



Magnets on the bottom of the cards aid in placement.

STONEHENGE

A Robotic Digital Clock

CrustCrawler arm entertains as it tracks time

by Steve Norris steven@botmag.com

I've always had a fascination with clocks. Whether by mechanical or electronic means, I've always been interested in the measurement of time. I built my first digital clock way back in 1975 (it's still running), and I recently built a clock that displays time in binary digits. Only a true geek can read it. But we are now in the second renaissance of robotics (I consider the 1980s to be the first), so I was determined to combine my love of clocks with my love of robotics.

Over the past few years, I've seen various videos of industrial robot arms simulating analog clocks. They usually accomplish this by moving around pegs that indicate the hours and minutes in a circle that represents a clock face. Although I could have taken this approach, I instead chose to emulate a digital clock much like the one that sits on your night-stand and wakes you in the morning. I thought this would be far more interesting and, as far as I knew, it had not been done before. Needing more precision than a typical servo-based robotic, I decided that the CrustCrawler Smart Arm (reviewed in the July-August 2008 issue) would be perfect for this project. The Smart Arm is unique in its use of a new type of actuator—the Dynamixel AX-12 by Robotis.

DYNAMIXEL AX-12

The AX-12 robot actuator contains a precision DC motor, gears and electronics all in one modular package. It has 300 degrees of movement in 1,024 increments with full control over speed and torque. The AX-12 has its own internal controller and uses a half-duplex network to receive commands and send back responses. This simplifies the cabling needed because you simply daisy-chain the AX-12 together using three conductor cables.

The default baud rate is 1M BPS, but it can be reduced to

allow the use of slower microcontrollers such as the BASIC Stamp. Each AX-12 is assigned a unique ID that identifies it within the network. Using the ID, the main controller addresses and commands the AX-12 by writing to its internal control table. In addition, status and real-time operational information can be read from this table.

The AX-12 has built-in thermal-shutdown capability and can automatically shut down when heat thresholds are exceeded. With position, voltage, temperature, speed and load feedback capabilities, it's possible to build applications that can react to overheating, overloading and errors in positioning.

CRUSTCRAWLER SMART ARM

The Smart Arm is made of anodized, 0.063-gauge, 5052, brushed finished aluminum components. It uses seven AX-12 actuators to provide four degrees of freedom plus the gripper. Four of the AX-12s are paired for the wrist and shoulder joints, giving each of these joints additional torque while the base, wrist-rotate and gripper joints each use one AX-12. The base has four carbon-steel ball bearings to handle heavier loads and an eight-way adjustable angle bracket. The gripper has a built-in adjustable sensor mount for installing cameras or other sensors, and it has slotted gripper paddles for attaching pressure and touch sensors.

THE CLOCKWORKS

The heart of the Stonehenge clock mechanism is the robotic arm that is used to select, grasp and move the necessary digits from the storage area to the display area and back again. The display area is directly in front of the arm and consists of four positions used to hold the current time in hours and minutes.

PHOTOS BY STEVE NORRIS

NGE

It's 12:47.
Do you know where your robot is?



There are two storage areas—one to the left and one to the right of the arm and arranged in a semicircle equally placed around the arm. Each storage area has seven positions, giving a total of 14 storage positions. Each position can hold one display “card.” A card has two sides and is used to display one of two digits. The front of the card has an odd number printed on it and the back has even number. There are 14 storage positions that give us a total of 28 displayable digits. Why do we need 28 digits?

I chose to have this clock display the time in 12-hour format. The hour 10’s display position requires a 0 or a 1 (one card). The hour 1’s position needs the digits 0 through 9 (five cards). The 2-minute display positions require 0 through 5 (three cards) and 0 through 9 (five cards), respectively. So, as you can see, to display the time, we need 28 digits printed on 14 cards. There may be clever ways to reduce the card count by reusing digits, but I chose this system because of its symmetrical simplicity.

One of my early problems with the clock was its inability to consistently place a card in a given position. Despite the excellent positioning accuracy of the Smart Arm, there were inevitability small positioning errors that crept in when it moved the cards. The cards have a tendency to shift when being gripped and lifted. These small errors accumulate over time and result in the misplacement of a card. To compensate for this error, two neodymium magnets (grade N42) are glued into place on the bottom of each card. Each position in the storage and display areas has a corresponding set of 1/4-inch 4-40 machine screws that will mate with the magnets in the card. Using this magnet scheme, a card can be off by as much as 1/2 inch or more and still be accurately positioned.

CONSTRUCTION

I made the project entirely out of the 1/4-inch foamboard that’s available at any office-supply store. Having been glued with hot glue or Duco cement, foamboard can be remarkably rigid. The base is 20 inches wide by 22 inches deep. To aid layout and construction, I marked off the entire base in 1-inch squares. The Smart Arm is mounted in the center on a 3 1/2-inch-high square foamboard box. This box provides the additional height needed for the Arm to be able to reach down and pick up a card. I also made the 1 1/2x4-inch cards out of foamboard and mounted strips of foamboard on the top and bottom. They resemble I-beams when viewed from the edge. These strips provide stability when placing a card and more surface area for the gripper to grab on the top. The digits were created in Photoshop, printed on heavy photo paper and then glued to the cards to give them a deceptively professional look.

With the cards in their storage positions, the clock has a Stonehenge look—hence, its name. You never know; perhaps the ancient Druids built a now long-gone robotic arm in the middle of the real Stonehenge. And since they didn’t have foamboard, they had to use stone.

THE CLOCK BRAIN

Stonehenge uses the Parallax Propeller chip as the main controller. As you may know, the Propeller is actually eight individual microcontrollers, called “Cogs,” built into one chip. Each Cog has its own memory in addition to the main 32K of RAM that is shared by all eight. The Cogs can simultaneously perform tasks cooperatively or individually while sharing resources through a common hub. The Propeller has 32 I/O pins of which four are generally pre-allocated to an external EEPROM and the serial programming lines.

To simplify the construction of the controller, I used the Propeller Demo Board. It includes a built-in Propeller, EEPROM and a 5MHz

crystal prewired to connectors for interfacing to devices such as a mouse, a keyboard, a TV, or a VGA monitor and speakers. Using the demo board’s breadboard area, the Propeller is easily connected to the Smart Arm’s AX-12 network. In addition, a two-line LCD and three pushbuttons are connected. These are used to implement a simple user interface allowing the time to be set and starting, stopping and resetting the clock.

The Propeller’s speed allows it to communicate directly with the AX-12 at the default baud rate of 1M BPS. You can download the Propeller/AX-12 driver (written in Propeller assembly language by Mike Gebhard) directly from the CrustCrawler site. This driver provides all the functions needed to read and write to the AX-12. In turn, I wrote a Smart Arm wrapper object in Spin that encapsulates all the functions needed to position the Smart Arm.

SOFTWARE OPERATION

The Stonehenge application is written in the Propeller Spin language and uses the Smart Arm wrapper object already mentioned to control the arm. Five low-level software functions are used to manage the movement of cards within the clock.

1. Put card in display. This function puts the card currently in the gripper into a designated position in the display area.
2. Get card from display. This function retrieves a card from the designated position in the display area. It assumes that the gripper is empty.
3. Put card in storage. This function puts the card in the gripper into the designated position in the storage area.
4. Get card from storage. This function retrieves a card from the designated position in storage area. It assumes that the gripper is empty.
5. Flip card. This function will rotate the card currently held in the gripper. It is used to select the card’s even or odd digit side.

These five simple operators are the building blocks for creating the sequences needed to display the time. Using the current state of the display area and the new desired state, the planning function (called RoboTime and its subordinate ShowDigit) determines which functions to use and the order of their execution. You can download the software from the link provided at the end of this article.

Although you may not want to build your own Stonehenge, the lesson to take away from this whimsical robot is its underlying design methodology. By taking a “deconstructionist” approach, you can create complex systems by breaking down the overall operation into a set of simple operators. These operators can then be combined and executed in any desired order to create seemingly complex behavior.

FUTURE WORK

There is still plenty of room on the base for additional cards. I plan to add a 24-hour display format as well as an a.m./p.m. indicator when using the 12-hour format. Adding an alarm capability like a gong or a bell that can be struck by the arm is another interesting possibility.

See a video of this clock in action and download the source code, mechanical drawing and schematic for it from www.botmag.com/issue13. ©

Links

CrustCrawler, www.crustcrawler.com, (480) 577-5557

K&J Magnetics, www.kjmagnetics.com

Parallax, www.parallax.com, (888) 512-1024

Robotis, www.robotis.com

Steve Norris website, www.norrislabs.com

For more information, please see our source guide on page 89.



Propeller Demo Board