ASME Early Career Technical Journal, Vol. 5, No. 1, pp. 8.1-8.8, October 2006.

CERBERUS: DEVELOPMENT OF A HUMANOID ROBOT

Mehmet Ismet Can Dede, Salim Nasser, Shusheng Ye, and Sabri Tosunoglu
Florida International University
Department of Mechanical Engineering
10555 West Flagler Street
Miami, Florida 33174

ABSTRACT

The motive behind building humanoids is to design a robot that can duplicate the complexities of human motion, decision making, be able to help people and even accomplish tasks that cannot be carried out by humans. Building humanoids has always attracted scientists throughout the world but although the aim is seemingly simple, the task is never easy. This paper briefly describes the final design and gait definitions of the humanoid robot named Cerberus. Also, the manufacturing process and hardware selection is presented and wireless communication capability via remote control is explained. Software development for the Cerberus humanoid robot as well as the initial tests conducted on the prototype are addressed. The software package that is specifically developed for this biped includes the capability to describe the robot's gait on the screen graphically by the click of a mouse, which is then transparently converted into the P-Basic code to control the servos. This approach significantly saves time and eases software development in defining new gaits for the robot. Various gaits have been developed and tested to assess the capabilities and limitations of the robot. The key design feature of Cerberus is that it has minimal amount of degrees-of-freedom to perform tasks such as moving forward and backward, making turns in any direction, and walking in quadruped or in biped configuration.

INTRODUCTION

The motive behind building humanoids is simply to design a robot that can duplicate the complexities of human motion and help people. Although this motive seems simple, the goal is quite challenging to achieve. For instance, it took Honda engineers more than 18 years of persistent study, research, and trial and error before they achieved their dream of creating an advanced humanoid robot named ASIMO [1].

Biped walking can be investigated by using either a static or dynamic walking model. Static walking in bipedal humanoids keeps body's Center of Gravity (COG) over the base foot area throughout the motion. Such robots are designed and controlled from a kinematic standpoint (trajectory or displacement-controlled), and as a consequence, they have relatively large feet and they walk at a slow speed. A static-walking biped, such as the Honda P3 Humanoid, "does not move quite like people do and is energetically inefficient… it moves with a nonpendular appearance and uses about 2 kW during walking—more than 20 times the muscle work rate of a walking human of the same size" [1]. Dynamic stability is needed to walk quickly and over varied terrains. In this case, the center of gravity can lie outside the supporting leg base area during walking; the robot tumbles forward to its next step in dynamic equilibrium.

Passive-dynamic walking can be added as a third group to the list of different types of walking models. Un-powered toy soldiers or penguins were constructed as early as a century ago that could walk down a gentle incline without any motor control. Through the careful selection of the lengths and masses of their arms and legs, these toys maintain their balance while consuming very little energy (from gravity) when walking. Such models walk in a rigid manner, but their constructions are simple. Using this as the starting point, more degrees of freedom (DOF) can be added. These DOFs may be powered and controlled to produce a more fluid walking motion.

The goal of this study is to design, construct and control a robot with minimal degrees-of-freedom that can perform rather a wide range of gaits including bipedal and quadruped walking. The reason to have quadruped motion capability in Cerberus is to enhance its mobility even when walking on inclined surfaces where biped motion is difficult to maintain. Quadruped configuration will also be useful to avoid obstacles since it will allow the robot to lower its height. Another aim in the robot design is to have minimal amount of sensors. Therefore, a static walker with the capability of changing its mode from a biped to quadruped walker is developed. The next section gives a brief history of the humanoid studies up to now. Later, a detailed explanation of the final design and the preliminary gait definitions are presented to complete the design of the Cerberus.

The sections that follow the design of Cerberus describe the parts selection and robot construction. In these sections, selection of the main body parts, the electronics, the servomotors and the controller card is presented.

Following this, the prototype that is built is described through its construction phases.

The last part of the paper is dedicated to the software development and gait creation for the Cerberus. The unique software package that is developed to create gaits for the Cerberus is explained in detail. Some screen shots with their explanations are used to show the functions of the software. Finally, the conclusions and future work are addressed.

HISTORY OF HUMANOIDS

Robotic studies and applications have shown a great increase in the past thirty years. Robots were first used for industrial purposes in assembly lines. As they evolved and became more "intelligent," their interaction with humans in daily life increased.

As the research on humanoids is increased, the cost of producing humanoids for daily life needs is decreased. Some of the humanoid projects that are developed at various universities and some commercially available humanoids are briefly reviewed below.

The National University of Singapore has a bipedal walking robot project underway shown in Figure 1(a). Their robot has 12 DOF and it is 1.2 meters tall. The joints are driven by DC motors [8].

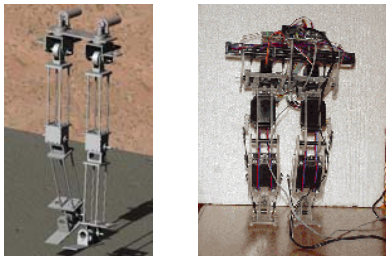

(a) (b)

Figure 1. (a) Biped by National University of Singapore, and (b) Biped by Alexander Vogler

Alexander Vogler from Vienna, Austria, built a bipedal robot, and named it V-3. It is 30cm tall and it weighs 1.2 kg. V-3 has 12 DOF [9].

Some of the commercially available humanoids are listed below along with their specifications.

Lynxmotion Robonova: Robonova manufactured by Lynxmotion is a sixteen servo biped walker that uses five degrees of freedom per leg. Robonova has the abilities to walk forward or backwards and to turn in place left or right with variable speeds [2].

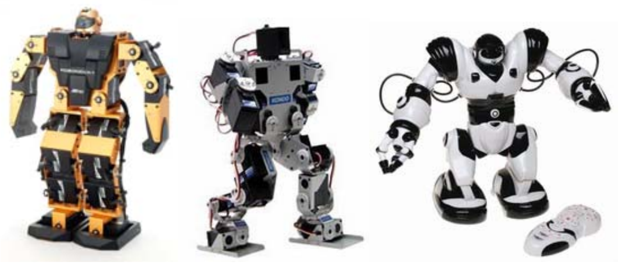

(a) (b) (c)

Figure 2. (a) Lynxmotion Robonova, (b) Kondo KHR-1 Humanoid, and (c) Robosapien

Kondo KHR-1 Humanoid Robot Kit: The KHR-1 Robot kit from Kondo has a total of 17 DOF. It can perform a variety of motions, such as walking, kung-fu fighting, soccer kicks and acrobatics. [3].

Wow Wee RoboSapien: RoboSapien is a biped walker and it is probably the first robot based on the science of applied biomorphic robotics as the designers state. It has the capability to walk forward and backward, turn in any direction, provide responses to the inputs from the remote controller by either singing or shouting. It has also tactile sensors to prevent possible collusions with obstacles [4].

HRP-2: HRP-2 was designed and manufactured by Kawada Industries, Inc. together with Humanoid Research Group of National Institute of Advanced Industrial Science and Technology (AIST). Yaskawa Electric Corporation provided the initial concept design for the arms and AIST 3D Vision Research Group and Shimizu Corporation provided the vision system. HRP-2 has a height of 154 cm and a mass of 58 kg including batteries. It has a total of 30 DOF [5].


(a) (b) (c)

Figure 3. (a) HRP-2, (b) HONDA ASIMO, and (c) SONY QRIO

HONDA ASIMO: This humanoid has a total of 26 DOF, a mass of 52 kg and a height of 1.20 meters [1]. It is a highly sophisticated humanoid that has various motion sensors incorporated. Honda engineers validate the size selection by declaring "The robot's size was chosen to allow it to operate freely in the human living space and to make it people-friendly." [1].

SONY QRIO: QRIO was built to walk on two feet, run and dance dynamically. Sony engineers used a new actuator for the joints to achieve a smoother actuation, which they called Intelligent Servo Actuator (ISA). They state "it made it possible to build a robot with compact body design that could move its body smoothly and dynamically" [6].

FINAL DESIGN DESCRIPTION

The main concept of the Cerberus was that it would have the capability to walk in biped mode and then change its mode to a quadruped walker.

The ability of the robot to maintain its stability while walking is an important design criterion. Final design is a static walker, where the robot keeps its center of gravity within the zone of stability. The abilities to become quadrupeds give the robot the ability to walk over a greater variety of surfaces. Final design has 8 DOF. The number of

DOF was minimized since having more degrees of freedom will make the robot more complex and costly.

The final design has the following features:

- Total of 8 degrees of freedom:
  - 3 dof on each leg
  - 1 dof on the waist
  - 1 dof for the arm actuation
- Reconfigurable: Biped to quadruped and quadruped to biped walking
- Controller: Basic Stamp II
- Servos: RC servos with potentiometers

For the process of kneeling down and changing mode to a four-legged robot, the robot required at least 3 DOF on each leg. Although it has this many DOFs on each leg, it cannot go side ways but it can only go forward and backward, and turn right or left. To add more capabilities, at least one more DOF should be added to each leg (which operates about the roll axis).

The DOF on the torso is used for balancing the robot while it walks on two feet as shown in the figure below. In order for the robot to walk, it uses its waist to shift the center of gravity from one side to the other as the corresponding leg takes a step. The design keeps the distance between legs to a minimum in order to reduce moments created about the center of gravity when a step is taken.

In the final design, once the robot has all for extremities on the ground, it propels itself forward by moving its rear legs in a crouched walking position while its upper limbs roll the wheels at the end of each extremity.

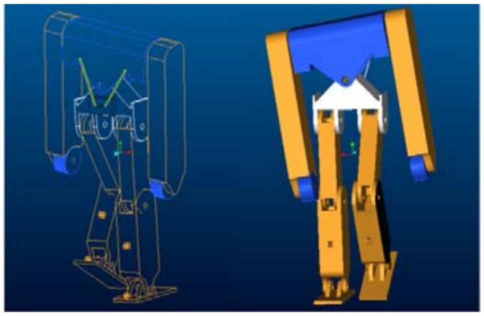


Figure 4. A gait of the second design concept

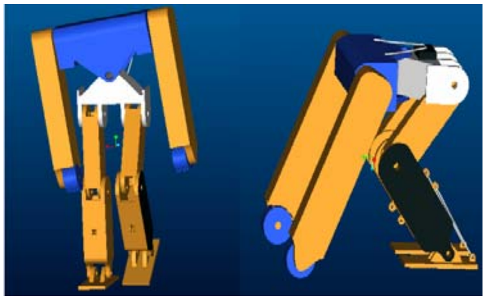


Figure 5. From biped to quadruped mode

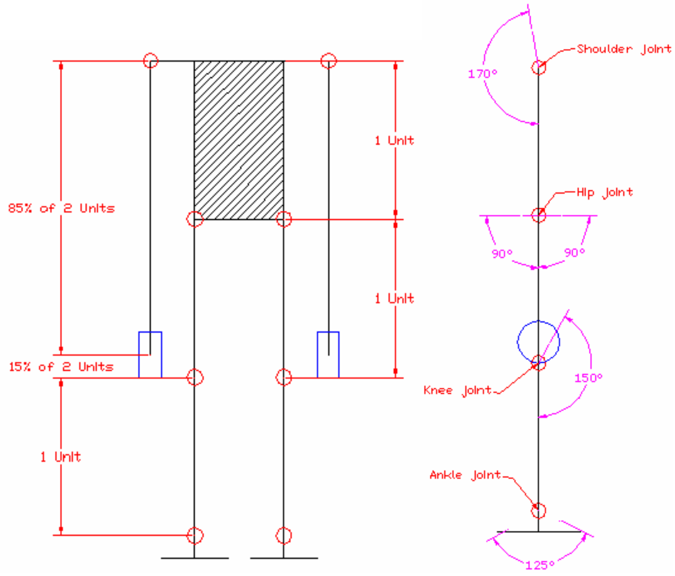
For the preliminary selection of the servos, the biped walking shown in Figure 4 can be considered as the worst-case scenario. The reason to select the configuration shown in Figure 4 as the worst-case scenario is that the robot stands on one foot. Also, the servomotor on the right ankle is the one carrying the most weight since the moment arm from the COG to the servomotor is the largest.

The total humanoid center of gravity can be assumed to be at the middle of the humanoid *0.15* m up from the ground and the weight can be estimated as *12* N as it is in [9]. While taking a step, it can be assumed that the COG is shifted by *10°*. In this static case the moment acting at the ankle servo of the right leg can be calculated as: $M_{ra} = 12 \times 0.15 \times \sin(10) = 0.312$ Nm or *44.2* oz-in.

It is important to note that in the early stages of the work deciding on the exact amounts for the link lengths was not advisable. Therefore, the humanoid robot's proportions were chosen in terms of a unit length and checked to make sure the robot's workspace requirements could be fulfilled.

Figure 6(a), shows how the robot's basic architecture is based on a unit value. All dimensions are multiples of the unit value. In our case, the upper and lower legs, and torso each has a dimension equal to 1 unit, while the arms are 2 units each. The preliminary unit value selection, based on the servo size and controller is *1 unit=10 cm*.

The servos chosen have a max range of motion of 180 degrees. Figure 6(b) below shows the desired range of motion at each joint. As the figure shows, none of the joints' range of motion exceeds the servos maximum range.



(a) (b)

Figure 6. (a) Humanoid proportions, (b) Humanoid range of motion

GAIT ANALYSIS

Preliminary gait analysis is done to see if the range of motion defined previously meets the requirements to walk in biped mode, to change the mode from biped to quadruped and finally to walk in quadruped mode.

The walking pattern synthesis is based on Zero Moment Point (ZMP). The ZMP is defined as the point on the ground about which the sum of all the moments of the active forces equals zero. If the ZMP is within the convex hull of all contact points between the feet and the ground, it is possible for the biped robot to walk. Hereafter, this convex hull of all contact points is called the stable region. As seen below the basic idea involves shifting the waist to the side opposite to the leg in motion in order to maintain stability.

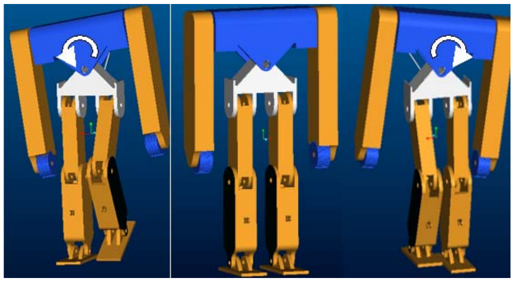


Figure 7. Waist motion for stability in biped walking

A general mode changing gait analysis was done graphically in Figure 8. The purpose was check to see that the leg and arm workspace was acceptable, check for limiting positions, and use data from gait "snapshots" as guides for controller algorithm development.

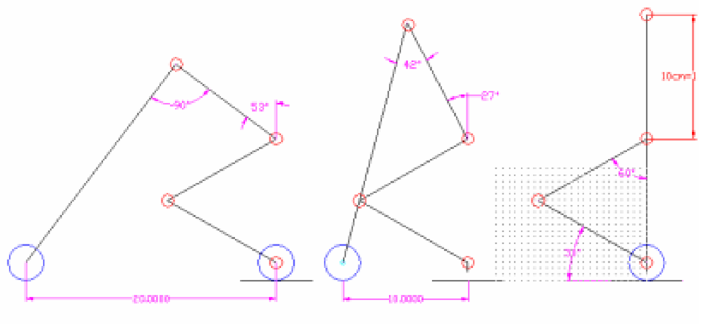


Figure 8. Gaits for changing from biped to quadruped

Figure 9, 10 and 11 show the snapshots of the robot quadruped walking gait pattern. As the figures show, the end positions for the first and second steps are the same; hence, the walking motion is a repetition of this gait.

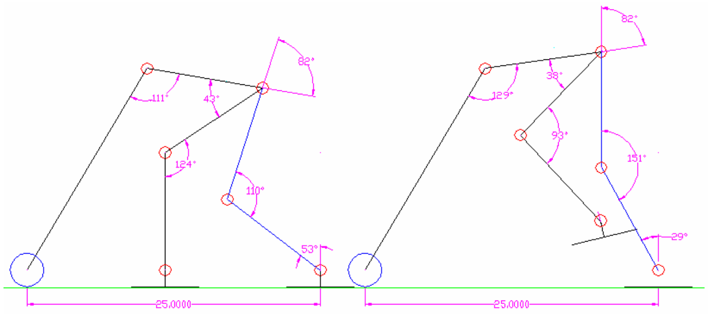


Figure 9. First step – left leg

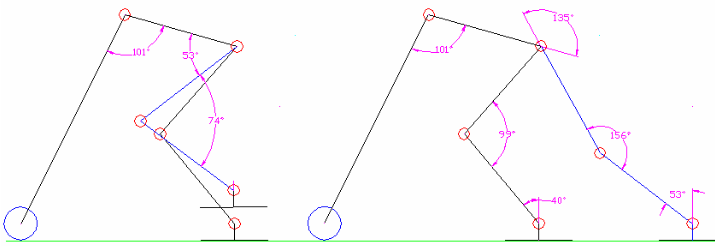


Figure 10. Second step – right leg part 1

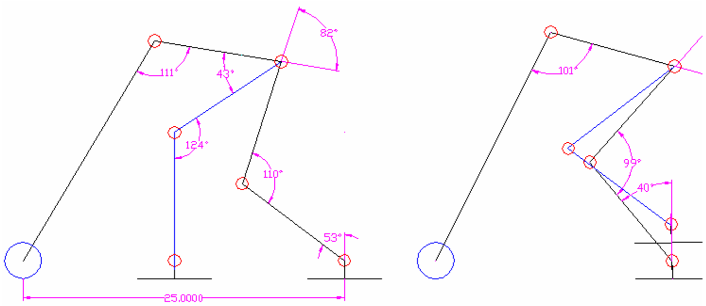


Figure 11. Second step – right leg part 2

REQUIRED PARTS FOR FINAL DESIGN

In the construction of small bipedal walkers (under 24" and 10lbs), weight reduction is considered as the most important issue. A lightweight biped reduces the size of motor required for operation (and lowers the associated cost). Further, a lightweight biped will allow all of its other essential components to be carried on board, such as its processor, sensors, and power supply. A survey of typical actuation systems and microcontrollers are discussed briefly.

The body of the robot is typically made of lightweight plastics such as Lexan or PVC panels. Some commercial models have used aluminum sheeting that a typically very thin. Other specialized design competitions have made use of Lego Mindstorm™ parts [10].

Standard servos with built-in position control capability are suited for the actuation of smaller robots. The limited 180 degrees rotation of standard servos was not a major drawback when used in bipedal robots, except for some arm models that require 360 degrees of articulation. The servos of continuous rotation type are more suitable for vehicular platforms. For much larger robots, pneumatic actuators are more common.

Commonly used microcontrollers for the control of small robots included the Parallax, Inc. Basic Stamp II™ and the Pontech SV203 servo controller. Larger laboratory robots can be controlled via PCs, which confines the robots to the vicinity of the PCs.

The final biped design required the use eight R/C servomotors with as many sets of servo holders and C-brackets. The Homework Board containing the Basic Stamp II microcontroller was used to hold and run programs that controlled the robot.

PART SELECTION

Standard servomotors were chosen for this study because of their low cost, ease of control, and low weight and compact size. The servomotors used in this study were the HS-422 standard servos with 57oz-in output torque at 6DCV. These motors weigh 1.66oz each, run on dual "oilite" bearings, and travel 60 degrees in 0.16 second. The motors have a torque to weight ratio of 34.3oz-in per oz, and a torque to price ratio of 4.4oz-in per dollar; these figures are among the best in the class of standard servos (under 100 oz-in range).

Figure 12. A single limb segment consisting of one standard servo, one servo holder, and one C-shaped bracket

The Basic Stamp II™ microcontroller was selected; it can hold up to 500 lines of code and control up to 16 servos simultaneously with its available sixteen (16) output ports. The Homework Board™, with a surface-mounted BS II™, was used because it contained all the necessary circuitry ready for immediate use (serial port, power supply, etc). The Homework Board™ thus has the added advantages of compact size, lightweight, and cost-effectiveness.

The remote control system came from a small remote-controlled toy car. The part is inexpensive, and most importantly, the receiver is compact enough to fit on the breadboard on the Homework Board™ and it is very lightweight. The purpose of having a remote control is to allow the user to issue motion commands to the robot remotely. These include the commands “go forward,” “go backward,” “right turn,” “left turn,” and so on.

A set of eight (8) servo holders and C-brackets was also used to facilitate ease of construction. Other parts of the robot, such as the feet, the chest, and the back, were made using 1/8” thick PVC panels.

CONSTRUCTION AND ASSEMBLY

The use of all servos in the construction of the robot was limited to identical units of servo-holder-C bracket combination setup in a modular configuration. One such unit segment is shown in Figure 12.

Each leg of the robot consists of three such modules. Both legs were constructed to move in the Y-Z plane only (see Figure 13(a)). The torso was fixed to the hip (consisted of a pair of upward-extending C-brackets). The torso motor however, was pivoted about the X-Y plane. The shifting of the center of gravity (COG) of the torso segment is the key in achieving static walking.

The selection of the final design concept with 8 degrees of freedom (DOF) meant that only one servo was used to actuate both arms (Figure 13(b)). This did not detract from the functionality of the arms since they were used only in assisting the torso to move between the upright and leaned-over positions. The two arms were made of the PVC panel and were fixed to the torso at the C-bracket via two threaded rods. The arms were passively telescoping and they can reach between 6” to 10”.

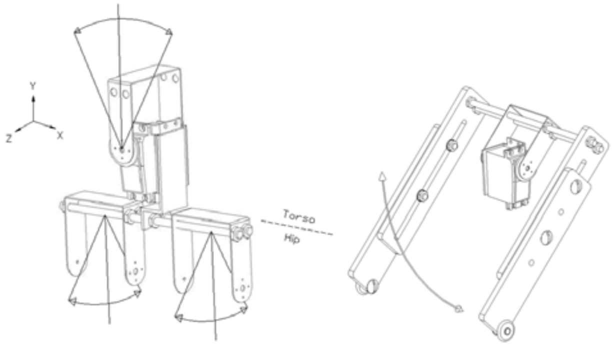


Figure 13. (a) The torso rotates about the Z-axis (X-Y plane) whiles the legs moves about the Y-Z plane, (b) A pair of telescoping arms actuated by a single servomotor

The controller was housed to the inside of the chest panel. The chest plate is hinged at the lower torso, and it can be swung open to expose the main circuitry for initial setup and subsequence maintenance (see Figure 14(a)).

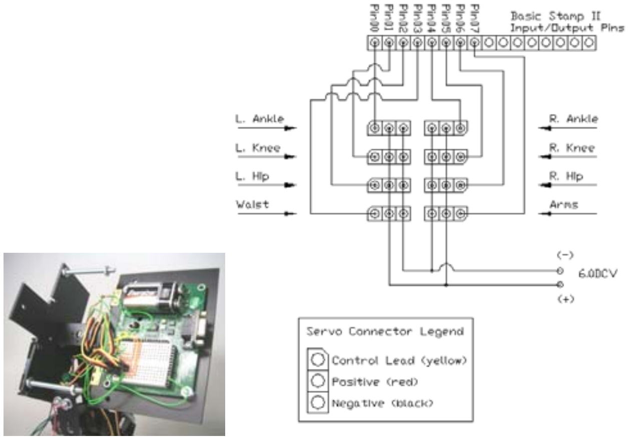


Figure 14. (a) The Homework Board™ as installed on the robot chest plate in the open position, (b) Circuit diagram showing the connection of eight (8) servomotors to the controller

All servos are placed in a parallel circuit with power supplied by a 4-pack AA battery. The parallel arrangement, as seen in Figure 4, allowed the servo connectors to be closely packed on the breadboard, thus leaving space for the remote control unit and other sensors (see Figure 14(b)). The battery pack is carried on the back (torso) of the robot and it helps to counter the weight of the controller on the chest.

The output signal that controls the DC motor (in the toy car) at the receiver end of the remote control system is used as an input to control the robot. The weak input signal is amplified using a transistor; the DC output port on the controller was tapped as the power supply, and the amplified positive voltage was registered as a “high” on one port designated as an input port.

The input port – pin 12 – was set to receive signals from the remote control transmitter unit. The “Vin” ports on the controller provide 9.0DCV to the collect pin of the transistor. When the receiver unit receives a “go forward” signal from the transmitter, the positive voltage signal its processor sent to the DC motor was redirected to the base pin of the transistor. This signal effectively acted as a switch and allows current to flow out of the emitter pin of the transistor, thereby registering a “high” on the controller input port. The negative lead from the receiver is grounded to the controller at the “Vss” port.

Only one control signal was necessary for the robot to perform many tasks. The robot can be made to walk, crouch, stand, and stop using an appropriate algorithm. The use of counters in the program is one way this can be accomplished; the program can also understand if a button is pressed several times or has been held pressed for a few seconds.

The completed version of the robot measured 15” in height and 7.5” wide at the shoulder. Each foot measured 2.75”x5.5”, and together covered a area of 5.75” width by 5.5” deep. The robot can stand upright un-powered, indicating its COG is properly centered within the base area.

Preliminary data indicated that the ankle joints tended to yield when the torso leaned forward or backward by more than 15 degrees. When set to stand on one leg, the ankle was also the joint that failed to hold the weight of the robot. The servos at both ankles were therefore replaced with the more powerful HS-700BB servomotor. The new motors weighed 3.6oz each, and can produce 174oz-in of torque using a 6DCV power supply. Servos at the knees and hip were not changed because the HS-700BB was heavier, and the correspondingly larger servo holders and C-bracket would increase the weight of the robot even further.

SOFTWARE DEVELOPMENT

Creating a defined walking gait for bipedal robot is a complex process dependent on several factors such as service of torque capacity, robot design (center gravity, link lengths), and the type of walking desired (static or dynamic). The process involved in moving from a theoretical walking gait to the physical realization of such gait by the robot involves several steps. First, there is the need to develop a walking pattern or gait that is appropriate for the given robot. These gaits can be solved using predefined trajectories of the feet [7] or software that will be explained in this section can be used to define gaits keeping the robot stable. Secondly, the gait has to be translated into an algorithm or code that can be interpreted properly by the given processor being used to control the robot’s servos. Finally, the algorithm/code must be compiled and read on to the processor by means of some type of software interface that allows communication between the controller and the computer. It was our intent to create a software package that would integrate the first two steps of this process, which are the most complex and time-consuming of the three. Through the use of a graphical 2D/3D interface for gait creation and automatic code generation, the amount of time spent developing a specific motion or gait and then producing the corresponding code will be dramatically reduced by use of this software application. This program was developed specifically for the Cerberus humanoid robot, using its geometry and range of motion to develop the graphical on-screen representation. Nevertheless, its general algorithm can ultimately be adapted for use with robots/mobile platforms of varying characteristics. The integration between the software and the robot was achieved using a Basic Stamp II Homework Board. The programs created using the gait generator software were compiled using the Parallax Basic Stamp Editor and loaded onto the controller using a serial connection. Figure 15 shows how physical robot links can be represented graphically.

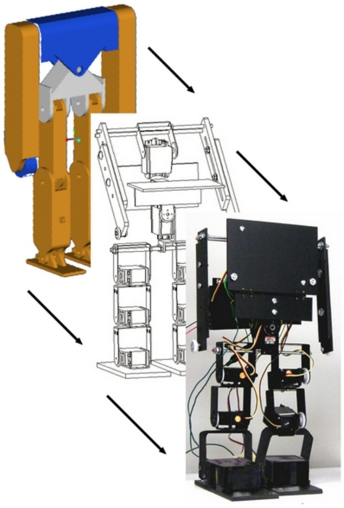


Figure15. The finished robot standing without applied power to its servomotors

Figure 16. (Right) Cerberus humanoid lower body with position markers. (left) Graphical representation of links throughout the gait using APPAS program

Servo Control

PBasic is the programming language used by the controller chosen and so the type of algorithm defining the walking gait of the robot is determined by said language. The method of control for standard servos involve using a simple for loop, which includes the PULSOUT command, used to active the servo. There are two types of loops that can be used in order to produce a prescribed motion; point-to-point and stepped increments. As the names implies, a point-to-point loop simply moves a servo from an initial angular position to the next desired position.

```
For i=0 To  100
PULSOUT left_hip,         1080
PULSOUT left_knee,         750
PULSOUT left_ankle,        740
PAUSE 17
Next
```

The code above is an example of a point-to-point loop. The servos for the left leg will rotate from their current position to positions prescribed. The rate at which the microchip processes information line by line is in the order of one millisecond per line, therefore, this change will happen rather abruptly even with the pause command included. Because of the lack of control with regards to the rate of change in the angular positions of the servos that the point-to-point method was replaced by the stepped increment approach in programming the walking gaits. Using step increment loops, one can control the rate at which angular position varies. The following is an example of how one would change the position of a servo(s) using this approach.

```
For i=0  To  100   Step  4
PULSOUT left_hip,          1080    - ( i * 3  )
PULSOUT left_knee,          750    -  (i * -2  )
PULSOUT left_ankle,         740    -  (i *  0  )
PAUSE 17
Next
```

The stepped increment method provides the capability to control both the speed and the final position. An increase in the value of the steps will increase the rate of change and decreasing the steps size will do the opposite. By using whole numbers as multipliers of the loop variable, the total change in angular position can be control. As seen from the sample code, the left hip would decrease by 300 increments, the left knee increases by 200 increments, and the left ankle would maintain its current position at the end of this loop. This type of loop is the basis for the walking algorithms used to program the Cerberus humanoid robot.

SOFTWARE DESCRIPTION AND DESIGN

The process of developing a proper gait and writing the corresponding code can be very time consuming. This is especially true when trial and error is involved in the process. In theory, devising a gait or prescribed motion involves the measurement of angles throughout the motion at the defined critical positions. Combining this process of "reverse engineering" along with writing, what often ends up being hundreds upon hundreds of lines of code, the trial and error approach becomes almost an impossibility. It was because of this that a software package, which simplifies and integrates these two processes, was developed using Visual Basic 6.0.

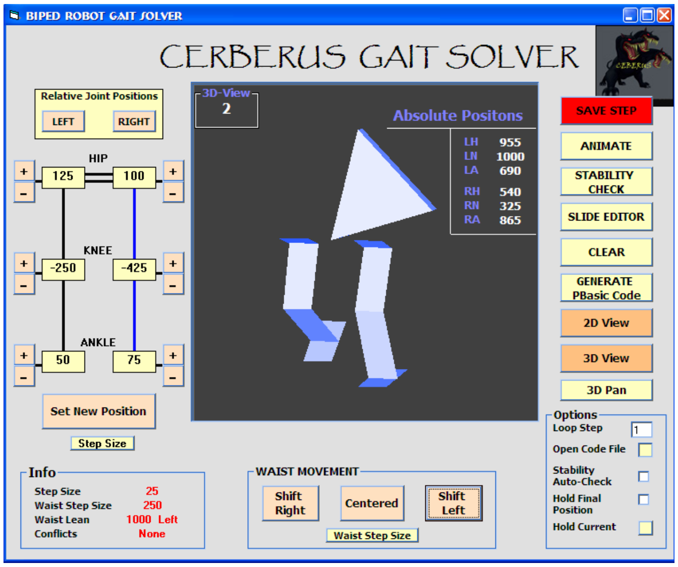
The basic idea behind the Cerberus Gait Solver (CGS) program is to allow the user to develop a specific walking gait or motion using a graphical representation of the robot. Once the user has created said gait or motion, the entire corresponding Pbasic code is generated with the click of a button.

Graphical Interface

The CGS program was developed to be as a user-friendly as possible. A two dimensional representation of the Cerberus robot is shown on the screen as links. The black lines represent the left leg while the blue lines represent the right leg. The interface was designed in such a way as to allow the user to change the positions of any of the joints by simply clicking the corresponding hip, knee, or ankle plus or minus buttons on the left-hand side of the interface.

The values entered here are relative joint positions with respect to the defined default position, which in our case is when the robot is standing with its legs perfectly straight. The absolute position values for each joint can be found on the lower part of the picture screen. The units that define both the absolute and relative position values are a result of the servos resolution and range of motion. The servos used have a maximum range of motion of 180 degrees and a resolution of 0.18 degree per unit increment. That is to say, a relative change of 100 is equivalent to an 18-degree change in the position of a servo. The program offers the user the ability to change the steps size or increment value in order to fine-tune the desired motion. Due to the fact that the Cerberus robot walks by means of shifting its waist position from left to right, the ability to include this in the development of the gait is made available in the Waist Movement box. The user can shift the waist to the left, right, or center as well as controlling the waist steps size. The reason for including this is to produce a code that is complete in terms of overall control of the robots gait. Stability Check feature was included to alert the user whether the waist is being shifted in the proper direction and/or to remind the user to shift the waist in the proper direction if he/she has failed to do so.

There are five general steps involved in producing the PBasic code for a desired gait or motion. They are: key frame/slide creation, stability check, animation, generation of PBasic, and slide editing.



Hip: 125 | 100
Knee: -250 | -425
Ankle: 50 | 75

3D-View 2

Absolute Positons

LH	955
LN	1000
LA	690
RH	540
RN	325
RA	865

Info

Step Size	25
Waist Step Size	250
Waist Lean	1000 Left
Conflicts	None

Options
Loop Step 1
Open Code File ☐
Stability Auto-Check ☐
Hold Final Position ☐
Hold Current ☐

Figure 17. CGS main window in 3D mode

Program Flexibility and Ease of Use

The CGS program was designed to be very flexible in terms of creating a gait. The user can create small sections of the gait, generate the code, and check to see if the robot responds in the desired fashion or simply check the animator. If changes need to be made, it can go back and make changes using the editor without having to start from the beginning. The program is set up so that you can continue adding slides to the gait after having tested the current ones. This allows for the creation of a gait or motion in one try or by segments. The flow chart below shows the flexible nature of the software.

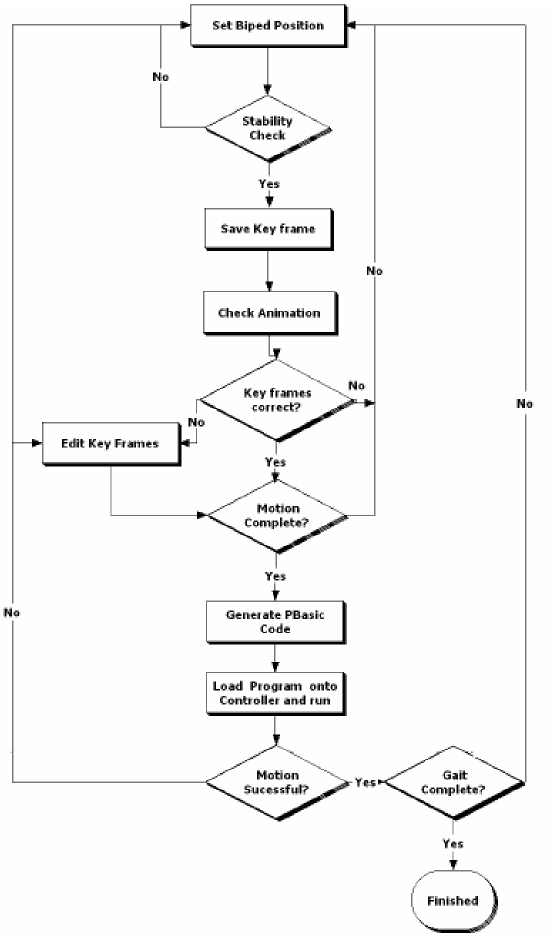


Figure 18. Flow chart of gait generation process

SOFTWARE AND HARDWARE INTEGRATION

At the heart of Cerberus's control system is the Basic Stamp II Homework Board. This controller uses a PIC 16C57 micro controller, which runs at about 20 ms. It has a built-in EPROM with a capacity of 500 lines of PBasic code or 2KB of memory and 32 bytes of RAM. The BS-2 has 16 ports, which can be designated as either input or output, depending on the needs. It connects to the PC using the normal support interface. In the case of the Cerberus, the BS-2 controls all eight servos of the robots joints.

After the code is entered into the editor, the Cerberus robot's control board is connected to the computer via a serial cable. The code is then run in the Editor and loaded onto the control board's EPROM. Once this is done, the robot will perform instructions programmed.

CONCLUSIONS

Cerberus robot that was designed and built in this work has a unique design with minimal DOF and sensors to accomplish biped as well as quadruped motion.

Humans have at least 16 major DOF in the legs and waist that allow them to walk. In contrast, Cerberus is designed to have a total of 7 DOF to walk. This reduced DOF limits its movements relative to a human, but simplifies the robot design. Even this reduced DOF proves to be a challenge when it comes to program the joints to produce a specific gait. To overcome this difficulty, a special software package named Cerberus Gait Solver (CGS) was developed.

Using CGS-produced gaits, it was possible to test the capabilities of the Cerberus robot in bipedal motion. Motions created using CGS's graphical interface were reproduced accurately with a high trial and error rate. Ultimately, Cerberus took its first steps using a gait generated from the CGS. This initial gait produced a program that exceeded the processor's memory capacities. Hence, the gait was optimized for speed, accuracy, and stability while minimizing the size of the code using the slide editor. The final result was a gait that noticeably increased the physical capacities of the robot, producing an almost human-like walking pattern.

The development of the Cerberus Gait Solver made the trail-and-error process quick and simple. The time necessary to develop and test a given gait was reduced from hours to minutes. The program capabilities have proven to be invaluable in producing a robot with human-like walking gaits.

Several gaits are designed and successfully tested on the prototype. As a result of these tests, it is observed that the system needs to be optimized more to have more flexibilities in terms of various motions. This optimization needs to be carried out in body construction as well as in the servo selection and additional memory for the controller. Future work on this study is to complete the optimization stated above and to perform quadruped motion tests on the robot.

REFERENCES

[1] HONDA HUMUNOID ROBOT ASIMO, HONDA http://world.honda.com/ASIMO/ accessed April 2005.

[2] "Biped Scout", Lynxmotion http://www.lynxmotion.com accessed April 2005.

[3] "Kondo KHR-1 Humanoid Robot Kit", Robot Shop http://www.robotshop.ca/c215005p16471614.2.html accessed April 2005.

[4] Robosapien http://www.wowwee.com/robosapien/robo1/robomain.html accessed April 2005.

[5] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K, Akachi, and T. Isozumi, "Humanoid Robot HRP-2" Proceedings of the 2004 IEEE, International Conference on Robotics & Automation, New Orleans, 2004.

[6] "SONY QRIO", SONY http://www.sony.net/SonyInfo/QRIO/ accessed April 2005.

[7] Qiang Huang, Kazuhito Yokoi, Shuuji Kajita." Planning Walking Patterns for a Biped Robot," IEEE Transactions on Robotics and Automation, Vol. 17, No. 3, June 2001.

[8] LUS Legged Locomotion Group http://guppy.mpe.nus.edu.sg/legged_group/ accessed April 2005.

[9] Alexander Vogler http://members.chello.at/alex.v/ accessed April 2005.

[10] Henrik Hautop Lund, Luigi Pagliarini, Leonid Paramonov, and Morten Winkler Jørgensen, "The VIKI 4. Humanoid - An Example of Embodied AI," Proceedings of the Third International Symposium on Human and Artificial Intelligence Systems: The Dynamic Systems Approach for Embodiment and Sociality (HART2002), Fukui, December 6-7, 2002.