Reliability:
- Flip-up foot design, no tons, no leases, for minimal joint number
- Faulhaber precision DC motors and gearbox
- High flux 5M ribbons for the same and legs
- For power connection, fine-strand remote cable
- Optical sensing of foot contact; medical implant sensor cable
- D-shaped shaft pulley connections, for positive grip
- Optical encoder position and velocity measurements
- Over-travel limit switches for feet
- Assembled with high-adhesion Metregrip 303 epoxy

Autonomous operation, with minimal power consumption and long battery life:
- Efficient Faulhaber permanent magnet DC motors
- ST Micro VNH2 integrated motor drivers, 0.019 ohms on
- Processing, memory, and I/O are largely on a single Freescale 56F8347 chip
- Switching voltage regulator for electronics: 2.6 volt total from battery
- 186 watt-hour max. (80 usable) 11.1 volt lithium-ion batteries
- Low power Radiotronix WI-232DTS radio transmitter module

Minimal mass:
- Carbon-fiber leg tubes
- Sheet aluminum boxes, the same carbon composite, to reduce weight
- CNC-machined hip tubes to minimize overhang mass on outer legs
- Lightweight cable drive for the feet
- 7075 aircraft aluminum for the feet
- Lightweight, high-energy lithium-ion batteries

Steering (at least a little):
A “4-leg” walk is not easily turned, but for long-distance walks we needed it to be able to make gentle turns (e.g., at the end of a running track). A single-cable system is used to drive the feet downwards; a pair of long, near-constant-force springs pull the foot upwards when the foot is moved through the center. This means that, for the outer feet, if the left foot is pulled a little more or less than the right, the foot will move correspondingly, and the right-left symmetry of the robot will be broken. If, as a result, the right foot slips more than the left (or vice versa), the robot will turn.

Steering motor motion is driven by an eccentrically mounted on a gear, powered by a radio-control servo motor and control unit.

Walk Controller Operation:
Three parallel state machines are used, one for the hip control, one for the swing feet, and one for the stance feet. (Not shown – an experimental additional state to swing the leg back to catch the robot if it begins to tilt backward.)

Stance Feet
- Post-heelstrike: The stance feet are held at the same preset angle as when the stance leg angular rate, sampled at midstance (fully upright), rises to a certain level.
- Transition to the next stance occurs when the stance leg angular rate, sampled just before heelstrike, exceeds a given parameter value.

Swing Feet
- Pushoff: A predetermined torque is applied to the swing foot until it reaches the commanded angle. When the applied torque drops to a negligible level (as a result of successful pushoff), the foot state machine goes to the next state.
- Pushoff: The stance feet go to a new angle, in preparation for pushoff. Transition to the next state occurs at heelstrike.

Hip
- Hold up: The feet are flipped up out of the way, and held via position control. Transition to the next state occurs when the leg is within a fixed angle of its target position, most of the way through the swing.
- Swing: A preset torque is applied to swing the leg forward. Transition to the next state occurs when a preset angular velocity is achieved.
- Brake: A reverse torque is applied, based on leg angular velocity at the start of the state. Transition occurs when the leg stops, or reaches the desired swing angle.
- Hold down - swing: The foot is flipped back down to a preset angle, ready for heel strike, the next transition condition.

Heelstrike (swing leg contacts ground)
All three state machines undergo a state transition at this point, marking the start of another step. It is at this point that the former stance leg and feet become the new swing leg and feet, and vice versa (often confusingly).

Embedded computer controls the robot configuration, and the microcontroller is the brain of the robot. The microcontroller is at the center of all the hardware connections, and it reads and writes to the robot’s memory and parameters.