

Project Ideas

Projects

Firstly, if you have an idea for a project, then talk to the demonstrators, partly to see if they think you will be able to complete it in the time available, and also to check that we can source any equipment required.

For some interesting sensors, have a look at www.sparkfun.com

Here are some ideas for projects. As well as providing some suggestions, they also give guidance to the expected complexity level for a project.

The projects offer a range of challenges in terms of hardware and software, and in implementation some are more open ended than others. You should also consider the extent to which you will be able to work incrementally, or whether everything must work before you can get a useful outcome.

In some cases the project has been attempted before, but in others the idea is untried. We expect you to do preliminary calculations to see if what you hope to do is possible, but you won't be penalised if the project doesn't work fully due to something you couldn't have foreseen (for example a sensor not working as well as the manufacturer claims).

To make writing your report easier, take notes in a lab book as you progress, including feasibility calculations (for example how much storage is needed), design options, preliminary designs, tests of intermediate steps, and also what didn't work as expected, and why.

Consider at the design stage how you will test that your project is working as expected. Measurements or graphs of parameters like current consumption will allow you to show an expected battery lifetime for battery powered devices.

So far you have used prototyping boards, plus a printed Circuit Board (PCB) to interface to the LCD. Many of the projects require the same set of basic parts, so we have made a PCB available. Details are available here: http://www.cl.cam.ac.uk/teaching/1415/P31/docs/project_pcb.pdf

Alternative ideas

Owing to time constraints, not all of the features of the ATMEGA series of devices have been specifically covered in the workbooks, for example:

- I2C peripherals, such as fast AtoD converters, memory, sensors. There is library support for I2C available.
- Pulse Width Modulation, often used to control servos.
- Digital to Analogue conversion (using devices with a DtoA built in or resistor ladders)
- Analogue to Digital conversion using external AtoD converters, which may be faster or more accurate than the ones in the ATMEGA644P.
- 1 wire serial devices

However they may prove invaluable either in the suggested projects or if you have an idea for a project you would like to try.

Here are some example devices with part numbers

- DS3231 Precision Real Time Clock using the I2C bus. This device has an error of about 1 minute per year.
- ADXL345 3 axis Accelerometer which uses the SPI bus.
- SHT71 Temperature and humidity sensor on the I2C bus.

- GPS receivers like the Quinetic Q20 which output RS232 serial.
- SCP1000 pressure sensor with both I2C and SPI bus connections.

Project suggestions

Data logger

Consider making a logger which records a set of readings referenced to location (GPS device), or time (Real time clock). There are a huge number of sensors available, interfacing to them using serial, the I2C bus or the SPI bus are all straightforward. Alternatively, sensors which output a voltage can be read using an ADC, and ones which output a frequency can be read directly.

RS232 diagnosis

The problem:

RS232 communications can be difficult to debug, partly because of confusion over whether the Transmit and Receive signals need to be crossed or straight, and also because unless the baud rate is right, a terminal will probably display nothing.

Create a device to sample a bidirectional serial signal, deduce the baud rates from the pulse widths observed within the signal, decode the traffic and display it on a 2 line LCD, for example top line transmit, bottom line receive. Use two PCs and two of the USB to serial devices to set up a serial link to test. The ATMEGA644P device you used in the workbooks has two serial interfaces, allowing you to use its two receive serial pins to snoop the transmit and receive lines between the PCs.

Where is the hot spot in the PC ?

You have already used the MCP9700A temperature sensor in workbook 2. Using this device it is possible to make a very repeatable and reasonably accurate temperature sensor. Between 4 and 6 sensors need to be deployed in a PC to find where the heat flows are within the case, and which parts get hot in use. The ATTINY45 microcontroller has all the functionality you will need for the temperature sensor part, and as it is in an 8 pin package, it would need less soldering.

Combine several microcontrollers to find the hot spot in a PC. One might be a master, and communicate with several others over a shared serial line, or they might each have a simple SPI type interface back to the master. The sensors give very repeatable results, but will need initial calibration.

Pet tracker

Use a GPS receiver like this one: <http://www.coolcomponents.co.uk/catalog/em406a-unit-p-38.html> (70mA @ 5V so not a low power device!) or this one <http://www.coolcomponents.co.uk/catalog/ultimate-breakout-channel-updates-p-1109.html> (20mA @ 3.3V) to make a pet tracker.

If your pet is small or sleeps a lot, or both (cat) then putting the unit into sleep mode is going to be necessary to get any life out of the batteries. Some GPS receivers have logging built in, but often you can't change the parameters, so controlling the logging will also help save battery power.

Movement detector

We have sensors which are very sensitive to movement. Make a data logger to detect human movement and log it to decide whether the wearer has an active lifestyle. See: http://www.cl.cam.ac.uk/teaching/1415/P31/project_docs/ms24.pdf Alternatively, an accelerometer-

Project Ideas

ter will mean more work to do the interfacing but will give a much better measurement of movement.

Consider whether it is more efficient to log start/stop times, or active/inactive every second. Use sleep mode to extend the battery life. How long can the device work from a miniature coin cell, or even a large capacitor. Consider how the data might be stored and later offloaded, and how to tell the wearer whether they are being fit or a couch potato. The project PCB is probably ideal for this logger.

Radio link

Use 2 microcontrollers and a radio transmit/receiver pair such as the ER900TS/ER900RS to make a radio link, which would work as a serial extender. Using a real-time clock or the timers in the microcontroller, or more probably both, synchronise the transmit and receive so that a batch of data is sent every minute, and sleep the rest of the time to save power. See: <http://www.cl.cam.ac.uk/teaching/1415/P31/docs/ER900TS.pdf>

Evaluate the tradeoff between latency and power consumption by altering the time between bursts of data, and also consider the cost in power of switching the radios between active and sleep modes.

Radio Received Signal Strength Indicator (RSSI)

Use a radio receive chip such as the ER900RS which has an RSSI analogue output to make a received signal strength meter, perhaps using an LED array to display the strength. You will probably also need to program another microcontroller to send a signal every few seconds so that you have a signal to measure. Make both devices battery powered for maximum flexibility, and switch off the LEDs between readings to save power. See: <http://www.cl.cam.ac.uk/teaching/1415/P31/docs/ER900TS.pdf>

.wav file player

Use a 16 bit Digital to Analogue converter (DAC) on the SPI bus, and the project PCB to create a device which can play .wav files stored on a micro SD card. Initially use 8 bit data and a low sample rate, which will mean you will need to create or convert .wav files using a utility like audacity or ffmpeg. Eventually you might reach 44.1kbits/second and 16 bit data width for CD quality mono, but this will push the microcontroller towards its throughput limit. Reading 512 bytes from the SD card takes about 3mS, so you will need to buffer the data and send it to the DAC under interrupt control. You will almost certainly need to use the SPI mode of the USART in order to get a high enough throughput to the DAC. This project is challenging, as you will need to control two processes (read data, send to DAC) both of which cannot accept undue delays.

Baby pacifier

Babies can be soothed to sleep by a lullaby. When they wake again, they tend to either cry or move about, either of which could be used to play the lullaby again, and maybe the baby will go back to sleep.

Use a Digital to Analogue converter (DAC) and the project PCB to create a device which will play a short .wav file stored in EEPROM or FLASH ROM in a repetitive loop for a few minutes. Use a vibration sensor such as this one http://www.cl.cam.ac.uk/teaching/1415/P31/project_docs/ms24.pdf and a microphone input to retrigger playing the .wav file. In this case, as the file will be held in the microcontroller itself, there is no delay when reading, and no need to buffer the file. This is also challenging as it is in effect the .wav player (though reading from flash, which is much quicker than from an SD card) with extra bits.

Project Ideas

Telltale	<p>Create a 'telltale' - a device whose purpose is to show whether a door has been opened since the last time the device was checked. Use a microswitch or tilt switch to detect door opening, a reed switch or hall effect switch to interrogate the device, and two leds to show the state. A green led indicates that the door switch hasn't operated since the last check, a red one shows that it has. Alternatively use an accelerometer to detect if a package has been roughly handled in transit, again with a reed or hall effect switch to interrogate/reset the device.</p> <p>You will need to use sleep mode to make battery life as long as possible. You might want to check the different power consumptions of the ATTINY45, ATMEGA168A and ATMEGA644P for minimum power in sleep mode. A battery life of several years should be possible using 2 or 3 AA cells, or you might use a Lithium coin cell.</p>
Security / knock pattern sensor	<p>Some of the roadside cabinets for telecommunications are fitted with shock sensors to detect attempts at forcible entry. To avoid the alarm going off during authorized entry, the engineer might be required to enter a characteristic sequence of taps to the cabinet to disable the alarm. This has been done in the past with a shock sensor such as: http://www.cl.cam.ac.uk/teaching/1415/P31/docs/MS24.pdf, but this sensor is very hard to interpret, and using an accelerometer instead would allow for a more realistic tap code.</p>
Digital safe	<p>Create the electronics for a digital safe. The lock mechanism would be driven by a servo, and you would need to implement a 4 or 6 digit key code to gain entry. These devices are battery powered, so it is important to sleep for nearly all the time, and to energise the servo for only a very short period. With such a safe, you enter the code, the door unlocks, then stays unlocked until you close the door, as detected by a microswitch at which point the mechanism is relocked by the servo. What lifetime of typical usage would you get from say 3xAA cells.</p>
Digital compass	<p>Using the HMC6352 chip which has an I2C interface, create a digital compass. You will need to devise a suitable display for the user to show them their heading. Then, make a logger which records the heading every few seconds, and which can calculate a very rough bearing back to the starting point, and display it to the user.</p>
LED display	<p>We have a PCB which has an array of 20x28 LEDs and electronics to drive them from an ATMEGA644P microcontroller. Two of these can be stacked horizontally to give 40x28, and stacked vertically to any depth. The microcontroller can't drive the 560 LEDs directly, but can drive the 28 rows in rapid succession, relying on the persistence of vision effect.</p> <p>Now, what can you do with 560 LEDs ? You will need to write a driver for the LEDs on the ATMEGA644P, and write an application which uses the display on the same microcontroller (very hard) or a second microcontroller, then get the two microcontrollers to communicate.</p> <p>The ATMEGA644P has 2 serial ports, which makes daisy chaining a possibility, too.</p>
Phone LCD display	<p>We have display PCBs taken from a Nokia phone, of 48x84 pixels. The displays are very small, but they are easy to interface to as the pixels are presented as an array of bytes accessible using the SPI bus.</p>

Project Ideas

	<p>48x84 pixels is rather limiting, but for something like a small game they should be enough. Data transfer via the SPI bus should be quick, in fact the limiting factor for the display may be the LCD update rate.</p>
Precise rotation speed using a stepper motor	<p>Stepper motors have the property that up to some maximum frequency they will faithfully step once per incoming pulse. This makes them ideal to drive a turntable which needs to rotate at one of a number of precise speeds, for example for playing vinyl records, for which the standard speeds are 33rpm, 45rpm, 78rpm.</p> <p>The interfacing to the stepper motor is a little tricky but then you should find it easy to make it rotate at a very precise speed once this is done. Next check the limits of speed. Lastly you might try more precise control when energising the coils to try to extend the range of speeds available.</p>
Precise rotation speed for a DC motor	<p>Ironless-core DC motors run very smoothly, and start at a low applied voltage. Accurate control of the speed of a DC motor is difficult as it depends on the load, plus there is a delay between adjusting the supply voltage and the motor reaching the required speed.</p> <p>A photodetector on a notched wheel should allow you to measure the speed. You can then try either a pulse width modulation scheme or a variable voltage supply to control the motor with the objective of getting it to run at a precise speed.</p> <p>For this project the microcontroller part is straightforward, but the mechanical and feedback parts are quite hard to do well.</p>
Rotating display	<p>Create a display using an array of LEDs, perhaps 10 or so which are spun around an axis.</p> <p>To avoid the difficult problem of the wiring to the LEDs needing to go through commutating rings, fit the microcontroller which will output the message to the moving part, complete with its battery. It will need to use a photodetector or similar to measure the rotation speed. Rotation of the mechanism would be through a small DC motor driven from a power supply, which should give a consistent rotation speed for the microcontroller to work with.</p> <p>This project will require some mechanical work, and this may become time consuming. We have a laser cutter available to make parts from acrylic, and we can buy gears, axles etc as required.</p>
Useless machine	<p>Search for 'useless machine' on youtube. Make one with two servos, so that the delay between the flap opening and the switch operating is pseudo random. Next try to make the mechanism as interesting as possible, by varying the speed of operation of one or both servos. Finally, use sleep mode to cut the power consumption. The aim is a two year battery life using three AA batteries.</p>
Remote power control	<p>The Department of Geography and the Computer Lab have jointly developed a spectrometer to monitor volcanic activity. The instruments are battery powered and are deployed in locations which are difficult to visit, sometimes requiring a helicopter. The instrument sends its data back by serial, but it sometimes needs a hard reset, by interrupting the power for 2 seconds.</p>

Project Ideas

What is needed is a low power device which can monitor the serial traffic for the string 'REBOOT' and interrupt the downstream power. That part is easy. The harder task is to make it very robust, low power, and more reliable than the instrument it is controlling, using a wake on serial, plus a watchdog to get it out of any loops even if the serial data behaves in unexpected ways.