September 2002

Mobile robot for uneven terrain

Kenneth Chin
University of Pennsylvania

Robert Breslawski
University of Pennsylvania

Terry Kientz
University of Pennsylvania

R. Vijay Kumar
University of Pennsylvania, kumar@grasp.upenn.edu

Postprint version. Published in *Proceedings of the 2002 ASME Design Engineering Technical Conference (DETC 2002)*, 27th Biennial Mechanisms and Robotics Conference, Volume 5B, pages 1327-1333.

Mobile robot for uneven terrain

Abstract
This paper outlines the details of the development of a mobile robot than can navigate uneven terrain. The robot incorporates a combination of wheels and legs. The legs are based on a parallel-drive 2-R linkage that allows the motors to be located on the robot frame. The legs are driven through servo motors while the wheels are powered through DC motors. A PIC microcontroller is used to control the system, while a novel IR-based communication module allows the user to remotely control the device. In the proof-of-concept prototype, a human operator can control the approximately 6x9x4 inches (15.24x22.86x10.16 cm) and approximately 8 lb. (3.63 kgs) robot (with onboard electronics and control systems) to climb and descend steps. Future versions can be expected to be autonomous and equipped with cameras and ad hoc networking cards for field operations.

Comments
Postprint version. Published in *Proceedings of the 2002 ASME Design Engineering Technical Conference (DETC 2002)*, 27th Biennial Mechanisms and Robotics Conference, Volume 5B, pages 1327-1333.

Proceedings of DETC '02
2002 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference
Montreal, Canada, September 29–October 2, 2002

DETC2002/MECH-34361

MOBILE ROBOT FOR UNEVEN TERRAIN

Kenneth Chin
University of Pennsylvania
GRASP Laboratory, MEAM
3401 Walnut Street
Philadelphia, PA 19104.
E-mail: kwchin@seas.upenn.edu

Robert Breslawski
University of Pennsylvania
GRASP Laboratory, MEAM
3401 Walnut Street
Philadelphia, PA 19104.
E-mail: robert.s.breslawski@kodak.com

Terry Kientz
University of Pennsylvania
297 Towne Building, MEAM
220 South 33rd Street
Philadelphia, PA 19104.
E-mail: tkientz@seas.upenn.edu

Vijay Kumar
University of Pennsylvania
GRASP Laboratory, MEAM
Room 301C, 3401 Walnut Street
Philadelphia, PA 19104.
E-mail: kumar@cis.upenn.edu

ABSTRACT
This paper outlines the details of the development of a mobile robot than can navigate uneven terrain. The robot incorporates a combination of wheels and legs. The legs are based on a parallel-drive 2-R linkage that allows the motors to be located on the robot frame. The legs are driven through servo motors while the wheels are powered through DC motors. A PIC microcontroller is used to control the system, while a novel IR-based communication module allows the user to remotely control the device. In the proof-of-concept prototype, a human operator can control the approximately 6×9×4 inches (15.24×22.86×10.16 cm) and approximately 8 lb. (3.63 kgs) robot (with onboard electronics and control systems) to climb and descend steps. Future versions can be expected to autonomous and equipped with cameras and ad hoc networking cards for field operations.

INTRODUCTION
There are often potential risks in exploring unknown environments in which it is useful to dispatch robots as scouts. The horrific Oklahoma City bombing, the recent tragic fire in Worcester, and the attacks on the World Trade Center are all unforgettable events where humans were often ill-equipped and incapable of searching for victims. Having a small compact robot with sensors and the capability of navigating uneven terrain is attractive. The challenge is to push the boundaries of what is available now and create smaller, better sensing robots with greater mobility.

Most land based mechanized locomotion systems are based on the principle of the wheel for two principal reasons. First, in contrast to such actively coordinated vehicles as walking machines, the load is supported passively. The actuation is used only for propelling the vehicle forward and this results in a more reliable system. Second, rolling contacts between the wheel and ground allow for efficient locomotion on flat, prepared surfaces. However, the performance of wheeled systems is adversely affected by uneven terrain. In contrast, legged locomotion systems have the ability to pick footholds and to actively control the distribution of forces, and are therefore potentially more agile and versatile (Kumar and Waldron 1989b, Song and Waldron 1989). In addition, the actively controlled legs give the vehicle an active suspension that can be controlled to provide a desired ride.

The versatility of legged vehicles comes at the price of increased cost/complexity, and poor reliability. Legged vehicles are inefficient because they must perform isometric work in order to just support the vehicle. The actuators have to support the weight of the vehicle, in addition to providing the tractive force, which translates to low overall payload/weight ratios for each leg. If non-backdriveable transmissions are used to reduce the actuator forces and the isometric work, it is difficult to control the foot forces, a feature that is essential for adapting to different terrain and for an active suspension. The reliability and stability of legged vehicles can be improved by increasing the number of legs. However, this is at the expense of increased complexity in design and control, cost, and size. All this makes it very difficult to design a compact and reliable legged vehicle with a high payload to weight ratio.

Motivated by the above observations, we consider a hybrid vehicle that can use both legs and wheels for operation on uneven terrain (Krovi, 1995). However, unlike in a legged vehicle, the legs are not required to support the entire weight of the vehicle. Further, the vehicle can use its powered wheels to navigate on prepared surfaces without deploying its legs. Because the vehicle can be passively supported by the wheels in a statically stable configuration, it is safer than a legged vehicle. Finally, when not being used for locomotion, the legs can also be used as manipulators to interact with the environment. Thus, the hybrid system (Kumar et al., 1996) can perform many tasks that can be accomplished by the traditional legged vehicle, and yet is simpler, safer and less expensive.

Another important consideration in applications such as search and rescue is control. Often autonomy is too difficult a goal. In applications such as bomb-sniffing, exploration, toxic waste elimination, and search and rescue, it is useful to have a human being able to control the robot at a distance. Automated and manual controls are both needed in order to dictate actions efficiently. Implementing such a control system is complicated, although remote control units such as ones commonly used with radio-controlled cars and televisions offer us examples of simple implementations for discrete control. However, currently there are no such implementations for robotic systems.

In this paper, we first describe the mechanical design of the system. The design specifications, kinematic models, and preliminary design are described in the next section. We next describe the basic control system, along with the remote controller for the system. We finally describe the experimental prototype with results from the experiments.

PRELIMINARY DESIGN

The performance objectives and the design criteria for the intended function and optimization of the vehicle included: 1) restricting the wheelbase length of the robot to approximately 10 inches (25.4 cm), 2) having the mobility and ability to traverse flat ground and travel over objects similar to size with the use of two sets of arms at its front and back ends, four arms altogether, 3) low cost – hence, utilizing servo motors to drive the arms and DC motors to drive wheels, and 4) being as light as possible with an upper weight limit of no more than 10 lbs. (4.54 kgs). It is important to note that the mobility of the robot was also limited by the availability of motors, their power ratings, the type of control desired, and the desired weight.

A kinematic model of the vehicle climbing a planar step is shown below in Fig. 1. The center of mass, the wheel-ground contact, the axle, the shoulder/hip joint, the elbow/knee joint, and the foot-ground contact are all lower pairs forming a closed kinematic chain.

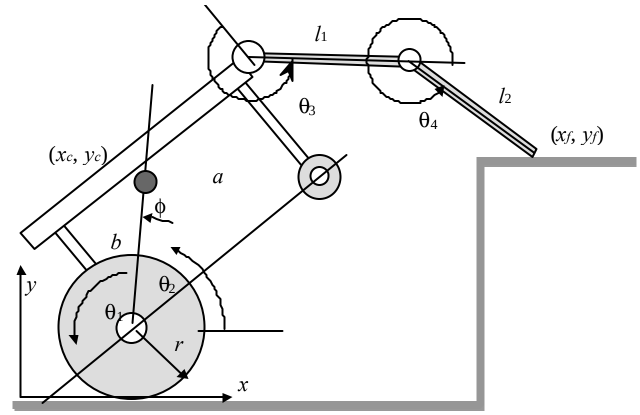


Figure 1. Arm and Wheels Torques as a Function of Angular Positions

$$x_{cg} = R\theta_1 + b\cos(\theta_2 + \phi) \quad (1)$$

$$y_{cg} = R + b\sin(\theta_2 + \phi) \quad (2)$$

$$x_f = R\theta_1 + a\cos(\theta_2) + l_1\cos(\theta_2 + \theta_3) + l_2\cos(\theta_2 + \theta_3 + \theta_4) \quad (3)$$

$$y_f = R\theta_1 + a\sin(\theta_2) + l_1\sin(\theta_2 + \theta_3) + l_2\sin(\theta_2 + \theta_3 + \theta_4) \quad (4)$$

Five pairs in a planar closed chain suggest that the three powered joints, θ_1, θ_3, and θ_4 are constrained. Only two of the four variables, θ_1, θ_2, θ_3, and θ_4 are independent and Eq. (3) and Eq. (4) can be used to solve for the two dependent variables based on knowledge of any two of the three variables. Since we have sensors measuring θ_3 and θ_4, we can always determine the other two in real-time.

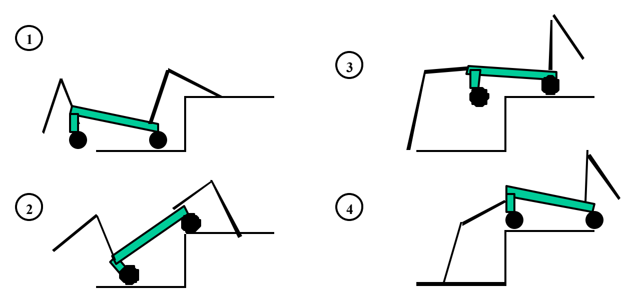


Figure 2. Ascent Sequence

Differentiating Eqs. (1) – (4) provide equations relating velocities. Using the principle of virtual work, one can obtain equations for the torques. Following previous work, we can determine the torques for maneuvers such as the ones shown in Fig. 2 and Fig. 3. A complete model for the rear links will include a rear leg with two joints, θ_5 and θ_6. Further details outlining the detailed analysis of torques and optimization are addressed in prior work (Wellman, et al., 1995).

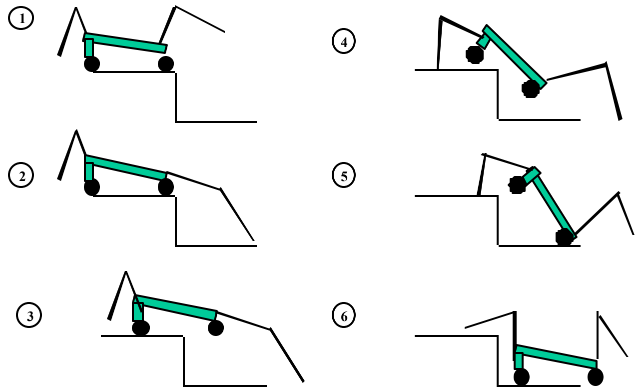


Figure 3. Descent Sequence

By using Working Model 3D software, a virtual prototype was constructed and tested repeatedly until a reasonable torque rating of about 280 ounce-inches (2.016 N-m) was reached (hence, the overall weight without electronics and controls could not exceed 5 lbs (2.27 kgs)). A parallel 2-R linkage is used to control each leg. In the planar model shown in Figures 1-3, a single motor controls both proximal joints on the front leg, and another motor controls the distal joints on the front leg through an idler pulley on the proximal joint using a cable drive. Similarly, two motors are used to control the rear leg.

CONTROL SYSTEM ARCHITECTURE

The control system architecture is segmented into three major modules - IR Communication, Servo Motor, and DC Motor Modules – that are dependent upon one another through the master microcontroller – PIC16F873 (a flash-based reprogrammable chip that will be referred to as the PIC). The master controller controls and processes the incoming and outgoing information for each of the sub-modules.

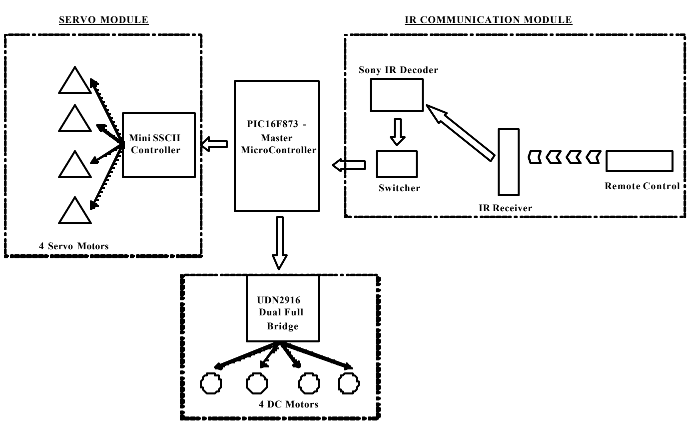


Figure 4. Control System Architecture

1. IR Communication Module decodes and transmits the incoming IR signal from the universal remote controller to the master microcontroller at a maximum of 2400 baud.

2. Servo Module handles the processing for the four individual servo motors that dictate the climbing motions of the robot where the maximum transmission rate is 9600 baud.

3. DC Motor Module incorporates a dual full-bridge PWM motor driver chip - the UDN2916B - receiving direction and speed data that generates PWM duty cycle and signal that bidirectionally controls each set of motors.

Motor Logic

Servo Motors. To control the servo motors, commands were sent from the PIC master controller to the Mini SSCII servo controller. Three packets of data were sent to dictate the position and motion control of the servos: sync, servo number, and servo position. Sync serves as an internal marker for the purposes specific to the operation of the Mini SSCII controller. The servo number dictates to which specific servo the transmitted data is intended for. Servo position contains the information for the servo's new position.

DC Motors. PWM current control is utilized for the DC motor speed control. Two logic level inputs select output current units of 0%, 33%, 67%, or 100% of maximum. A single logic level input (phase) allows load current direction (i.e. motor rotational direction). The table below summarizes the output current when we make the I0 and I1 on the chip logic high or low.

Table 1. DC Motor Control Logic

I_0	I_1	Output Current
L	L	V_{ref} / 10 R_s = I_{trip}
H	L	V_{ref} / 15 R_s =2/3 I_{trip}
L	H	V_{ref} / 30 R_s = 1/3 I_{trip}
H	H	0

Infrared Control

Behind the infrared communication, the PNA4601M – infrared detector – serves to detect the incoming signal from the universal remote. The PNA4601M converts the infrared signal to a logic level signal that is then received and decoded by the FT936 – Sony IR Decoder chip. Since Sony uses the standard SIRC protocol – where a pulse form is sent builds up a 12-bit serial interface where the first 5-bits contain the device code while the remaining 7-bits contain the button code – the buttons are decipherable and can be pre-programmed to perform a certain action.

Interrupts and Interrupt Handler Routine

The main program consists of polling for input and homing the robot's position, climbing stairs, or climbing down stairs. A sub-program continuously polls for input in the background from the user and interrupts the main program, then executing the interrupt-handler routine. One can either implement "external interrupting" or "interrupt-on-change." External interrupting allows a main program to run a program in the background in a continuous loop and immediately branches to the interrupt handler when interrupted. This is the desired action.

The logic behind interrupts for the PIC16F873 is outlined clearly. For "external interrupting," one must set the INTCON register by simply setting INTCON = %10010000 enabling external interrupts on PortB.0. For "interrupt-on-change," setting INTCON = %10001000 enables this type of interrupt on PortB.4.

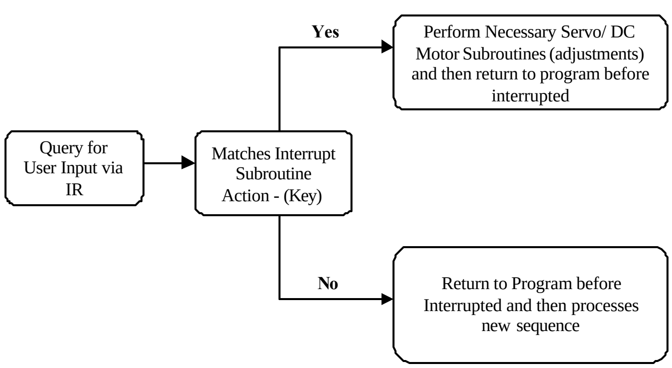
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF

Figure 5. Intcon Register (PIC16F873)

By utilizing "external interrupting," the (GIE) Global Interrupt Enable Bit and the (INTE) RB0/INT External Interrupt Enable Bit are set to 1. When the program experiences an interrupt, (INTF) RB0/INT External Interrupt Flag Bit is set to 1 and the main program halts and jumps to the defined interrupt handler routine. Upon completing the interrupt handler routine, the INTF bit is cleared to 0 to reenable interrupt service resetting the INTCON register.

Within the constraints of the defined system, "external interrupting" is more appropriate. By pushing a button from the universal remote, the main program would halt even if the robot was in the midst of climbing stairs and then result in entering the interrupt handler that then queries for "serial-in" data and sends new position and motion control information to the robot. A sample of the interrupt handler routine written in PicBasic Pro is depicted below.

It is important to understand the logic flow of how the interrupt handler integrates into the context of the main program. One must understand that the Main Program loop consists of completing the necessary defined subroutine sequences for the servos and DC motors for either: a) homing the robot, b) climbing stairs, or c) climbing down stairs. The user can interrupt any of these three sequences at any point in time that then immediately enters into the interrupt handler routine. The logic is outlined in the flowchart below.



Interrupt Handler Flowchart

Figure 6. Interrupt Handler Logic

PROGRAM LOGIC

The programming logic can be depicted in three distinct segments: Startup, Main Program, and Interrupt Handler.

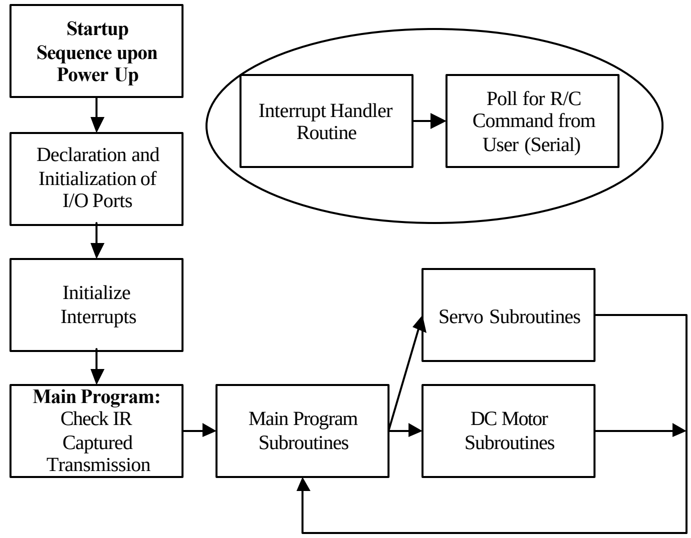


Figure 7. Program Logic

Necessary initializations are performed in "Startup" that also include declaration and functionality of the interrupt handler. The Main Program consists of checking the state of the captured data from the IR. Within the Interrupt Handler, the program jumps to specified sequences and sends necessary instructions to the servo and DC motors.

The subroutines that are embedded within the interrupt handler routine that cannot be interrupted include minor adjustments to the motions of the program.

Table 2. Universal Remote Key Directory

Button	Action	Description
0	Forward	Move the vehicle forward with DC motors
1	UP	Sequence to climb up step for robot
2	DOWN	Sequence to climb down step for robot
3	N/A	Future Expansion
4	N/A	Future Expansion
5	Back Distal Up	Adjusts the back distal link to swing slightly upward
6	Back Distal Down	Adjusts the back distal link to swing slightly downward
7	Backwards	Move the vehicle backwards with DC motors
8	Back Near Out	Adjusts the back near link to swing slightly outward
9	Back near Close	Adjusts the back near link to swing slightly inward
Enter	N/A	Future Expansion
Channel Up	Front Distal Up	Adjusts the front distal link to swing slightly upward
Channel Down	Front Distal Down	Adjusts the front distal link to swing slightly downward
Volume Up	Front Near Out	Adjusts the front near link to swing slightly outward
Volume Down	Front Near Close	Adjusts the front near link to swing slightly inward
Mute	HOME	Return all links to Home Position
Power	N/A	Future Expansion

For example, there are about 8 to 10 subroutines that are included to allow the user to make adjustments to various components of the robot in small increments. They include moving various links up and down while some control the motion of the DC motors (i.e. move the front distal link out further, turn on the motors a little more to get the robot closer to the step, lower the backnear arms so they touch the step more firmly, etc.). The various codes are preprogrammed onto the universal remote. Lastly, there are three automated sequences included: homing the various components of the robot, climbing the stairs, and climbing down the stairs.

EXPERIMENTAL PROTOTYPE

The prototype hybrid locomotion system shown in Figure 8 is capable of tackling a variety of terrain conditions. It is currently programmed to climb 2 inches (5.08 cm) high obstacles (like curbs), ascend (or descend) 30-degree inclines, and navigate omni-directionally on planar surfaces using wheels and on "difficult" terrain with both wheels and legs.

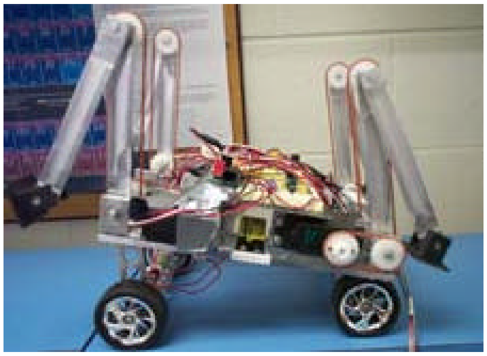


Figure 8. Prototype

Mechanical Design

The chassis is machined out of aluminum. Each arm is controlled by a single 2R Linkage. The mechanism couples the proximal links to one servomotor though cable chain, sprockets, and a rod while the distal links of that same end are coupled to another servomotor through cable chain, sprockets, and an idler. Furthermore, the cable chain is preloaded to remove backlash.

The prototype is symmetric front to back, except for the fact that the front hip joints (see Figure 8) are lower than the rear hip joints. This provides an asymmetry that can be used to advantage. The rear legs can be used for climbing while the front legs can be used to push the chassis up. (The opposite is true when descending).

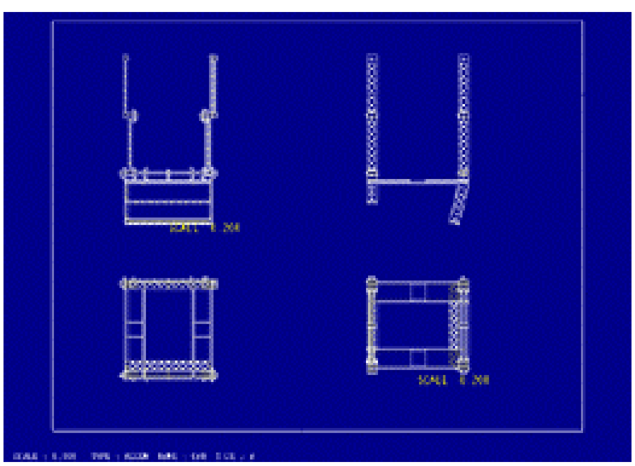


Figure 9. Parallel 2-R Linkage Drive Mechanism

The end effectors are footpads for the robot, and are made out of ABS using an FDM machine and machined aluminum. A passive revolute joint serves as an ankle, with torsion springs keeping the end effector in a home position. The compliant ankle allows the end effector to conform to variations in the terrain.

For traversing flat ground, wheels were chosen (as opposed to tank treads) as to have clearance under the robot during climbing. A very small DC motor with a gearbox, having a ratio of 262:1, was used for each of the four off-the-shelf wheels that have a radius of 2.5 inches (15.875 cm). The effective torque of each wheel is 89 oz-in (.623 N-m). The wheels were positioned relative to the chassis to raise the back end up while on level ground with an angle of inclination of 5 degrees to better maintain the center of gravity while climbing.

Table 3. Prototype Dimensions

Body Length: 10.5 inches (26.67 cm)
Wheelbase: 9 inches (22.86 cm)
Wheel Radius: 2.5 inches (15.875 cm)
Body Width: 7.5 inches (19.05cm)
Distal Links: 6 inches (15.24 cm)
Wheel Motors: effective torque of 89 oz-in (.623 N-m)
Arm Motors: 224 oz-in. (16.0 kg-cm) at 4.8 V

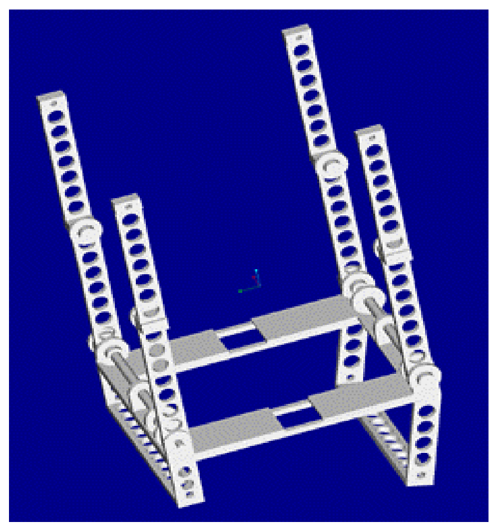


Figure 10. Body Platform and Base with Arms

The body of the prototype is completely manufactured from aluminum while the entire vehicle with electronic systems weighs approximately 8 lbs (3.63 kgs). It is important to note that both the near and distal links composing each of the arms were manufactured from ¼ inch (.635 cm) (thick aluminum to minimize bending moments and flexing. The length of the near and distal links are both 6 inches (15.24 cm). The maximum extension of a single arm when both the near and distal links are fully extended is 12 inches (30.48 cm).

Controls

A PIC16F873 that serves as the master microcontroller of the entire system that communicates with the various other subcomponents. A Mini SSC Controller directs the actions of the servo motors while the UDN2916B dual full-bridge PWM motor driver controls the four DC motors attached to each wheel. A universal remote control serves as the human interface sending infrared commands to the vehicle that is detected by the PNA4601M where the signal is then decoded by the FT936 decoder. The control systems is depicted below in further detail.

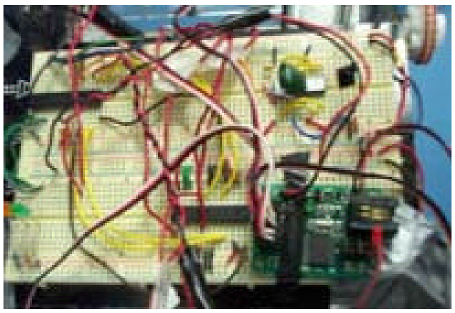


Figure 11. Control Systems

Implementation

After completely the modeling and simulations, the body was initially constructed followed by the implementation of the parallel 2-linkage drive mechanism along each of the four arms. Upon installation of the servo and DC motors, the servo and DC motor sub-systems were integrated with their respective drivers. Lastly, the infrared communication sub-system was tested and integrated into the existing control systems.

It is important to note that during implementation, a few problems arose. Due to limitation and variation in serial transmisson rates, assurance of similar configurations was required. Secondly, limitations in the PIC16F873 microcontroller's programming capacity prevented the implementation of having CALLS and GOSUBs more than 4 levels deep on the stacks or else a system crash would result.

EXPERIMENTS

The robot program contains three pre-programmed sequences (homing all components, climbing a step, and climbing down a step) that executes upon a single push of the button on the universal remote (respectively <<Mute>>, <<1>>, and <<2>>). The sequences are designed to accommodate a test step size of 2 inches (5.08 cm). However, the sequences can be adjusted to accommodate to different stair heights up to about 5 inches (12.7 cm) due to the parallel 2-R linkage system that accommodates varying heights.

Demonstrating the capabilities of the vehicle is best captured by a simulated sequence as shown earlier in Fig. 2 and Fig. 3. Upon starting up the position, we first "HOME" the position of all the links of the robot by pushing the <<Mute>> on the universal remote. By then pressing <<1>>, the user executes the climbing sequence which is depicted in Fig. 2. Upon reaching the top of the step, the user may wish to climb another step (hence, pressing <<1>> again) or traverse forward incrementally by pressing <<0>>. Upon deciding to climb down a step, the user presses <<2>> to begin the fully-automated sequence depicted in Fig. 3. It is important to that that at any time during these two climbing sequences, the user is able to pause the program and input new commands to make real-time incremental adjustments then followed by completing the execution of the previously selected automated sequence (incremental adjustment commands are listed in Table 2).

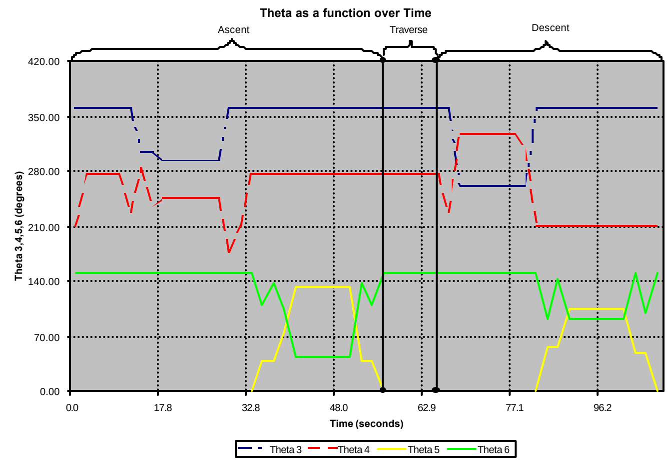


Figure 12. Angular position as a Function over Time

Figure 12 above outlines the results from the commands that are logged by the control systems and sent by the master microcontroller. It is important to notice that the angles correspond to those outlined in Figure 1. Hence, θ_3 and θ_4 represent the front proximal and the front distal set of arms respectively. θ_5 and θ_6 correspond to the back set of arms (Note that θ_5 and θ_6 are offset by –360 degrees in the figure for discussion and graphical purposes). Lastly, the ascent, traverse, and descent sequences have respective durations of 56.1, 7.5, and 40.6 seconds respectively.

Figure 13. Wheel Angular Velocity

Figure 13 above outlines the logged results for the wheel rotations that corresponds to θ_1 as shown in Figure 1. It is important to note that both the front and back set of wheels are all sent the same commands and can be correspondingly represented by a single plot.

CONCLUSION

We described a small, 6×9×4 inches, hybrid vehicle that weights approximately 5 lbs, and can scale up to 4-inch high steps. The vehicle consists of four wheels and four arms (legs) that are coordinated to traverse uneven terrain.

There are two important contributions to the paper. This is the first time a four-legged, four-wheeled vehicle has been demonstrated. The use of cable-drive parallel linkages and the use of symmetry minimize the number of actuators and the weight of the vehicle. The second contribution is a novel control system that incorporates infrared communication and control, with interrupt capabilities via the implementation of an interrupt handler routine. A user can use a remote control to have the robot simply climb up or down stairs with the single push of a button. A push of a button also interrupts the program at any time allowing real-time adjustments. The net result is an embedded learning process in the design of the programming architecture.

Additional improvements can be made in future models. Development of a closed-loop system that incorporates various proximity and infrared systems onboard can further enhance the control systems. This is a direction for future development. Hence, the user will not only have control over the motions of the robot, but sensors will also provide additional information to better govern and automate control. Additionally, future directions include automated transfer function generation in order to incorporate more precise control of the various motors to match the various needs as presented by the environment by utilizing XPC – a Matlab/ Simulink interface tool.

ACKNOWLEDGMENTS

We gratefully acknowledge advice and direction from Professors Venkat Krovi at McGill University and Tom Sugar at Arizona State University. We also thank Wally Szczesniak and Bob Miller at the University of Pennsylvania who helped with the manufacturing of the prototype.

REFERENCES

[1] Hiller, M., and Kecskemethy, A., 1987, "A Computer-oriented Approach for the Automatic Generation and Solution of the Equations of Motion for Complex Mechanisms," Proc. 7th World Congress on the Theory of Machines and Mechanisms, Sevilla.

[2] Hirose, S., 1984, "A Study of Design and Control of a Quadruped Walking," International Journal of Robotics Research, 3(2), pp. 113-133.

[3] Hirose, S., and Morishima, A, 1990, "Design and Control of a Mobile Robot with an Articulated Body," International Journal of Robotics Research, 9(2), pp. 99-114.

[4] Krovi, V., Wellman, P., and Kumar, V., 1994, "Design of a Walking Wheelchair for the Motor Disabled," Proc. 4th International Conference on Rehabilitation Robotics, pp. 125-131, Wilmington, Delaware.

[5] Krovi, V., 1995, "Modeling and Control of a Hybrid Locomotion System," Masters Thesis. Department of Mechanical Engineering, University of Pennsylvania, URL: http://www.cis.upenn.edu/~venkat/Publications.html.

[6] Kumar, V. and Waldron, K.J., 1988, "Force Distribution in Closed Kinematic Chains," IEEE Transactions on Robotics and Automation, 4(6), pp. 657-664.

[7] Kumar, V. and Waldron, K. J., 1989, "Actively Coordinated Vehicle Systems," ASME Journal of Mechansims, Transmissions, and Automation in Design, 111(2), pp. 223-231.

[8] Kumar, V. and Waldron, K. J., 1990, "Force Distribution in Walking Vehicles," ASME Journal of Mechanical Design, 112(1), pp. 90-99.

[9] Kumar, V., Wellman, P., and Krovi, V., 1996, "Adaptive mobility system," U. S. Patent 5,513,716.

[10] PIC16F87X Data Sheet, 2001, Microchip Technology Inc., 2355 West Chandler Blvd., Chandler, AZ

[11] PicBasic Pro Compiler, 2001, microEngineering Labs, Inc., Box 60039, Colorado Springs, CO

[12] Raibert, M. H., 1985, "Legged Robots that Balance," MIT Press., Cambridge, Massachusetts.

[13] Salisbury, J. K. and Roth, B., 1983, "Kinematic and Force Analysis of Articulated Mechanical Hands," ASME J. Mechanisms, Transmissions and Automation in Design, 105(1), pp. 35-41.

[14] Song, S. M., and Waldron, K. J., 1989, "Machines that Walk", MIT Press, Cambridge MA.

[15] Sreenivasan, S. V., and Waldron, K. J., 1996, "Displacement Analysis of an Actively Articulated Wheeled Vehicle Configuration With Extensions to Motion Planning on Uneven Terrain," ASME Joumal of Mechanical Design, 118(2), pp.312-320.

[16] Thring, M. W., 1983, "Robots and Telechairs: Manipulators with Memory, Remote Manipulators, Machine Limbs for the Handicapped", Ellis Horwood; New York: Halsted Press.

[17] Waldron, K. J., Vohnout, V. J., Pery, A., and McGhee, R. B., 1984, "Configuration Design of the Adaptive Suspension Vehicle," International Journal of Robotics Research,. 3(2), pp. 37-48.

[18] Wellman, P., Krovi, V., Kumar, V., and Harwin, W., 1995, "Design of a Wheelchair with Legs for People with Motor Disabilities," IEEE Transactions on Rehabilitation Engineering, 3(4), pp. 343 - 353.