The PumpSpark Fountain Development Kit

Paul H. Dietz [1]
[1] Microsoft Research
One Microsoft Way
Redmond, Washington, USA
paul.dietz@microsoft.com

Gabriel Reyes [1,2]
[2] College of Computing
Georgia Institute of Technology
Atlanta, Georgia, USA
greyes@gatech.edu

David Kim [3]
[3] Microsoft Research
21 Station Road
Cambridge, UK
davidkim@microsoft.com

ABSTRACT

The PumpSpark Fountain Development Kit includes a controller, eight miniature water pumps, and various accessories to allow rapid prototyping of fluidic user interfaces. The controller provides both USB and logic-level serial interfaces, yielding fast (~100ms), high-resolution (8-bit) control of water streams up to about 1 meter high. Numerous example applications built using the PumpSpark kit are presented. The kit has been the subject of a student contest with over 100 students, demonstrating its utility in rapid prototyping of fluidic systems.

Author Keywords

FUI; fluidic user interfaces; fluid; water-based interfaces; PumpSpark; pump controller; prototyping; student contest.

ACM Classification Keywords

H.5.2 [User Interfaces]: Prototyping

INTRODUCTION

Liquid water is a remarkable substance. It is infinitely deformable, but largely incompressible. Due to its polar nature, water naturally attracts itself, minimizing surface energy. This also allows it to be attracted to electric charges. Although inherently non-conductive, most water contains impurities that allow it to carry electricity. It is transparent, allowing it to serve as a light guide. Water has a high heat capacity, and is thermally conductive. It carries sound very well. For the purposes of experimentation, it is convenient that it is also inexpensive, widely available, non-toxic, and easy to pump.

In addition to these physical properties, water is also aesthetically pleasing. Waterfront property has always been some of the most desirable real estate. City parks often feature fountains and pools. Just the sound of flowing water is so appealing that numerous manufacturers offer devices that play recordings of crashing waves and babbling brooks to help people sleep. The sheer joy of a child running through a sprinkler on a hot day is unmistakable.

DIS '14, June 21 - 25 2014, Vancouver, BC, Canada.

http://dx.doi.org/10.1145/2598510.2598599

Figure 1: The PumpSpark Fountain Development Kit

Considering all this, it is perhaps surprising that user interfaces do not more frequently make use of water. One reason may be that this requires a curious mix of skills. It is the rare individual who has expertise in pumps, plumbing, electronics, firmware and application development. Another fundamental reason is that contact with liquids leaves the user's hands or clothes wet, which may be inconvenient in certain situations.

The goal of this work is to create a simple, inexpensive, multi-pump system that allows researchers to begin to explore fluidic user interfaces (FUIs). The PumpSpark Fountain Development Kit (Figure 1) includes an 8-channel, brushless pump controller, 8 miniature pumps and a variety of accessories. The controller presents both USB and logic-level serial interfaces, and uses a protocol identical to standard hobby servo controllers. This allows it to work with existing, off-the-shelf, control software. The system is both precise and fast, with 8-bit resolution and step response times on the order of 100 milliseconds.

We present a number of examples of FUIs that can be created using PumpSpark. These range from simple displays to complex, interactive systems. Lastly, we report on a student contest that featured the PumpSpark kit, demonstrating its ease of use and versatility.

RELATED WORK

The PumpSpark Fountain Development Kit is a prototyping tool that allows non-experts to quickly create complex systems. We were inspired by the success of other hardware development systems such as Arduino [1], Gadgeteer [15], and Parallax's Basic Stamp [4]. These have empowered many to focus on the larger system design, rather than implementation details. Currently, those that wish to explore fluidic interfaces have no option but to immerse themselves in the tasks of selecting pumps, plumbing accessories and appropriate control systems. Many will be disappointed when they realize that pumps with internally controlled brushless motors are not compatible with standard motor controllers. We are in the process of arranging to make the PumpSpark kit commercially available to enable others to innovate in the creation of fluidic user interfaces.

While it is not the intent of this work to provide a detailed review of fluidic interfaces, it is instructive to consider some examples. Arguably, the most famous water-display is that at the Bellagio Hotel in Las Vegas, Nevada [5]. This 8-acre fountain was created by Wet Design, and features highly choreographed water movements, sound and light. Disney's World of Color show at Disney California Adventure [34], another immensely popular water-based show, adds fire, projection, and a variety of other effects.

Fluids are also often used to create ephemeral user interfaces [10]. These are interfaces that may last only a moment, such as text made from falling water, rising bubbles, or a momentary soap bubble. Examples include Space Printer [19], Bit.Fall [6], the Information Percolator [16], and Soap Bubble Interface [28].

Explorations have also taken advantage of the properties of water for indoor and outdoor gaming. Water Games [24] encouraged children to play together. In order to actuate a fountain, the children had to hold hands in a circle, and then rotate around the fountain. Water Ball Z [18] allowed kids and young adults to 'fight' each other in a virtual world using water streams to provide physical feedback without a wearable device.

Casual Profanity [8] has shown a number of fluid sculptures. These use carefully woven, clear tubing, with segments of liquid pumped through. The effect is that of segments rapidly chasing as if through a maze. Submerging Technologies [9] consisted of three interactive water sculptures, each of which fundamentally used the water itself as the sensing element. The Tantalus fountain used water as a capacitive proximity detector to withdraw from an approaching hand. The AquaHarp and TouchPond detected frustrated total internal reflection in a water waveguide, and reacted with sound and light respectively. A later project, Touché [27], further explored the use of capacitive sensing in water.

WaterTouch [20] also uses a water waveguide, but adds depth information by using a camera to see the depth of finger penetration into the water. AquaTop [29] turned water into an interactive surface display where users can interact with information projected on the water without using any electronic devices. A number of displays making use of water as a diffuser have also been proposed for entertainment and virtual reality [2, 3, 11, 12, 13, 30].

Finally, Mann is well known for his water-based, musical instruments. These include the Hydraulophone [22] and the Water Hammer Piano [21]. In these instruments, interaction with the water creates the resonators, that in turn produce the tones.

THE PUMPSPARK FOUNTAIN DEVELOPMENT KIT

The PumpSpark Fountain Development Kit includes the pump controller, a 13V/4A power supply, 8 pumps, and a variety of accessories.

The Pumps

Pumps are used as the primary water control mechanism. We also considered a valve-based design. Valves can be fast, and inexpensive. However, they require a source of pressurized water to function. While this could be provided by the public water supply, that would preclude recirculation, and a drain would be required. An alternative would be to create a local, pressurized, reserve using a pump, an accumulator, and a control system. But this adds considerable complexity and cost. In addition to these concerns, inexpensive solenoid valves are on/off. We were unable to find inexpensive, submersible, fast, proportional valves.

The DC30A-1230 [32] pump was chosen for the kit. It is extremely low cost (around US$10 in single unit quantities), and quite small (5cm x 5cm x 5cm including the ports and mounting bracket). The performance characteristics of this pump are shown in Figure 2. The DC30A-1230 can deliver a maximum flow rate up to 4L/min. The maximum head is 3m, which corresponds to a pressure of 4.35psi. It should be understood that these represent two different points on the operating curve, and that the available flow rate depends on the pressure. With an appropriate restricting nozzle, these pumps can easily shoot a significant stream of water a meter into the air.

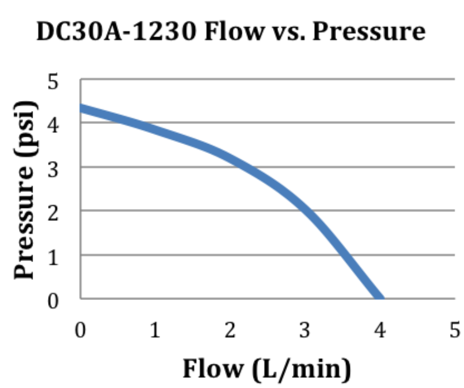


Figure 2: Performance of the DC30A-1230 brushless water pumps running at 12V.

The DC30A-1230 pump uses a brushless DC motor, and includes an internal controller to handle the commutation. Because of the design of this internal controller, the pump cannot be directly controlled via pulse width modulation - directly switching the supply on and off confuses the internal circuitry. Fortunately, the pump output is well controlled by the applied voltage. The pump requires a minimum of about 4.5V to turn on, and is rated for continuous operation at up to 12V. However, once the pump is running, the voltage can be decreased to ˜2.0V and continue to operate.

These pumps are not self-priming, which means fluid must be supplied to the inlet port. Conveniently, these pumps are fully submersible, so they can simply be placed underwater to provide the intake.

The PumpSpark Controller

The PumpSpark controller (Figure 3) was specifically designed to power and control eight DC30A-1230 pumps simultaneously. It provides both USB and a logic-level serial interface. It essentially serves as an 8-channel, programmable power supply. Full details on the design of the controller are given in the Appendix.

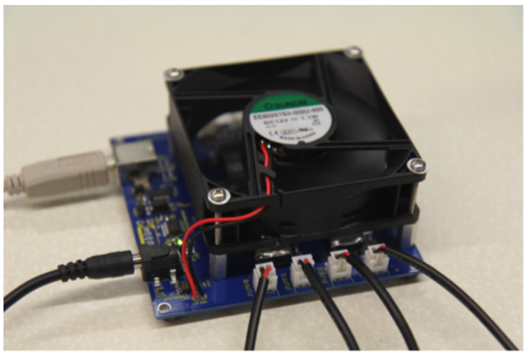


Figure 3: The PumpSpark 8-channel pump controller.

Accessories

Various accessories are included with the PumpSpark Development Kit. These make it simple to quickly prototype fountains and water-based interfaces: 3M 4952 VHB double-sided foam tape, tubing, T-connectors (for splitting and combining flows), nozzles, mesh screen (prevents splashing), and check-valves (prevents backflow).

In addition to the items provided with the kit, we recommend a number of optional accessories. Recently, inexpensive (˜$50) waterproof servos have appeared on the market [17]. These are ideal for creating interactive water systems that incorporate physical motion. The Hitec HS-5646WP provides over 10kg-cm of torque and speeds up to 300 degrees/sec in a standard size servo. These can easily be stacked to make high performance pan/tilt mechanisms for directing nozzles.

One non-optional item is a container for the fountain. We recommended plastic under-bed boxes. These were not included with the kit because the cost of shipping such a large item is prohibitive.

FUI EXAMPLES

To demonstrate the effectiveness of the PumpSpark Fountain Development Kit, we have created a number of example Fluidic User Interfaces. These show the diversity of systems that can be quickly created.

Water Show

The Bellagio Hotel in Las Vegas [5] features a superb example of a modern, computer-controlled, fountain. It uses digital control data to create a dramatic display. Similarly, we can use PumpSpark and Visual Show Automation [33] to quickly create a tabletop fountain (Figure 4a) that puts on a pre-scripted show, without the need to write any code.

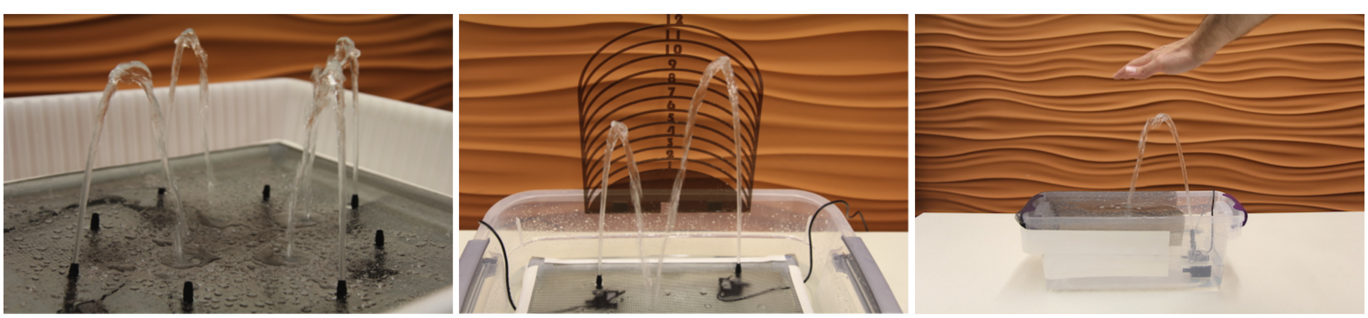
Using Visual Show Automation (VSA), one can load an audio or video file. While this is played back, the pumps can be controlled in real-time using a joystick. This is recorded to create a control track. Multiple control tracks can be created in this fashion. There are also tools for editing. Novices using this technique can create complete shows in minutes.

Since VSA is an animatronics control package, it is straightforward to include motion, lights and other effects in a show using commercial systems. For even greater sophistication, Console is an Active X control for VSA that allows it to be directed programmatically. For example, sensors could be used to trigger different shows; arriving email could trigger shows based on the sender.

Clock

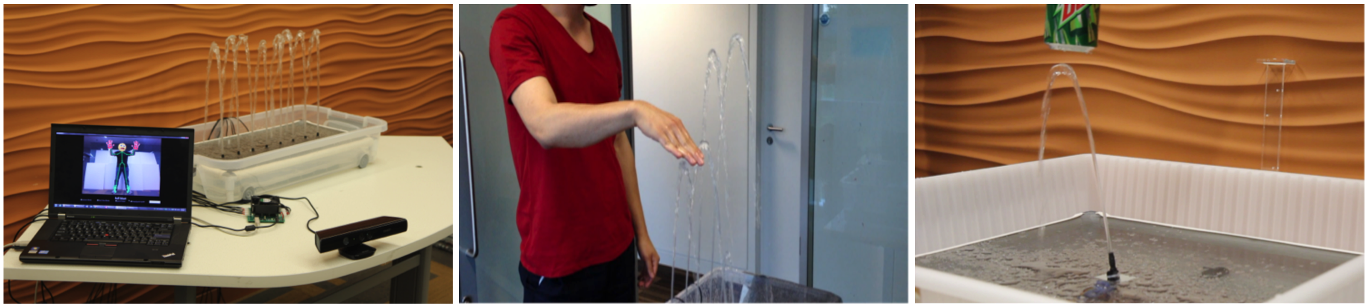
There have been numerous examples of fountains that display time. The Fountain Clock [14] uses directed water streams falling on a dial. Lit water streams act as a light guide, illuminating the proper numbers. The famous fountain clock at Osaka station created by Koei [19] shows the current time as a series of characters formed by falling water.

Our fountain clock uses PumpSpark and an Arduino (Figure 4b). Two pumps stream upwards - the one on the left for hours, and the one on the right for minutes. The height of the water indicates the time.



(a) Fountain show (b) Water clock set to 5:50pm (c) Frustrating fountain

Figure 4: PumpSpark FUI Examples.



(a) Kinect skeletal tracking (b) Sculpting water using computer vision (c) Optically tracking and shooting targets

Figure 5: PumpSpark FUI Examples.

Frustrating Fountain

A fun, interactive FUI is the Frustrating Fountain. This gag device (Figure 4c) appears to be an automatic drinking fountain. It turns on automatically when the user approaches. But as he or she dips down for a drink, the stream recedes, always staying just out of reach. The effect is achieved using an optical range finder connected to an Arduino controlling the PumpSpark controller.

Skeletal Tracking & Gesture Recognition

Kinect's skeletal tracking provides detailed information about the position of twenty joints of the user's body in the camera's field of view. We implement tracking of the user's hands, hip and shoulder joints to control the height of the water stream from 8 pumps (Figure 5a). Three static positions are detected with hands above the shoulders, hands below the shoulders and above the hips, and hands below the hips. Each position allows the user to raise and lower their hands (and corresponding water stream) as if conducting an orchestra. Gesture recognition using the Kinect SDK also enables detection of other hand interactions to control the water pumps. A pyramid position with both arms together in a praying stance creates a triangle shape fountain, swipes to either side of the body create instant bursts of water traveling in the direction of the swipe, both arms extended outward turns the pumps off, and a hand wave activates a different flow pattern every time with a random pump value generator.

Direct-Touch Water Sculpting

Water is an inherently tangible medium, but when grasped, it quickly slips through one's fingers. In this example, we create the illusion of permanence - allowing one to shape the water. We use computer vision techniques to track the position of the user's hands, sculpting the water fountain via direct touch.

In this setup (Figure 5b), a Kinect depth camera is pointed towards the fountain interaction area, and a bounding box is manually fitted around each stream. Any depth data outside these boxes is discarded. Connected components labeling is run on the remaining depth pixels. The vertical component of each centroid is then mapped to control the height of the corresponding stream.

The depth camera cannot reliably detect the water streams, so a manual two-point calibration process is used to match the height of the water to the position of the hand. Automatic fitting of the bounding boxes and mapping pump strength to hand height may be achieved by utilizing a non-transparent opaque liquid that can be robustly sensed with the depth sensor or an RGB color camera.

Our setup currently allows users to press down or lift up individual water streams using multiple hands and also to instantly set water streams to the desired height. Other compelling uses include controlling sliders of a sound or light mixer (e.g. for DJs/VJs), and directly shaping and programming water fountain shows. Please refer to our video figure (attached) to see a live demonstration.

Motion Tracking Using Waterproof Servos

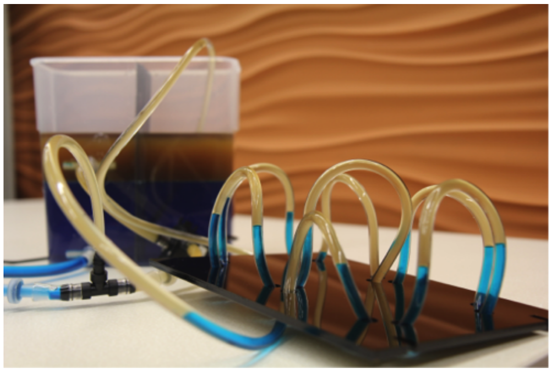
The previous examples all used fixed position nozzles. However, there are lots of interesting applications where we would like to dynamically direct a water stream in different directions. Waterproof servos make this easy. Using a simple pan/tilt mechanism, we can direct a nozzle in any direction. If the pump speed is also controlled, the water can be directed to any point in a 3-dimensional space.

In our demonstration system (Figure 5c), we attach a nozzle to a Hitec HS-5646WP [17] waterproof servo, and connect it to a pump via a short length of very flexible tubing. The servo is connected to the PC via a Pololu Micro Maestro Servo Controller [25]. A standard webcam is used to see colored objects, which the system tracks, and squirts.

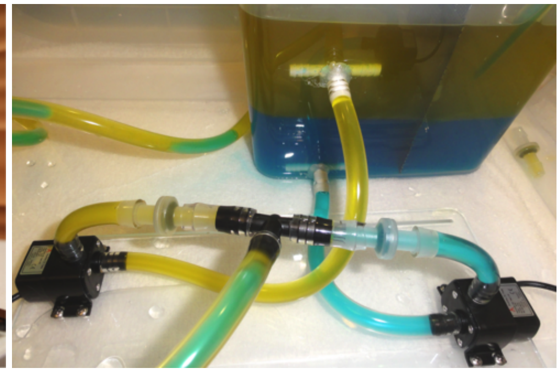
In keeping with our rapid prototyping approach, this demonstration was driven by RoboRealm [26] - a commercial software package for prototyping robotic vision systems. Because RoboRealm already understands the Serial Servo Controller protocol used by PumpSpark, no special drivers were required.

7-Segment Numerical Display

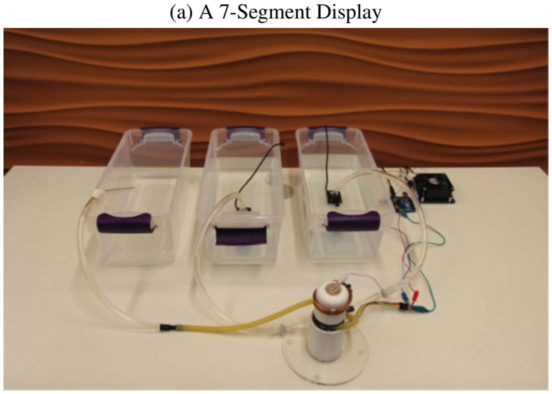
Recently, artists have used clear tubing to create stunning visual displays [8]. The basic idea is that tubing can be thought of as a visual shift register. By loading in successive segments of differently colored, immiscible fluid, patterns shoot through the tubing. By carefully controlling the amount of fluid in each segment, any visual pattern can be serially shifted into the tube.



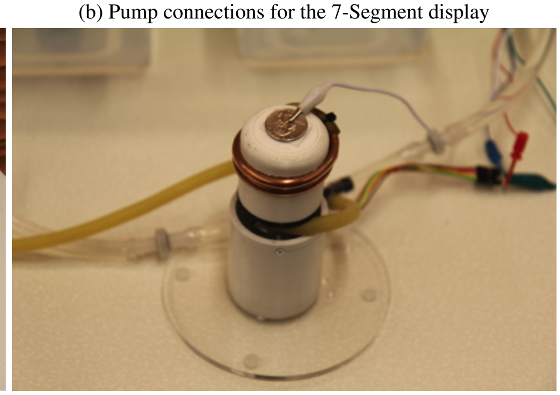
(a) A 7-Segment Display



(b) Pump connections for the 7-Segment display



(c) Reservoirs for thermal feedback knob



(d) Close-up of near-instantaneous thermal feedback knob

Figure 6: PumpSpark FUI Examples.

We created a fluidic 7-segment display modeled on the ubiquitous LED numeric displays (Figure 6a). A single length of tubing was weaved through a frame to create the segments.

Two pumps with check valves (Figure 6b) are used to inject two differently colored fluids into the tube in a controlled fashion. The fluids were water (with blue food coloring) and various cooking oils (olive, sesame, and sunflower oil). These oils were chosen because they are immiscible with water, and quickly separate into layers when mixed together. The outlets are positioned in the reservoir to pump the two fluids independently. This allowed for a closed, single tank system. A grating placed in the tank between the turbulent return side and the calm outlet side improved the separation.

Visual Show Automation was used to create the appropriate pump sequences to generate each digit. Because the oils are much more viscous than water, they pump much more slowly. The rate at which the pumps could inject fluid into the long tube depended on the current amount of oil in the tube - the more oil there was, the harder it was to pump. As such, we decided to flush the system between each digit to prevent dependence on the previous display. In a future version, we hope to add visual feedback, allowing the system to pump until a certain volume has passed, independent of the oil load.

Knob with Thermal Feedback

Consider the problem of setting the temperature for a bath, or for a room. Thermostats allow one to set a desired level. Typically, a visual display of some sort lets the user know what temperature has been set. This is necessary because heating and cooling systems typically take a long time to respond. Unfortunately, users often lack a sense of the specific temperature they desire, and go through a series of adjust-and-wait cycles before finding a comfortable level. Instant thermal feedback could alleviate this problem.

We created a knob (Figure 6c, 6d), which provides near-instantaneous, thermal feedback. Rather than heating and cooling the knob by traditional means (resistive heating, thermo-electric coolers, etc.), we exploit the high thermal capacity of water. The control knob surface is fitted with thin wall copper tubing. Two pumps are used to send water through this thermally conductive tubing - one to pump hot water, the other cold water. By controlling the relative proportion of each, the temperature of the knob can be rapidly changed in response to user input. The system is effective because only a small volume of nearby water needs to be kept at the high and low temperatures. A capacitive proximity sensor (attached to a coin placed on the knob) determines when the knob is in use, minimizing the pump run time.

THE CONTEST

The PumpSpark Fountain Development Kit was the subject of the 2013 ACM User Interface Software & Technology Symposium Student Innovation Contest [31]. Approximately 25 teams, typically of 4 students each, participated. The students built an amazing variety of water-based systems. Some used the pumps to provide power for mechanical motion (e.g. moving a ball, driving a boat, filling an elastic actuator, etc.). One used a pump to control the volume of water in an acoustic resonator to make music. Many teams created water-based displays that conveyed data in various forms. Others used the pumps to serve beverages in a controlled fashion. Contest judges were impressed at the outpouring of creativity.

Informal interviews with the contest participants confirmed the ease of use of the kit. Most reported that they achieved controlled pumping within minutes. None reported facing any significant obstacles in making the kit function. Both the USB and logic-level serial interfaces were used, without a clear preference for one over the other. In retrospect, having both helped open up the system to more users.

All of the teams used the kit as a starting point, adding a staggering array of additional hardware, including video displays, Kinect sensors, RFID, a laminar flow nozzle, projectors, audio systems, etc. We examined their additions to see if there were common items that should have been included with the kit. Only one item fit this description - a large container for their water reservoir.

CONCLUSION

We have presented the PumpSpark Fountain Development Kit. The kit allows for the rapid prototyping of interactive, fluidic user interfaces. We have demonstrated the utility of this kit by creating a number of sample systems, highlighting its compatibility with a variety of microcontroller platforms and integration with sensors and actuators. Finally, the kit has been tested with over 100 users in a student contest, and shown to inspire creative design.

ACKNOWLEDGEMENTS

We would like to thank Bruce Cleary and the Microsoft Research Hardware Lab for electronic assembly of the PumpSpark controller boards, and Phillip Dietz and Olivia Dietz for preparing kit components (screen, tape, tubing, connectors). Special thanks to the UIST 2013 conference organizers, and the many participants in the Student Innovation Contest. We also gratefully acknowledge the donation of 24 Microsoft Surface tablet computers (the contest prizes) arranged by Panos Panay and the Surface team.

APPENDIX

The PumpSpark Controller

The goal of the controller (Figure 3) is to act like a programmable power supply. Figure A1 shows one channel of the output stage. The input is a 3.3V PWM signal from a PIC16F720 microcontroller. This is filtered by a simple RC circuit to provide a DC control voltage. This is amplified up to a 12V range via a TCA0372D power op-amp used in a non-inverting configuration. A 13V supply rail yields an output swing of 1-12V.

The linear power output stage produces a significant amount of waste heat. At a command voltage of *Vout*, the power dissipated in the amplifier is:

$$Power = (13V - Vout) \times \frac{Vout}{Rp}$$

Rp is the equivalent resistance of the pump (˜24ohms). Figure A2 shows the power dissipated as a function of the control voltage. Worst case, each amp must dissipate ˜1.8W. The TCA0372D has two amplifiers in a package, and if both are utilized, ˜3.6W must be dissipated worst case. In order to provide a modest temperature rise, a cooling fan is utilized.

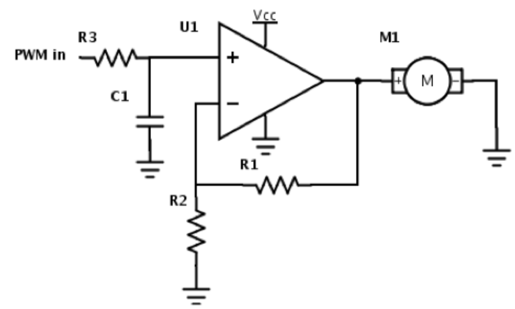


Figure A1: PumpSpark pump driver circuit.

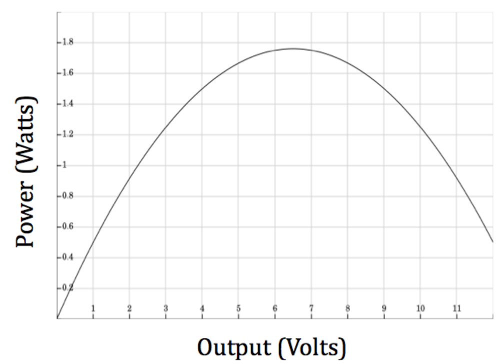


Figure A2: Power dissipation in the output stage as a function of commanded output voltage.

The PumpSpark controller features both USB and line-level serial interfaces. An FTDI FT231XS-U provides the USB input and produces a logic-level serial output, which idles high. The board's separate logic-level serial interface is brought in via an opto-coupler. This connection, shown in Figure A3, provides a logical AND of the two signals. There are a number of advantages to this serial interface design:

1. It provides electrical isolation, eliminating power supply sequencing issues and any unintended ground loops.
2. It allows both interfaces to be simultaneously connected, and used in a non-overlapping fashion.
3. The design works with 3.3V and 5V microcontrollers.
4. Both inverting and non-inverting serial signals can be used. For example, to connect an Arduino Uno serial output, *SIN+* is tied to +5V, and the serial connection comes in via *SIN-*.

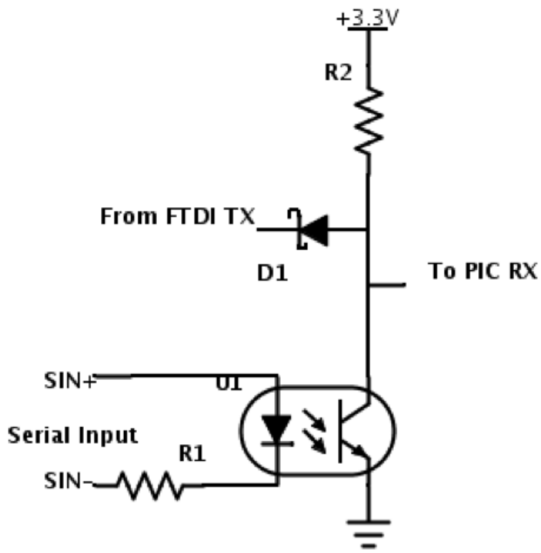


Figure A3: USB and line level interface circuit.

The Firmware

The purpose of the firmware is to read serial commands and produce 8 PWM outputs. While there are microcontrollers available with 8 hardware PWM peripherals on-chip, these would add significant expense to the controller. Instead, the signals are generated in software. A common technique for generating software PWM signals is to use a timer interrupt to periodically update the PWM outputs appropriately. Unfortunately, this method yields PWM signals with a rather long minimum bit time (the interrupt period). Variable interrupt latency can also lead to error if not properly compensated.

The DC30A-1230 pumps have a response time on the order of 100ms. Using a simple RC for smoothing, a PWM period of 1ms or less is desirable to minimize ripple. This is achieved by using a bit plane technique [7]. This produces signals which are on for the correct amount of time in each cycle, but which may switch on and off a number of times within the cycle. The output port is initially loaded with the LSB value for each signal, and then turned off as soon as possible. The port is then loaded with the next least significant bit, and left on for twice the minimum time. The next bit will be held for four times the minimum time. This pattern is continued up through the MSB. Generating the bit patterns may take a small amount of time during which the outputs are all off, making it impossible to achieve a 100% duty cycle. So long as this time is consistent, the result will simply mean that the duty cycle range is slightly smaller. Linearity is not compromised. This small gain error can be compensated by increasing the gain taken with the op-amp.

The bit plane approach allows the LSB to be a single instruction cycle. With a 4MHz instruction rate, an 8bit PWM signal can have an update rate greater than 10kHz. The MSB takes half the total time. This provides an ideal opportunity for the processor to service other needs without interruption from updating bit plane outputs. With 9600 baud serial data, characters arrive at a maximum rate of 960/sec. Thus, no more than one character can arrive while the system is off processing all of the bit planes other than the MSB.

A simple three-byte communications protocol is implemented:

Sync (0xff) : Pump (0x00-0x07) : Rate (0x00-0xfe)

This provides compatibility with the popular Scott Edwards Mini SSC II Serial Servo Controller [23]. There are a number of animation control packages that directly support this protocol, notably, Visual Show Automation from Brookshire Software [33]. This application allows pump control sequences to be created, edited and played back, along with synchronized audio or video. This software can also direct servo motor controllers, lighting controllers, etc., allowing users to easily create complex water, motion, light, audio and video sequences that can be easily triggered through software in response to events.

REFERENCES

1. Arduino. `http://www.arduino.cc/`, retrieved 1/19/2014.
2. Barnum, P., Narasimhan, S., and Kanade, T. A Projector-Camera System for Creating a Display with Water Drops. In *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on* (June 2009), 17–23.
3. Barnum, P., Narasimhan, S., and Kanade, T. A Multi-layered Display with Water Drops. In *ACM SIGGRAPH 2010 Papers*, SIGGRAPH '10, ACM (New York, NY, USA, 2010), 76:1–76:7.
4. Basic Stamp — Parallax Inc. `http://www.parallax.com/microcontrollers/basic-stamp`, retrieved 1/19/2014.
5. Bellagio — Las Vegas Hotels. `http://www.bellagio.com/`, retrieved 1/14/2014.
6. Bit.Fall — Science Gallery. `https://dublin.sciencegallery.com/surfacetension/bitfall/`, retrieved 1/19/2014.
7. Bit Plane. `http://en.wikipedia.org/wiki/Bit_plane`, retrieved 1/17/2014.
8. Casual Profanity. `http://casualprofanity.com/`, retrieved 1/19/2014.
9. Dietz, P. H., Han, J. Y., Westhues, J., Barnwell, J., and Yerazunis, W. Submerging Technologies. In *ACM SIGGRAPH 2006 Sketches*, SIGGRAPH '06, ACM (New York, NY, USA, 2006).
10. Döring, T., Sylvester, A., and Schmidt, A. A Design Space for Ephemeral User Interfaces. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction*, TEI '13, ACM (New York, NY, USA, 2013), 75–82.

11. Eitoku, S., Hashimoto, K., and Tanikawa, T. Controllable Water Particle Display. In *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, ACE '06, ACM (New York, NY, USA, 2006).
12. Eitoku, S., Nishimura, K., Tanikawa, T., and Hirose, M. Study on Design of Controllable Particle Display Using Water Drops Suitable for Light Environment. In *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, VRST '09, ACM (New York, NY, USA, 2009), 23–26.
13. Eitoku, S., Tanikawa, T., and Suzuki, Y. Display Composed of Water Drops for Filling Space with Materialized Virtual Three-dimensional Objects. In *Virtual Reality Conference, 2006* (March 2006), 159–166.
14. Fountain Clock. `http://www.instructables.com/id/Fountain-Clock/`, retrieved 1/19/2014.
15. Gadgeteer. `http://www.netmf.com/gadgeteer/`, retrieved 1/12/2014.
16. Heiner, J. M., Hudson, S. E., and Tanaka, K. The Information Percolator: Ambient Information Display in a Decorative Object. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*, UIST '99, ACM (New York, NY, USA, 1999), 141–148.
17. Waterproof Servos — HITEC RCD USA. `http://hitecrcd.com/products/servos/waterproof-servos-2`, retrieved 1/19/2014.
18. Hoste, L., and Signer, B. Water Ball Z: An Augmented Fighting Game Using Water As Tactile Feedback. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction*, TEI '14, ACM (New York, NY, USA, 2013), 173–176.
19. Koei Co., Ltd. `http://www.koeiaquatec.co.jp/`, retrieved 1/12/2014.
20. Mann, S., Janzen, R., and Huang, J. "WaterTouch": An Aquatic Interactive Multimedia Sensory Table Based on Total Internal Reflection in Water. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, ACM (New York, NY, USA, 2011), 925–928.
21. Mann, S., Janzen, R., Huang, J., Kelly, M., Ba, L. J., and Chen, A. User-interfaces Based on the Water-hammer Effect: Water-hammer Piano As an Interactive Percussion Surface. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '11, ACM (New York, NY, USA, 2011), 1–8.
22. Mann, S., Janzen, R., and Post, M. Hydraulophone Design Considerations: Absement, Displacement, and Velocity-sensitive Music Keyboard in Which Each Key is a Water Jet. In *Proceedings of the 14th Annual ACM International Conference on Multimedia*, MULTIMEDIA '06, ACM (New York, NY, USA, 2006), 519–528.
23. Mini SSC II Serial Servo Controller, User Manual — Scott Edwards Electronics, Inc. `http://www.seetron.com/docs/ssc2mnl.pdf`, retrieved 1/15/2014.
24. Parés, N., Durany, J., and Carreras, A. Massive Flux Design for an Interactive Water Installation: Water Games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, ACE '05, ACM (New York, NY, USA, 2005), 266–269.
25. Micro Maestro 6-Channel USB Servo Controller (Assembled) — Pololu. `http://www.pololu.com/catalog/product/1350`, retrieved 1/19/2014.
26. RoboRealm. `http://www.roborealm.com/`, retrieved 1/19/2014.
27. Sato, M., Poupyrev, I., and Harrison, C. Touché: Enhancing Touch Interaction on Humans, Screens, Liquids, and Everyday Objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, ACM (New York, NY, USA, 2012), 483–492.
28. Sylvester, A., Döring, T., and Schmidt, A. Liquids, Smoke, and Soap Bubbles: Reflections on Materials for Ephemeral User Interfaces. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '10, ACM (New York, NY, USA, 2010), 269–270.
29. Takahashi, Y., Matoba, Y., and Koike, H. Fluid Surface: Interactive Water Surface Display for Viewing Information in a Bathroom. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '12, ACM (New York, NY, USA, 2012), 311–314.
30. Tao, J., Geng, Z., and Fan, Q. A Digitized Water Display System Based on RS-422 Bus. In *Electrical and Control Engineering (ICECE), 2010 International Conference on* (June 2010), 39–43.
31. UIST 2013 Student Innovation Contest. `http://www.acm.org/uist/uist2013/contest.php`, retrieved 3/19/2014.
32. Vovyo Brushless DC Pump (DC30A). `http://www.dcbrushlesspump.com/Brushless-DC-Pump(DC30A)-p24.html`, retrieved 1/19/2014.
33. Visual Show Automation — Brookshire Software. `http://www.brookshiresoftware.com/vsa_overview.htm`, retrieved 1/19/2014.
34. World of Color — Wikipedia. `http://en.wikipedia.org/wiki/World_of_Color`, retrieved 1/19/2014.