Omnibot 2000: Development of an Autonomous Mobile Agent for the Disabled and Elderly

2000 Florida Conference on Recent Advances in Robotics
May 4-5, 2000, Florida Atlantic University
Scott D. Nortman, Dr. A. Arroyo, Dr. Eric M. Schwartz
Machine Intelligence Laboratory
University of Florida
Gainesville, Florida 32608

ABSTRACT

Omnibot 2000 is designed to be a personal assistant, capable of helping the elderly or disabled. Additionally, Omnibot 2000 can entertain and perform. Behaviors include obstacle avoidance, wall following, obeying commands, and performing. The user determines the behaviors using voice recognition. Commands are issued to Omnibot, and it responds by repeating the words, and performs the specified behavior. Omnibot contains four different sensor suites, including infrared emitters and detectors, bump switches, voice recognition, and low-resolution vision. During wall following behavior, Omnibot will turn away from objects in its path. When it is doing wall following, it will follow the walls of a room, and it will also avoid bumping into obstacles. When Omnibot is in its obeying commands behavior, the user can instructs it to move the arms, grippers, head, and body. Omnibot is a slave, performing any tasks the user requests. When it is told to dance, it will start singing and dancing to YMCA, or any other song programmed.

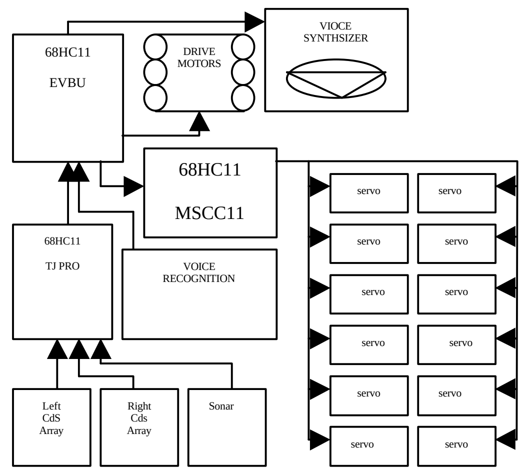
PHYSICAL SYSTEM

Electrical Subsystem

Omnibot's control suite consists of three Motorola 68HC11 microcontrollers. One is operating in single chip mode on a Mekatronix MSCC11 circuit board. The other two are operating in expanded multiplexed mode with 32K of SRAM. One is on a Mekatronix TJ Pro circuit board and the other is on a Motorola EVBU board. *Omnibot* also has a voice synthesizer module, voice recognition, low-resolution vision, 12 infrared emitters and detectors, and four bump sensors.

The block diagram of the control system is shown in Figure 1. The 68HC11 on the EVBU board executes the main program, and takes in data from the sensors. It also sends commands to the 68HC11 on the MSCC11 via the SCI port to control the servos.

Figure 1:



Actuation and Output Devices

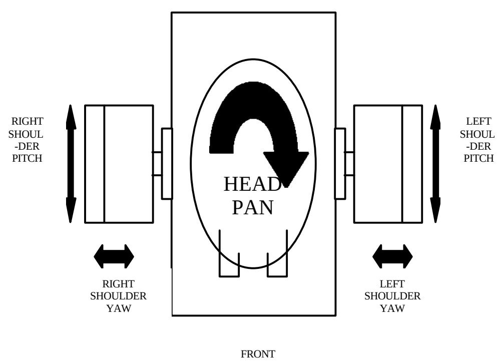
The goal of *Omnibot 2000* is to help and entertain people. Help is provided by means of the robotic arms; they can grasp objects, move the objects to a new location and then release the objects. For example, the arms can grab a can of soda, then lift the can, rotate the wrist to pour the soda into a glass, and then return the can to its original position. As for entertainment, the robot uses the arms to form the letters “Y”, “M”, “C”, and “A” while singing and dancing to The Village People’s *YMCA.*

The original toy robot had small DC motors in the right arm providing movement to the shoulder pitch, the rotation of the wrist, and the actuation of the gripper. However, these motors did not have enough torque for the desired application. Seven more degrees of freedom were added. This includes additional DOF to allow the head to pan, both shoulders move about their yaw axis, both elbows to pitch, and both grippers open and close, and both wrists to rotate. The same control algorithms were needed for all of the joints, therefore the motors that came with the robot were removed and replaced by servos.

To accomplish the specified goals, *Omnibot 2000* needs servos that can supply enough torque to move all of the joints. The servo torque must also provide strength for the arms to lift objects in the grippers. The grippers also need high torque servos to grasp objects without slipping. Due to these requirements, *Omnibot 2000* uses precision Cirrus Hobbies servos.

The shoulder joints have two degrees of freedom along the pitch axis and yaw axis. They may be moved independently, each using a CS-600 servo. The servos provide 333 oz-in. of torque, which allow the arms to rotate about each axis and lift objects. The pitch axis of each elbow uses a CS-400 servo, which provides 180 oz.-in. of torque. Each wrist rotates about its roll axis, and uses a CS-50 servo with 45 oz.-in. of torque. Finally, each gripper uses a CS-80MG servo with 130 oz.-in. of torque allowing the gripper to open and close. Figure 2 shows the movements for *Omnibot 2000*.

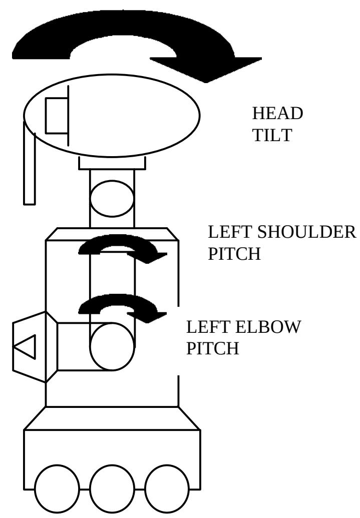
Figure 2(a):
TOP VIEW



FRONT VIEW (2b)

A Motorola 68HC11 microcontroller on a Mekatronix MSCC11 circuit board, operating in single chip mode, controls the servos. This microcontroller generates the pulse width modulation (PWM) signals needed by the servos. The signal is a TTL level pulse, with a period of 20 milliseconds. The duty cycle varies between zero and ten percent. A zero percent duty cycle causes the servo to position the output shaft at zero degrees, while a ten percent duty cycle positions the output shaft at 180 degrees.

LEFT VIEW (2c)



The single chip microcontroller receives data from another 68HC11 microcontroller through the asynchronous serial port (SCI). The transferred data frame contains three bytes for each action performed by the servo. The first byte is a zero, indicating the beginning of a frame. The second byte contains a number between zero and 15, determining which servo should be moved. The third and final byte of the frame is a number from zero to 128, indicating the position the selected servo should move to.

Although the above method of control is easy to implement, there are some disadvantages to using the SCI port. First, because the frame of data is transferred with the SCI port, a terminal may not be used to troubleshoot *Omnibot* during debugging or during use. Second, when any code is downloaded to the main microcontroller, the SCI lines must be disconnected from the main microcontroller because the data will also be transferred to the single chip controller, causing the servos to move to undesired positions. An alternate to disconnecting the SCI lines is to add a separate power switch to the single chip microcontroller so the PWM signals cannot be generated.

For *Omnibot 2000* to move around, drive motors are used. The base contains two drive motors from the original design. They are housed in a gearbox between the sets of wheels. The motors are gear-reduced to increase their output torque and reduce their speed. There is a transmission feature in the gearbox, allowing the output of the gearbox to have a "high speed" and a "low speed" selection. This feature was not used for the current design, and the transmission was left in the "high speed" position. The drive motors and gearbox do have some limitations. They were not designed to carry the weight of the modified *Omnibot 2000*.

The low current requirements of the drive motors permitted the use of a single chip motor driver. The chip is a Texas Instruments L293D Quadruple Half H Driver. The chip can provide enough current, however a heat sink was needed because the chip is designed to handle 1200 mA per coil and the measured stall current is 1270 mA for each coil.

Another Motorola 68HC11 microcontroller, operating in expanded multiplexed mode, provides the control signals. This controller is on an EVBU board, with a Mekatronix ME11 daughter board containing 32k of SRAM, a 74HC390 clock divider chip, and a memory mapped output port. Each of the drive motors requires two control signals from the microcontroller, the first for direction and the second for speed. The direction control lines used are port D bits 4 and 5. The speed control lines are from port A, bits 5 and 6.

Sensors

Omnibot 2000 has four different sensor suites. The first uses 12 infrared (IR) emitters and detectors, arranged around the base, to determine the proximity of objects from the base. The second suite consists of four bump sensors. Two of the bump sensors are in the front of the base, and the other two are in the back. Next is voice recognition. This sensor is actually located in a separate, wearable module. The final sensor is low-resolution vision.

There are 12 IR emitter/ detector pairs arranged around the bottom of the base. They emit IR light modulated at 40kHz. The emitters are connected to a memory mapped output latch. Writing a "one" to the corresponding bit at address 0x7000 turns them on. The detectors are modified Sharp GP1U58Y IR sensors. They have been modified so they output an analog voltage in response to detected IR light. Therefore, the returned voltage is in proportion to the proximity of objects. The eight detectors are connected to port E of the 68HC11 operating in expanded multiplexed mode. Port E connects them to the internal analog to digital (AD) converter. The IR sensors will prevent *Omnibot* from contacting any objects. However, if it does hit an object, the bump sensor will indicate a collision. The robot will then say, "Ouch!" and either back up or turn away or speed up, depending on which bump sensor(s) was contacted. If both front and back sensors are pressed at the same time, it will shut off the motors to prevent the motor driver from overheating.

Voice recognition was chosen because speech is a natural way to communicate. Software that performs speech recognition is currently available for personal computers. These programs operate simultaneously with the computer operating system. However, the disadvantages include the requirement of a compatible sound card, and the necessity of a desktop or laptop computer. While these speech programs are impressive, they cannot be inexpensively incorporated into an autonomous mobile robot. Fortunately, another option is available.

Another more viable voice sensor approach is with hardware that does not require an external computer. The use of this sensor on an autonomous mobile robot is made possible by a Hamatsu HM2007 IC chip. The chip provides the options of recognizing either 40 words .96 second long or 20 words 1.92 seconds long. The designed circuit selects the .96-second word length enabling the chip to recognize 40 independent words.

Although only 40 individual words may be recognized, these words control all of the functions of *Omnibot*. The words that may be understood are shown in Figure 5, and may be used to change behaviors. Accordingly, if a user says, "Slave," *Omnibot* will stop its current behavior and respond to spoken commands only, as opposed to its other sensor readings. If the user says, "Avoid," *Omnibot* will avoid obstacles. Additionally, these words may be used in combination so simple phrases may be understood. For example, the phrase, "Rotate left shoulder forward nine zero degrees," will be understood and it will then rotate its left shoulder forward 90 degrees.

Figure 3:

WORD NUMBER	WORD	WORD NUMBER	WORD
1	ONE	21	ROTATE
2	TWO	22	GRIPPER
3	THREE	23	WRIST
4	FOUR	24	SHOULDER
5	FIVE	25	ELBOW
6	SIX	26	HEAD
7	SEVEN	27	NECK
8	EIGHT	28	DANCE
9	NINE	29	AVOID
10	ZERO	30	SLAVE
11	FORWARD	31	FOLLOW
12	BACKWARD	32	PITCH
13	LEFT	33	YAW
14	RIGHT	34	MINUS
15	BODY	35	PLUS
16	BEGIN	36	CLEAR
17	STOP	37	INTRO
18	MOVE	38	SHOW
19	OPEN	39	*EXTRA*
20	CLOSE	40	*EXTRA*

Since the recognition is dependent upon the proximity of the user's mouth to the microphone, it could not be mounted on *Omnibot*. This would cause inconsistencies in recognizing the words. Due to these inconsistencies, the sensor is positioned in a separate module. This module is self-contained and wearable. It contains the HM2007 IC and additional circuitry required for transmitting the data via an RF signal. The module has a 12-button keypad, with buttons for 0-9, train, and clear. It also has two seven segment displays that indicate which word, if any was recognized. There is also a microphone/antenna jack, an on/off switch, and a status indicator LED (Figure 4).

Figure 4:

The voice recognition module may be trained by entering in the number of the word that is to be trained, then pressing the "train" button, and finally saying the word. The status LED blinks after the word is trained, and the seven segment displays show the number of the trained word. The process is then repeated for all of the words. To clear a word, the number of the word that the user wishes to clear is entered, and then the "clear" button is pressed.

The HM2007 is a digital signal microprocessor. It receives input from the microphone and samples the signal and a fixed frequency. The data collected during the sampling is stored in the non-volatile static random access memory (NVSRAM). When a new word is heard by the HM2007, it collects the data and stores it to a temporary location. It then compares the results of the new data to the data of known words stored in memory. If the stored data matches within an allowable range (accounting for allowable error), the processor indicates a match by placing the data on the data bus and asserting a data enable signal. This causes the data to be latched and displayed on the two seven segment displays.

This data must also be sent to *Omnibot.* This is accomplished by using a Holtek 640 encoder, a Holtek 648L decoder, a Linx Technologies TXM transmitter, and a Linx Technologies RXM receiver. The encoder receives the data from the data bus, and upon receiving the enable signal converts and encodes the data into a serial format, which can be transmitted via the RF modules, and then decoded be the Holtek IC. The data is then latched to the microprocessor with a memory-mapped input/output port. The block diagram is shown in Figure 5 and the schematic is shown in Figure 6.

Figure 5:

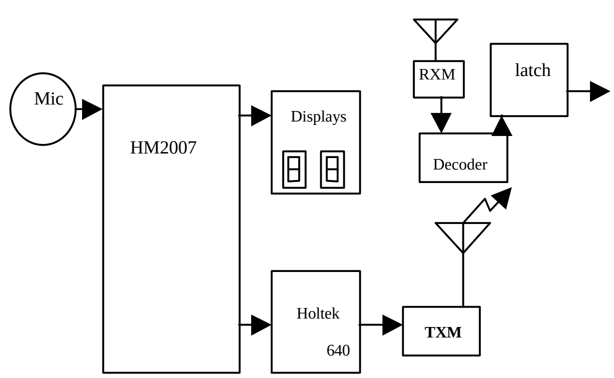
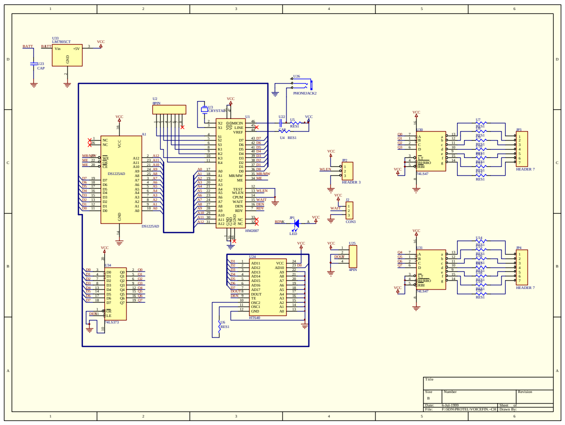
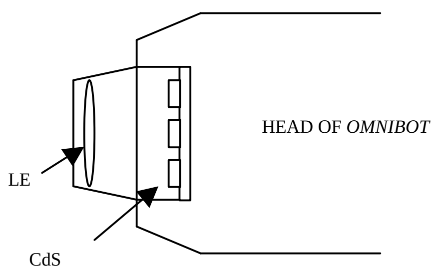


Figure 6:



The fourth and final sensor suite is a low-resolution vision system. This suite contains 50 CdS cells, seven analog multiplexers, a sonar emitter circuit, a sonar receiver circuit, and a Mekatronix TJ pro board with a Motorola 68HC11 microcontroller operating in expanded multiplexed mode with 32k of SRAM. In addition to the electronics, each eye contains a 50 mm lens, focusing images on each of the two CdS arrays, as shown in Figure 7.

Figure 7:



Behaviors

Omnibot 2000 is currently capable of four behaviors. These include obstacle avoidance, wall following, obeying commands, and entertaining. The behaviors are selected by issuing voice commands. By saying, "Begin avoid" it will start the avoidance behavior. The other behaviors are started in a similar fashion. *Omnibot* repeats the words that it hears, so the user will know if the command has been understood. Once a complete command is understood, *Omnibot* will then say a phrase indicating that it has understood the commands, and then executes the desired task.

The avoidance behavior uses IR emitters and detectors. Three are in the front of the robot, and three are in the rear. The sensors return a value between 88 and 120. There are no objects being detected when 88 is returned, and when 128 is returned the sensors are saturated. There is an avoidance threshold set at 100, and when the sensors return a value higher than that, *Omnibot* turns away from the corresponding high sensor direction. Additionally, if the sensors in the rear return a value higher than the avoidance threshold, the robot speeds up to get away from the object. This is obstacle avoidance behavior.

The wall following behavior uses the IR emitters and detectors. Three are on the left side, and three are on the right side. The values returned by the detectors on a side are compared to the threshold, and when the forward detector is above it and the rear detector is below it, *Omnibot* starts to turn away, and vice-versa. This situation is the same for the left and right sides.

The next behavior is called "slave" mode. The robot will listen to spoken instructions, and then perform the desired task. The words are used in combinations to make up phrases. The phrases that *Omnibot 2000* will understand are:

1. ROTATE(21) LEFT(13) SHOULDER(24) PITCH(32) MINUS(34) <AMOUNT>
2. ROTATE(21) LEFT(13) SHOULDER(24) PITCH(32) PLUS(35) <AMOUNT>
3. ROTATE(21) LEFT(13) SHOULDER(24) YAW(33) MINUS(34) <AMOUNT>
4. ROTATE(21) LEFT(13) SHOULDER(24) YAW(33) PLUS(35) <AMOUNT>
5. ROTATE(21) LEFT(13) ELBOW(25) MINUS(34) <AMOUNT>
6. ROTATE(21) LEFT(13) ELBOW(25) PLUS(35) <AMOUNT>
7. ROTATE(21) LEFT(13) WRIST(23) MINUS(34) <AMOUNT>
8. ROTATE(21) LEFT(13) WRIST(23) PLUS(35) <AMOUNT>
9. ROTATE(21) RIGHT(14) SHOULDER(24) PITCH(32) MINUS(34) <AMOUNT>
10. ROTATE(21) RIGHT(14) SHOULDER(24) PITCH(32) PLUS(35) <AMOUNT>
11. ROTATE(21) RIGHT(14) SHOULDER(24) YAW(33) MINUS(34) <AMOUNT>
12. ROTATE(21) RIGHT(14) SHOULDER(24) YAW(33) PLUS(35) <AMOUNT>
13. ROTATE(21) RIGHT(14) ELBOW(25) MINUS(34) <AMOUNT>
14. ROTATE(21) RIGHT(14) ELBOW(25) PLUS(35) <AMOUNT>
15. ROTATE(21) RIGHT(14) WRIST(23) MINUS(34) <AMOUNT>
16. ROTATE(21) RIGHT(14) WRIST(23) PLUS(35) <AMOUNT>
17. ROTATE(21) HEAD(26) MINUS(34) <AMOUNT>
18. ROTATE(21) HEAD(26) PLUS(35) <AMOUNT>
19. ROTATE(21) NECK(27) MINUS(34) <AMOUNT>
20. ROTATE(21) NECK(27) PLUS(35) <AMOUNT>
21. BODY(15) LEFT(13)<AMOUNT>
22. BODY(15) RIGHT(14) <AMOUNT>

23. BODY(15) FORWARD(11) <AMOUNT>
24. BODY(15) BACKWARD(12) <AMOUNT>
35. GRIPPER(22) LEFT(13) MINUS(34) <AMOUNT>
36. GRIPPER(22) LEFT(13) PLUS(35) <AMOUNT>
37. GRIPPER(22) RIGHT(14) MINUS(34) <AMOUNT>
38. GRIPPER(22) RIGHT(14) PLUS(35) <AMOUNT>

<AMOUNT> indicates that the user is to say two digits, each zero through nine, indicating the position the servo is to move to. However, when issuing any commands that start with the command word "BODY," <AMOUNT> indicates the amount of time *Omnibot* is to move in the specified direction in tenths of a second. For example, if the user wanted it to look straight ahead (head_pan servo equal to 0 degrees), the command would be, "ROTATE HEAD MINUS (or PLUS) ZERO ZERO." If the user wanted it to move forward for half a second, the command would be, "BODY FORWARD ZERO FIVE." By using these commands, the user can have *Omnibot 2000* perform basic assistive tasks.

The final behavior was developed to make *Omnibot* sing and dance to a song. The user programs in the spoken text, in combination with body movements, and *Omnibot 2000* will sing and dance. For example, the lyrics to the song *YMCA* were programmed into *Omnibot*, along with the body gestures. When the correct command words are spoken, *Omnibot 2000* will sing and dance to *YMCA.*

CONCLUSION

Omnibot 2000 is a successful proof of concept platform for the development of assistive technology. The combination of voice recognition integrated with a voice synthesizer provides a natural control interface. The benefits of a device such as *Omnibot 2000* include helping individuals gain independence if they require help for basic everyday tasks. A secondary benefit of *Omnibot* is that it has an inherent entertainment value due to its human like appearance.

REFERENCES

[1] Martin, Fred, The 6.270 Robot Builders Guide. MIT, 1992.

[2] Mekatronix ME11 Manual, 1995

[4] Mekatronix TJ Pro Manual, 1995

[3] J.L. Jones, B. A. Seiger, A. M. Flynn, Mobile Robots Inspiration to Implementation, Second Edition. A K Peters, Natick, MA, 1999.