM to retain during sleep
    2732*   |   C  |   B  |   A  |   B  | * = 24 pin device (must go
    2764    |   C  |   B  |   B  |   B  |     in pins 3-26 of socket)
    68764*  |   D  |   B  |   A  |   A  |
    ET-18-x*|   A  |   B  |   A  |   A  |
    --------+------+------+------+------+

  You must move any ROM installed in U417 on the CPU to the expansion
  card.
```

6.1.2 Memory board ribbon cable

```
  The cable is a straight through ribbon cable. You should be able to
  verify your cable by matching up pins on your 40 pin plug with the
  6808 socket (Use ohm meter with all the chips out). According to the
  schematic, all pins should go "straight through" except:
      1, 3, 7, 8, 21, 36, 38 and 39
```

6.1.3 Anything else I need to know?

```
  You also need to wire the standby (sleep) power on P102:
   - Tap green wire at S201-6 (power supply), and connect to P102-1
   - Connect P102-2 to ground
   - Connect P102-3 to +5V derived from a (new) 78M05 mounted on the
     chassis, which gets it power from S201-3 (red wire or red and blue
     wires).
```

6.1.4 But what about a CPU for it?

```
  The original memory board didn't come with a CPU either. You are supposed
  to move the one in the HERO. I don't know if you could find a 6808
  anywhere, but I used a 6802, which are the same CPU with internal RAM,
  which is disabled when it's installed in a HERO. The 6808 is a 6802 with
  missing RAM, and MUST have the RAM enable pin [36] tied LOW. I ran it
```

with a 6802 in it for a while, just to make sure they really
were compatible, and everything worked fine...

6.2 I don't have a memory card, any other options?

6.2.1 Upgrade hack to expand the Hero-1 to 8K

Looking at the schematic and remembering some of the hardware hacks I
did 10 to 15 years ago gave me an idea on how to double the user RAM
to 8K very easily. This is not pretty and I appreciate any criticism on
this. Essentially, it physically piggy backs 2 more chips on top of
the existing memory. I am taking advantage of the fact that memory
addresses are decoded by U403 in 8K blocks. This means the first 8K
of RAM is already accounted for. U406 selects 2K segments. Only 2 of
the 4 possible lines (for U406) are used. The other two (lines 13 and
12) are free for the taking.

The current address out of U406 are:
15 = $0000 - $07FF
14 = $0800 - $0FFF

the new ones will be:
13 = $1000 - $17FF
12 = $1800 - $1FFF

Before you do any of this, read and understand the schematics and
pages 79 - 80 of the Technical Manual.

I assume you have the schematics as you read this.

1) get 2 more 6116 memory chips
2) Bend pin 18 out of the way on the new chips (carefully)
3) pull the existing 6116s (U415, U416) from sockets
4) solder the 2 new chips onto the two existing ones pin for pin,
   EXCEPT for pin 18.
5) run a wire from pin 13 on U406 to pin 18 on the new chip that you
will insert into U416.
6) run a wire from pin 12 on U406 to pin 18 on the new chip that you
will insert into U415.
7) reinsert the memory chips.

Be sure your wires are long enough.
I recommend using 4 low power 6116s.

This will present an extra load on the data lines. Anybody know how
much? will it be within tolerance?

That's it, you now have 8K.

If you write a program that uses the new memory, be sure to jump
around $OEDF - $0FFF which are reserved for use by the Monitor.

6.2.2 Upgrade hack to expand the Hero-1 to 8K

This document describes how to upgrade a HERO 1's memory to 32K.

This solution won't work if you are using the optional ROM in socket
U417 and you have the address for this ROM set to 8 - 16K (J12 on).

The description below assumes you have a copy of the CPU Board
schematic.

I replaced both 6116s and the EPROM with a single 62256 (32K static
RAM) chip. I thought this was my best bet since I don't have the
EPROM and the MONITOR ROM occupies 56K - 64K. I figured having a
single chip occupying 0 - 32K would be a neat solution.

So I mapped the 6116 to the 62256 (see table below). The pins which
did not map I mapped the headers on the CPU board and to two
unused gates. Notice that the chips map quite nicely, only 2 pins
conflict.

I plugged a 28 pin socket into a 24 pin socket first bending pins 20,
23 and 26 out of the way. The sockets go pin 14 (28soc) to pin 12
(24soc). Pins 1,2,27 and 28 over hang. I then used 30 gauge solid
wire and connected pin 21 (24soc) to pin 27 (28soc) and pin 24
(24soc) to pin 28 (28soc). Using a 5 wire ribbon cable I connected
the remaining pins to the correct pins on header P408.
Specifically:

62256	P408
1	4
2	6
23	7
26	5

(The 5th wire gets used later)

Now for BAR CS (pin 20). I used A15 off the CPU (U401 pin 25),
VMA (U401 pin 5) and E (U401 pin 37).
These signals are found at the following locations:
A15 = P408, pin 3
VMA = U401, pin 5, plated hole on CPU board, stick a header pin in
E   = U423, pin 11, plated hole on CPU board, stick a header pin in

note that A15 and E are buffered, VMA is not.

All three are necessary to get the proper timing and select signal
(see the data sheet for the 6808 or 6802).

Now to properly combine these signals with unused gates on the CPU board.
What we want to select the chip is A15 low, E high, VMA high.

When A15 is low the upper 32K is not being used, E and VMA high tells
us the CPU is ready.

As it happens there is a free NAND gate (U423a) and a free OR gate (U414c).

So, runs wires from:
U423, pin 11   to U423, pin 1   plated hole
U401, pin 5   to U423, pin 2   plated hole

U423, pin 3   to U414 pin 10
P408, pin 3   to U414 pin 9

U414, pin 8   to BAR CS on the 62256 (pin 20) the last wire on my
                                        five pin hearer

That's it. You now have 32K of static ram instead of the original 4K.

Physically there are many ways to do the above. I decided to do this
all with headers, so the only changes to the CPU board are header
pins in 2 plated holes. If I want to get rid of this I don't have to
desolder anything.

I pulled U423 and U414 and put them into 14 pin sockets, bending out
the appropriate pins. All the wiring is done from these pins. You do
what works for you.

Function	6116 (2K)	62256 (32K)	CPU Board headers	Notes
A0	8	10	P406-10	
A1	7	9	-9	
A2	6	8	-8	
A3	5	7	-7	
A4	4	6	-6	
A5	3	5	-5	
A6	2	4	-4	
A7	1	3	-3	
A8	23	25	P408-10	
A9	22	24	-9	
A10	19	21	-8	
A11		23	-7*	Pin bent out
A12		2	-6*	Overhangs 24 Pin Socket
A13		26	-5*	Pin bent out
A14		1	-4*	Overhangs 24 Pin Socket
CS1	18	20	**	Pin bent out
GND	12	14		
I/O 0	9	11	P407-10	
I/O 1	10	12	-9	
I/O 2	11	13	-8	
I/O 3	13	15	-7	
I/O 4	14	16	-6	
I/O 5	15	17	-5	
I/O 6	16	18	-4	
I/O 7	17	19	-3	
OE	20	22	-2	
WE	21	27		Overhangs 24 Pin Socket attached to socket pin 21
Vcc	24	28		Overhangs 24 Pin Socket attached to socket pin 24

CS = BAR chip select $\overline{CS}$
GND = ground
OE = BAR output enable $\overline{OE}$
WE = BAR Write enable $\overline{WE}$
Vcc = Power (+5 V)
* new wire (not on original board)
** new wire to U414 pin 8

```
ROM Subroutines:              -7-   OUTBYT $F7AD - Write HEX byte
----------------             2   6  CLRDIS $F65B - Clear display
INCH   $F777 - Read keyboard  -1-   REDIS  $F64E - Reset Cursor
OUTCH  $F7C8 - Write char -> 3   5  OUTSTR $F7E5 - Wr PC chrs to !
OUTHEX $F7B5 - Write HEX nib  -4- 8 DISPLAY $F6F9 - Wr B bytes at X

ROBOT Interpreter:              5E      - Disable display
------------------              61 oo   - Speak, continue (,X)
02    - Abort Drive             62 oo   - Speak, wait (,X)
03    - Abort Steering          71 aaaa - Speak, continue     (ext)
04    - Abort ARM               72 aaaa - Speak, wait         (ext)
05    - Abort Speech            83      - Switch to NATIVE code
1C oo - Branch if Base busy     87 iiii - Sleep               (imm)
1D oo - Branch if Steering      8F iiii - Pause               (imm)
1E oo - Branch if ARM busy      BF aaaa - Jump when speaking
1F oo - Branch if Speaking      C3 ssii - Motor, wait abs     (imm)
21    - Zero (initialize)       CC ssii - Motor, continue abs (imm)
3A    - Return to executive     D3 ssii - Motor, wait rel     (imm)
41    - Enable light detect     DC ssii - Motor, continue rel (imm)
42    - Enable sound detect     E3 oo   - Motor, wait abs     (,X)
45    - Enable sonar            EC oo   - Motor, continue abs (,X)
4B    - Enable motion det.      F3 aaaa - Motor, wait abs     (ext)
4E    - Enable Display          FC aaaa - Motor, continue abs (ext)
51    - Disable Light det.      FD ...  - All motors ABS
52    - Disable sound det.            Extend, Sholder, Rotate,
55    - Disable motion det.           Pivot, Grip, Head, Steer
5B    - Disable sonar           All to end: FD 98 86 93 A5 75 C2 93

ss ii = mmmssDdd dddddddd           ss: 01=Slow  10=Medium  11=Fast
mmm:000=drive   001=extend  010=Shoulder  011=Wrist
    100=pivot   101=Grip    110=Head      111=Steering
D:0=Dn 1=Up(rel only) ddddddddd:Position(abs) or distance(rel)
```

7.0 Programmers information

7.1 OK, so what do I do with this Key Pad?

```
        Executive mode commands:    31 - Initialize
        ------------------------    32 - Home Arm
        1 - 6808 Monitor            33 - Write to TAPE (---FR, ----LA)
        4 - Manual Operation        34 - Load from TAPE
        6 -                         35 - Set TIME hhmmss AM=D PM=E 24=F
        7 - Learn (----Fr, ----LA)  36 - Set DATE yymmdd
        9 - User -> 0036            37 - Display TIME
        A - ROBOT Monitor           38 - Display DATE
        C - User -> 0033            39 - Write to RS232 (----Fr, ----LA)
        D -                         3A - Load from RS232
        F - User -> 0030            3C - ? Dload

        Monitor Commands:
        -----------------           8 - Single Step
        1 - ACCA                    9 - Set breakpoint (----br)
        2 - ACCB                    A - Autoload (address, data ...)
        3 - PC                      B - Backup to previous address
        4 - X                       C - Change displayed value
        5 - CC (Can't change)       D - Do (execute) at address
        6 - SP (Can't change)       E - Examine memory
        7 - Resume from BR or ST    F - Forward to next address
```

7.2 Memory Locations

```
                            0010 - Sonar hits
                            0011 - Sonar range
0000 - Extend   00-98       0027 - Motion detect IRQ vector
0001 - Shoulder 00-86       002A - Trigger IRQ vector
0002 - Rotate   00-93       002D - Exp. board IRQ vector
0003 - Pivot    00-A5       0030 - User KEY1 vector
0004 - Gripper  00-75       0033 - User KEY2 vector
0005 - Head     00-C2       0036 - User KEY3 vector
0006 - Steering 00-93       0EFC - Counts 0-FF @ 1024 Hz
```

7.3 Input Ports

```
      C220=Sonar C240=Sense C003=Keyboard C2A0=Exp.Bd

C200 (Int)  C260        C280 (Remt)     C300
7-EXP.BD    7-Tape in   7-Arm/Base      7- \
6-Wheel     6-St. CW    6-\ Speed       6- | Not Implemented
5-Trigger   5-Arm UP    5- >Motor       5- | in hardware
4-Time Clk  4-Hd. CCW   4-/ Dirctn      4- /
3-LB low    3-St. CCW   3-Left          3-\
2-MB low    2-Arm DN    2-Right         2- >Clock
1-Motion    1-Hd. CW    1-Sleep/Norm    1- >Data
0-Sonar     0-Spk Req   0-Trigger       0-/
```

7.4 Output Ports

```
       C200=ClrInt(1Bit/each) C220=Exp.Bd C110=Display

  C300 (clk)  C260        C280        C2C0
  7-Disab     7-\         7-\         7-ArmSel Rot,Sh,Hd
  6-Write     6->Steer    6->Extnd    6-ArmSel Pvt,Gr,Ex
  5-Read      5->ing.     5->Head     5-Speech Strobe
  4-Hold      4-/         4-/         4-Light/Sound
  3-\         3-\         3-\         3-\
  2->Clk      2->Pivot    2->Grip     2- >Time
  1->Data     1->Rotate   1->Shldr    1- >Clock
  0-/         0-/         0-/         0-/

  C2A0 Drive  C2E0 Power  C240 (Spk)
  7-Fwd/Rev   7-Eye/-Ear  7-\\Pitch
  6-On/Off    6-Main Pwr  6-|/Infl.
  5-\         5-Sense  P  5-|      \
  4-|Main     4-DisplayP  4-|Phonen<
  3-|Drive    3-Speech P  3-|Select<
  2->Speed    2-Motion P  2->Speech|
  1-|         1-Sonar  P  1->Data  |
  0-/         0-Tape out  0-/      /
```

7.5 Reading the Clock

```
  C300<-A0 C2C0<-Reg# C300 has value
  Reg#: 0=SeL 1=SeH 2=MiL 3=MiH 4=HrL 5=HrH (B3=24hr B2=pm)
        6=Wk  7=DaL 8=DaH (D2=leapfe) 9=MoL A=MoH B=YrL C=YrH
```

7.6 What about the Hero sound chips?

```
       Hero-1:         Votrax SCO1
       Hero-Jr:        Votrax SCO1
       Hero-2000:
```

8.0 What development tools are available for the Hero Robots?
8.1 What development tools are available for the Hero?

```
At the time of this writing, DDS offers CFLEA; a PC
       hosted development environment for the HERO-1 using
       either/both 'C' and Assembler.
```

8.1.1 Where do I get it?

```
       Dunfield Development Systems
       http://www.dunfield.com info@dunfield.com
```

8.2 What development tools are available for the Hero-Jr?

```
       I know of none available. If you hear of one please let me know.
```

8.3 What development tools are available for the Hero-2000?

```
       I know of none available. If you hear of one please let me know.
```

The Hero-2000 has an interlock on the sonar head. If you will short pins 2 & 3 on P11B (the connector sticking out of the top of the head) you can bypass the interlock. Note pin 1 is the black wire, two is the green and 3 is the red.
