Howdy,

Last evening's DPRG virtual discussion brought up the subject of ultra-sonic sensors, in the course of which I described briefly, from memory, the technique and interface of the Polaroid sensors, now sold by SensComp. http://www.senscomp.com/

So I went back today and looked at the actual implementation and I was of course wrong. Hmmmm. Carl, we need to have some way to go back and correct the videos... :) Here's the correct information:

1. The standard Polaroid interface has 4 pins (plus power and ground): Init, Blank, Blank-Inhibit, and Echo return. The Blank signal is only used if you want to listen for multiple echo returns from a single ping. Else it can just be tied to ground. That's what I do. Likewise the Blank-Inhibit signal is only needed if you don't want to wait for the device's default blanking timeout, which controls how close the sensor can detect. So each sonar is now down to just 2 GPIO pins, INIT and ECHO.

2. However, the sonar ECHO return outputs are active low, and are open collector. So all sonar outputs can be wire-or'd together with a pullup. That means only one GPIO pin is required, no matter how many sonar are in the array. (assuming that they are always fired in sequence.) With a 4 sonar array, that only requires 5 GPIO pins.

3. But on my robots the Blank-Inhibit is used, to reset the initial blanking period to 0.75 ms. That gives a minimum distance of about 6 inches. Note that distances < 6 inches will also return 6 inches. Basically that means that the transducer gets an echo return as soon as it starts listening. As a practical matter a reflection between 0 and 6 inches away will always be reliably detected, even if the precise distance in unknown.

4. So each sonar on my robots needs two output control pins, INIT and INHIBIT, and the ECHO returns all use the same input pin. The stereo sonar array on RCAT thus uses 5 GPIO pins, and the quad sonar array on jBot uses 9.

5. The robots use a 100KHz sample clock user interrupt provided by the RTOS to measure sonar distance. The calculation returns distance in 10th inches, determined like this:

Speed of Sound in Air at Sea Level approx 1100 feet per second.
1100 * 12 inches * 10 tenth inches = 132000 tenth inches per second.
132000/100000 sample rate = 1.32 tenth inch per sample count.
1.32/2 round trip to target and back = 0.66 10th inches per count

So distance to target in 10th inches = sample_count * 0.66.

Later that value was calibrated against real-world measurements in my garage and refined to 0.65. The sensors can detect reliably to a 10th of an inch out to about 16 feet, and somewhat less reliably out to about 25 feet.

cheers!
dpa