Fab@Home: The Personal Desktop Fabricator Kit

Evan Malone, Hod Lipson
Mechanical & Aerospace Engineering
Cornell University, Ithaca NY USA

Solid freeform fabrication has the potential to revolutionize manufacturing, perhaps even to allow consumers to customize and manufacture goods cost-effectively in their own home. At present, the florescence of the technology is limited by a "chicken and egg" paradox. There is insufficient consumer demand and too narrow a range of applications for SFF systems to allow mass production to reduce their cost and complexity. At the same time, consumer applications for, demand for, and indeed awareness of SFF technology is limited by its high cost and complexity. We posit that just as the personal computer revolution was spurred by development of computer kits, getting SFF technology into the hands of hobbyists and hackers will simultaneously generate applications for, and improvements to it. To this end, we have developed the Fab@Home personal SFF kit, and are developing a user-group website to promote exchange of ideas and improvements. The designs and software for Fab@Home will be freely distributed, and constructing a first generation kit should cost roughly $2000. The kit design and a working unit will be presented, along with our experience deploying a unit for public use.
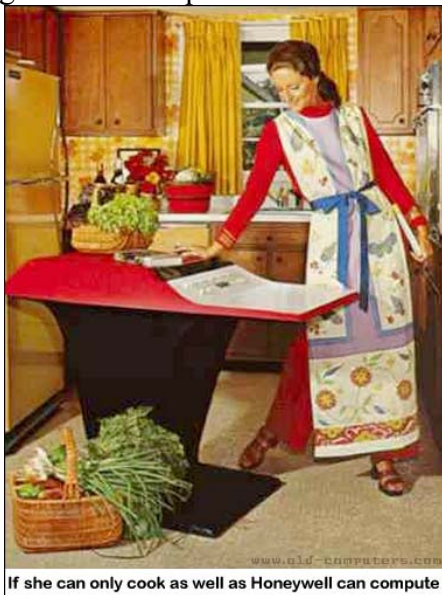
Introduction

Universal manufacturing embodied as today's freeform fabrication systems has – like universal computers – the potential to transform human society to a degree that few creations ever have. The ability to directly fabricate functional custom objects could transform the way we design, make, deliver and consume products. But not less importantly, rapid prototyping technology has the potential to redefine *the designer*. By eliminating many of the barriers of resource and skill that currently prevent ordinary inventors from realizing their own ideas, fabbers can "democratize innovation" [4,1,2]. Ubiquitous automated manufacturing can thus open the door to a new class of independent designers, a marketplace of printable blueprints, and a new economy of custom products. Just like the Internet and MP3's have freed musical talent from control of big labels, so can widespread RP divorce technological innovation from the control of big corporations.

Despite the formidable potential of rapid prototyping technology, its acceptance over the last two decades has remained disappointingly slow [8]. At present SFF systems remain very expensive and complex, focused on production of mechanical parts, and used primarily by corporate engineers, designers, and architects for prototyping and visualization. These factors are linked in a vicious cycle which slows the development of the technology: Niche applications imply a small demand for machines, while small demand for machines keeps the machines costly and complex, limiting them to niche applications. Alternatively, if one could provide either a large market for SFF machines and products or a simple and cheap SFF machine with which end users could invent products and applications, then this same feedback coupling could instead drive a rapid expansion in SFF technology and applications.

Learning from the history of the computer revolution

In attempt to break the vicious cycle of expensive equipment and niche applications, there are many lessons to be learned from the rise and growth of an equivalently universal technology: The computer. The parallels between universal computation technology and universal manufacturing technologies are astounding. Though the universal computer in its modern architecture was realized in the 40's [5], two decades passed before it reached any significant commercial acceptance. Early inventors themselves could not foresee its huge potential, notoriously anticipating a need "for as many as five or six machines in the US" [5]. The early commercial mainframes of the '60s were used mostly for niche applications such as payroll and military calculations. Like today's rapid prototyping machines, early mainframes cost tens and hundreds of thousands of dollars, required hours to complete a single job, had the size of a large refrigerator and required trained technicians to operate and maintain.
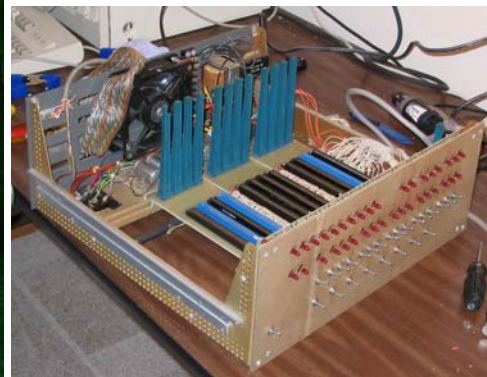

(a)

(b)

(c)

Figure 1. Early home computers trying to break into home market. (a) The Honeywell Kitchen Computer cost $7000 and targeted the cooking as the 'killer app'; (b, c) The general purpose Altair 8800, credited as starting the home computer revolution, came as a $400 kit [6]

Though it was clear to early manufacturers that the home market offered great potential, it was unclear how to successfully capture that market. Early attempts of the computer industry to break into the home market through niche 'killer apps' failed miserably: Some brands targeted niche domains such as Honeywell's "kitchen Computer" geared towards recipes (Figure 1a). Its high cost and narrow application prevented it from success. Though several other home computers came out in the early '70s [7], the MITS Altair 8800 (Figure 1b,c) is generally credit with sparking the home computer revolution. Designed and sold through Popular Electronics as a $400 kit, the Alltair 8800 broke the chicken-and-egg cycle: Hobbyists and experts could now afford to dabble with computers, develop and exchange software and numerous hardware accessory projects. Computer history had entered its exponential growth era.

Based on this history, it seems reasonable to imagine a low-cost multi-material SFF system in one's home, which could produce objects or even complete integrated devices from designs which are shared or purchased online [2]. Should such systems become as available as personal computers or printers are today, the invention and personalization of small devices could become as ubiquitous as music sharing is today. MIT's FabLab project [1] provides ample evidence that providing people with automated fabrication tools serves as an innovation catalyst; ordinary folk, with seemingly no technical background quickly learn to exploit these tools to design and realize new inventions.

The Fab@Home project

Inspired by this history, and the approach employed by Bowyer toward the goal of self-replicating manufacturing systems [3], we are developing an open-source, low-cost, personal SFF system kit, which we call "Fab@Home". The aim of this project is to put SFF technology into the hands of those same curious, inventive, and entrepreneurial citizens. In addition, taking a page from more modern history, we are developing a "wiki" web site through which we hope to inspire users of Fab@Home to exchange their ideas for applications and their improvements to the hardware and software with us and each other. We have built three complete Fab@Home systems, and two of them have been delivered to users outside of Cornell University – one to the AMTS FabLab in Pretoria, South Africa, and one to a life-sciences laboratory in Rockefeller University, New York City. The Fab@Home design, and preliminary conclusions about these first deployments are presented below.

Design

Fab@Home (Figure 2) is a 3-axis Cartesian gantry positioning system driven by stepper motors attached to lead screws. Material deposition tools are modular, and the first tool we have designed is a syringe-based extrusion tool, which uses a linear stepper motor to control the syringe plunger position. The electronics and firmware of the current version of the system provide for up to 6 axes of bipolar stepper motor control at 24V, with 2 limit switches per axis of positioning plus an optional one limit switch for the remaining 3 axes. A microcontroller controls the positioning of the axes and is in bidirectional USB communications with the PC. An application running on the PC displays the real-time state of the machine numerically and graphically, and allows the user to manually

position the axes, import and perform basic modifications to STL geometry data, apply specific material properties to each STL, and to generate and execute tool paths.
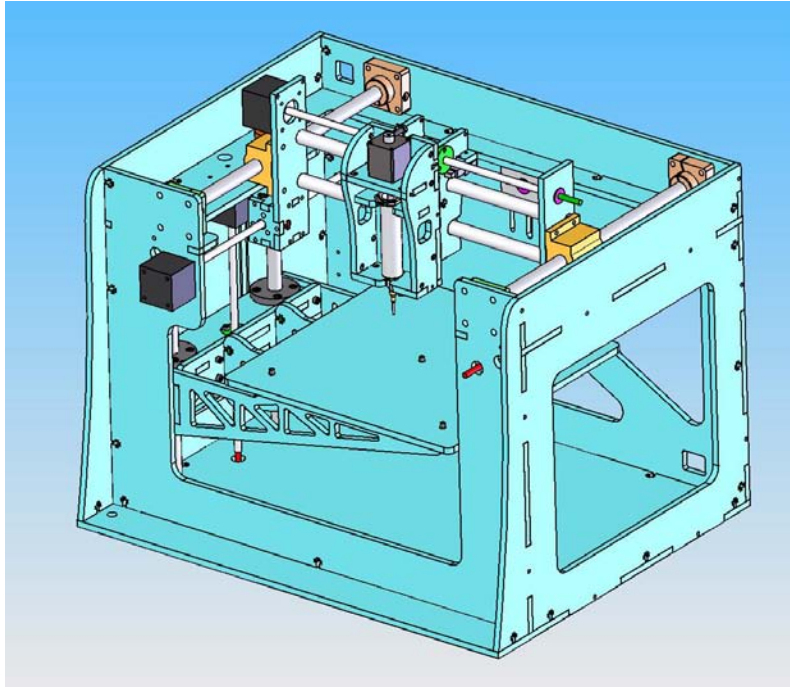

Figure 2. 3D CAD model of an assembled Fab@Home machine

One concern in developing our design is that there is probably a minimum threshold of quality required in new technology kit for hobbyists, below which the excitement of the new technology will be masked by the malfunctions, maintenance problems, and poor aesthetics. Users must have a sense of what the technology is capable of before they can grasp how to modify it and apply it to their own purposes. Our first design has therefore focused more on ease of use, reliability, and aesthetics than on minimizing cost.

We have tried to assume a modest availability of technical tools and skills for the envisioned end user. The person interested in our kit would need to have a laptop or PC with a USB port, and basic assembly skills and tools, including soldering and crimping of cables and connectors, but as much as possible we have tried to design the kit such that it can be fully assembled with standard components and tools such as Allen keys, screwdrivers, scissors, pliers, and soldering iron. The main assembly tasks are snapping together the acrylic structure, inserting nuts and screws and threaded inserts, bolting together the positioning system components and mounting them to the structure, making cables to connect the microcontroller to the amplifier boards, and the motors to the amplifier boards, mounting the electronic boards to the chassis, and bundling and routing of cables. The user is expected to have some patience as well: completely assembling a kit from parts to operation requires roughly 12 hours of labor. Currently, the parts cost is estimated to be $2000, excluding the cost of having acrylic parts laser cut, and not including shipping costs. Table I summarizes the kit hardware parts and costs.

Table I. Part types and cost breakdown

Category	Total # of Items / # of Diff. Types	Materials Cost
Motors	4/4	$556.70
Electronics	25/15	$488.06
Bearings/Transmission	53/18	$723.86
Fasteners/Hardware	360/23	$39.63
Acrylic	5/2	$228.81
Control Software	3/3	$0.00
Totals	**447/62**	**$2037.06**

Structure

The structural components of our system are built from laser-cut acrylic sheet parts held together with snap-fit joinery and simple "T-nut" style screw/nut fastening, in which a square or hex nut is inserted into a slot in one part, and a screw threads in through a hole in a perpendicular part (Figure 3). The tight manufacturing tolerances achievable with laser cut acrylic enable us to produce good orthogonality and alignment in the machine base, increasing the ease of setup and reliability of the machine. Our first three units have been produced in-house with an Epilog Helix 35W laser engraver, cutting parts directly from SolidWorks drawing files (Figure 3), which are in turn generated from a SolidWorks 3D model of the complete system.
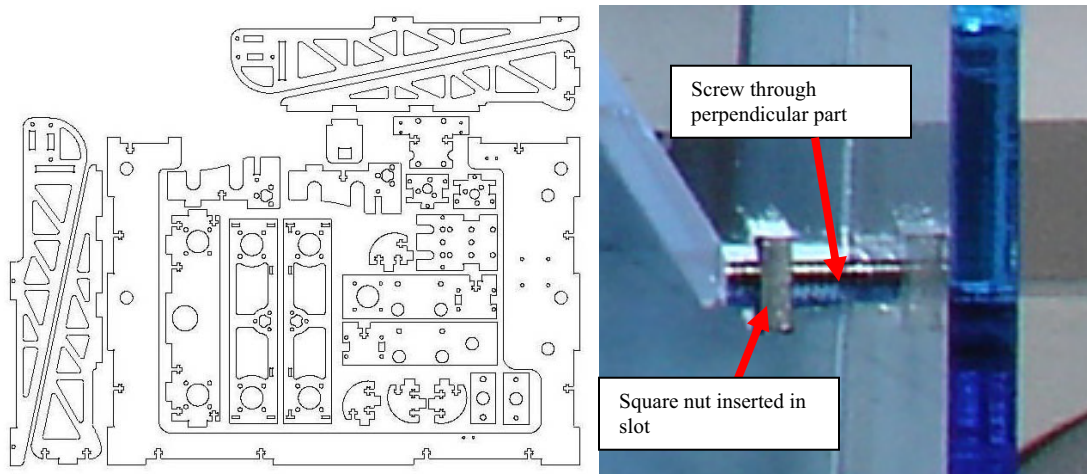

Figure 3. (left) Drawing of kit parts arranged for laser cutting from acrylic sheet stock; (right) "T-Nut" fastening of perpendicular acrylic sheet parts

There are 33 acrylic sheet parts in the chassis, plus an additional 7 acrylic parts for the syringe tool. Currently, six sheets of 18" X 24" X 0.236" cast acrylic are used to produce the acrylic parts. The packing of parts into these sheets is not optimal – only about 50% of the sheet surface area is converted to parts. More efficient packing is feasible.

Cutting the parts for an entire machine requires three hours on our 35W laser cutter. We are in the process of obtaining quotes for contract manufacturing of these parts to evaluate the additional cost associated with manufacturing them. An additional benefit to having the structural parts laser cut, is that there is a large installed base of laser cutters/engravers in the sign making, and trophy and gift engraving industries. The

custom nature of the work in these industries should make them amenable to kit builder's approaching them to have parts made. In addition, most laser cutters used in these industries do not require specialized CAM software – they operate from an application printer driver, so almost any image editing or vector drawing program can be used to design or modify designs for laser cutting and engraving. Thus, the Fab@Home kit hacker does not need to invest in 3D CAD software to make or publish new hardware designs, and the bitmap engraving possible with these laser cutters allows them to customize the appearance of their machine with images and text. It is our hope that we can negotiate some reduced cost for laser cutting services for the Fab@Home community, at least for a standard design, in exchange for advertising on the Fab@Home website and a large volume of customers.

Positioning

The linear motion components of the positioning system use off-the-shelf linear ball-bearing pillow blocks running on ½" diameter rails (McMaster-Carr, Inc.). The X and Y axes are in a gantry configuration with the deposition tool riding on the Y axis, which in turn rides on the X. The Z axis moves the build surface independently from the X and Y to minimize acceleration of parts as they are being fabricated. We have selected HSI Inc. linear stepper motors for our actuators because of their simple design, high resolution, and the semi-custom manufacturing focus of the company which permits specifying precisely the leadscrew and bearing journal dimensions required for our application, simplifying the overall design and assembly. For the X,Y, and Z axes we use NEMA size 14 bipolar motors with rotor-mounted lead screws. External polymer lead nuts are mounted to the axis carriages. In the case of the X axis, a timing belt and pulleys (Stock Drive Products, Inc.) are used to couple a slave leadscrew to the motor leadscrew to achieve symmetrical drive of the gantry. The force, maximum speed, and positioning resolution all depend upon the lead screw threading selected. We use the "B" series threading for the positioning axes, which allows for 15.8 μm travel per full step, a nominal top speed of 25 mm/s (1600 step/s), and a maximum thrust of 120 N.

Syringe Tool

The syringe tool structure (Figure 4) is also constructed of laser cut acrylic parts with snap fit joinery and T-nut fasteners. A linear stepper motor controls the position of the syringe piston. We employ as NEMA size 8 frame motor with a rotor mounted lead nut. The lead screw, which is not captive in the motor, has "7" series threading, 3.2 μm travel per full step, and the motor can achieve a top speed of 5.8 mm/s (1800 step/s), and a maximum thrust of 90 N. For the 10cc syringes we use, this amounts to a 1.1 cc/s maximum volume flow rate, and a maximum syringe pressure of 460 kPa (67 PSI). The current syringe tool has been designed to allow EFD Inc. 10cc disposable syringe barrels to snap in and out, and for the piston to be quickly attached and released from the motor leadscrew for quick changing of materials. A metal nut fits tightly inside of disposable syringe pistons, and one end of the motor leadscrew has threading to match the nut. When firmly threaded into the nut, the leadscrew is prevented from rotating with the motor rotor, and hence the rotor motion is converted to linear motion. Manually unscrewing the leadscrew from the nut allows exchanging syringes regardless of how full

without the need to move or remove the piston, facilitating fabrication of multiple-material objects, and conserving materials.
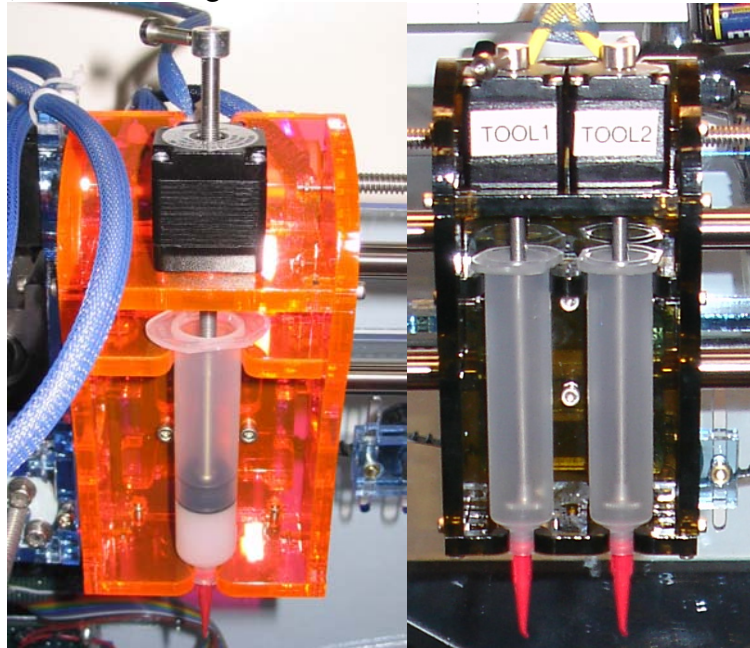

Figure 4. (left) The standard design, single syringe tool, driven by a linear stepper motor; (right) A two-syringe version for a life-sciences laboratory

We have also developed a dual syringe tool (Figure 4) which allows two materials to be loaded simultaneously and independently deposited. As mentioned before, tools are bolted to the positioning system, and tools are intended to be modular. We have selected a syringe deposition tool as the first design because of the broad range of materials useable with such tools, and for the intuitiveness of operation.

Electronics

Personal computers today typically provide several USB connections, but no longer have RS-232 serial ports, though this is still the *de facto* standard serial communication method for robotics and microcontroller hobbyists. As a result, we have opted to support direct USB connection to our Fab@Home system, despite the additional development work and (internal) complexity that this entails. We chose to use a microcontroller with an on-chip USB 2.0 peripheral, the Philips LPC-2148 ARM7TDMI (Royal Philips Electronics N.V.). This is a very high performance, 60MHz, flash-memory microcontroller with a wealth of peripheral functions for future expansion, including ADC, DAC, PWM, counter/timers, real-time clock, high-speed GPIO, UARTs, SPI, I2C, not to mention the USB2.0 peripheral. In addition, it has 512kB of flash memory, and 40kB of RAM. The large program memory has enabled us to make a very easily understood and extended packet data protocol for communication between the PC application and the firmware. We use the large RAM space to buffer motion commands so that real-time motion does not depend on variations in communication bandwidth. With our current protocol, we can buffer roughly 670 path points for 6 axes of control. The microcontroller is powered by the USB, and thus can be communicated with even

when the amplifier electronics are not powered. The high computational performance of the device enables the system to handle receiving and buffering path points, sending real-time status and position data, and controlling step and direction outputs for 6 axes at least 5kHz – we have not yet tested higher step rates. The microcontroller is available on a 1.5" X 2.5" board (Figure 5) with header connectors for all pins and a USB connector for $39.95 in single quantity (LPC-H2148, Olimex Inc.).
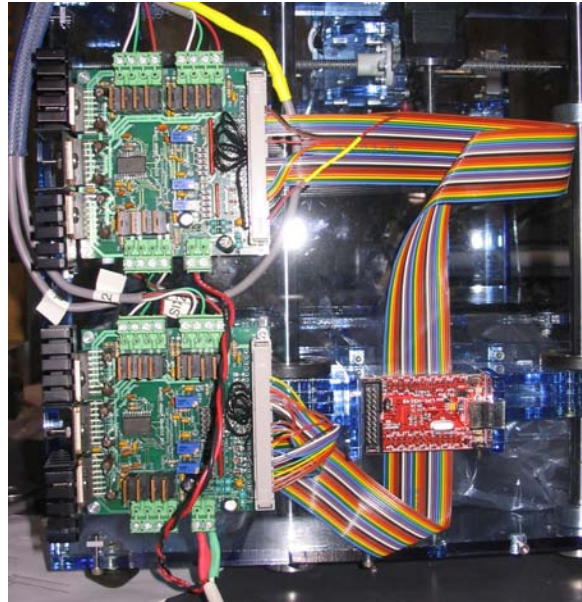

Figure 5. The electronics boards – two 3-axis stepper motor amplifiers on the left, LPC-H2148 microcontroller boards on right

Currently, we are using two 3-axis stepper motor amplifier boards (Figure 5) ($170 ea., Technological Arts, Inc.) to power the positioning system and syringe tool stepper motors. These boards have proven to be problematic, and we have had to upgrade some of the components on them to make them function reliably. These boards provide switch-mode current regulation for up to 3 bipolar stepper motors per board. The current regulation allows us to use 24VDC supply and 5V rated motors to get much higher acceleration (hence faster builds at finer resolutions), than would be possible at the nominal 5V. This is a design decision which increases cost significantly for the sake of making the technology more useable. We have found and are evaluating alternate amplifier boards which provide the same or improved functionality at less than half the price.

Software

The firmware for the LPC-2148 microcontroller was developed in C language, using the free GNU C ARM compiler, and Rowley CrossWorks for ARM development environment by Rowley Co. UK – available for 30 days unrestricted trial, and with a 99£ educational license, and a 499£ commercial license. CrossWorks is not essential - several freeware IDE's exist which work with GNU C ARM. The firmware performs the following main functions:

- receiving and parsing of packetized commands from the PC via the USB

- buffering of motion path segments for fabrication paths
- immediate execution of jog motion commands
- configuration of limit switches (present/absent for each axis and direction)
- communicating axis position, limit switch state, and other system status to the PC via the USB
- controlling step and direction outputs for up to 6 axes at > 5kHz step frequency

As mentioned before, the microcontroller has additional resources available for future expansion, and the firmware has been designed with ease of expansion in mind.

A PC application has been written which enables the user to control the machine, import, position, assign material properties to, and generate and execute manufacturing plans for geometry data which is imported in the form of STL files. In addition, the application has been designed around the concept that while the core application may eventually be improved by the user community, in the short term, the hardware configuration, the types of materials, and parameters for depositing materials would be far simpler and more interesting to explore and share. Thus the hardware configuration, material properties, and material deposition parameters are described in plain text parameter files. These files describe, for instance, the color and geometry data used to render the Fab@Home machine, the speeds and threading of the stepper motors, presence of limit switches, etc.

The PC application is targeted for the Microsoft Windows operating system. It is written in C++ using the Microsoft Visual Studio .NET development environment, Open GL for graphics rendering, and the Microsoft Foundation Class library for user interface components. The application has been designed with the aim of maximizing the intuitiveness of use (Figure 6). The user interface includes a 3D rendering of a Fab@Home machine which moves synchronously with the real-time position information sent back by the microcontroller. Dialog boxes allow importing and assigning material and tool properties to the part geometry, manual jogging of the axes via buttons and mouse scroll wheel, including the syringe tool motor, as well as a numerical view of the real-time position and status data from the microcontroller. The application also allows a rudimentary simulation of the fabrication process – the actual manufacturing plan is executed on a Fab@Home software emulator, and the motions are displayed in the GUI for quick checking of toolpaths.
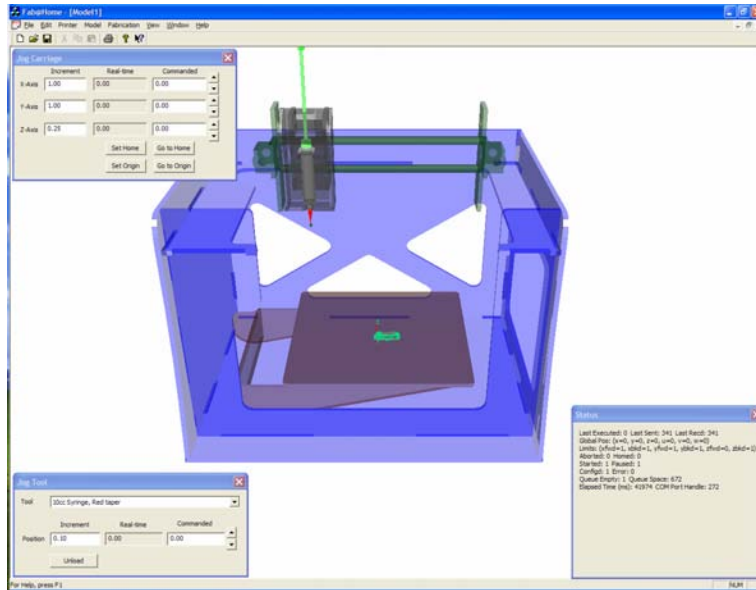
Figure 6. A screenshot from the PC application displaying a model ready for fabrication and dialog boxes for positioning and real-time status information

The workflow for Fab@Home consists of the following:
- connecting PC to Fab@Home via USB cable, and plugging in the Fab@Home power supply
- selecting and/or modifying/tuning parameter files to match the hardware and materials to be used
- starting the PC application
- loading the parameter files
- loading a syringe with piston, material, and nozzle, and mounting it in the tool; threading the tool leadscrew into the piston nut
- importing the geometry of the part to be fabricated
- assigning material and tool properties to part geometry
- automatic generation of a manufacturing plan
- if desired, simulated execution of the plan
- initialization of the Fab@Home printer
- homing of the axes so that GUI and physical positions match
- jogging axes to the desired origin for fabrication
- automatic execution of the manufacturing plan

Results

To date, we have produced three complete Fab@Home units, the first two of which have been delivered to users outside of Cornell University.

The very first unit was delivered to the AMTS FabLab in Pretoria, South Africa, on June 25, 2006. The AMTS FabLab is a publicly accessible CAD/CAM lab developed by the MIT Center for Bits and Atoms in partnership with the South African Government's Department of Science and Technology's Advanced Manufacturing Technology Strategy and the South African Council for Scientific and Industrial Research (CSIR). FabLab's, the invention of Prof. Neil Gershenfeld of MIT, are intended

677

to provide simple, highly automated CAD/CAM facilities and basic training to members of the general public in order to promote local invention and micro-enterprise – "Democratization of Invention". During the Digital Manufacturing Symposium held at CSIR during the week of June 25th, we were able to demonstrate Fab@Home to officials of the CSIR and AMTS as well as to visitors and staff of the FabLab. Unfortunately, it being the very first fully functional unit, both the software and the hardware had significant bugs, and demonstrations were limited to the deposition of single layers. Nevertheless, viewers were thrilled by the idea of a low-cost SFF system, and also with the visual appearance of the software and hardware. Gershenfeld (pers. comm.) views the Fab@Home kit as an example of a class of CAM equipment which can be cheaply manufactured using a the simple CAM tools available in a FabLab, with the result that FabLabs may be able to extend their own capabilities, and expand to have greater impact with little additional capital expense. The AMTS has committed resources to integrating Fab@Home into the CAM suite at the FabLab, and is using Fab@Home as a prototype for generating additional examples of kit CAM equipment.

Figure 7. The first unit being demonstrated for staff and users of the Soshanguve FabLab public access CAD/CAM laboratory, Soshanguve Township, South Africa

While in South Africa, we also demonstrated the Fab@Home at the Soshanguve FabLab, situated in Soshanguve Township, an entirely black African slum of 750,000 residents. Here again, the viewers, adults and children, most poorly educated, were quite excited at the appearance of the machine but typically failed to differentiate it from a CNC mill – a device with which they had some experience in the FabLab. A noteworthy reaction was that Fab@Home was an example of a cheap way to build a CNC mill in the FabLab. At this demonstration, we may have witnessed the aforementioned threshold of quality – the impact of the technology was lost for the lack of smooth operation and a meaningful example application. Nonetheless, the managers of the lab expressed and urgent desire for a Fab@Home unit for their lab – a request which we will fill in the near future.

We have very recently produced and delivered a Fab@Home unit (Figure 8) to a life science research laboratory at Rockefeller University, New York, NY. As of this unit, the hardware and firmware have been satisfactorily debugged. At this laboratory, research is underway which involves the slime mold organism *Dictostylium*, which transitions from single-cell independent life to colony organism with coordinated motion in response to scarcity of nutrition. The Fab@Home offers a uniquely simple and customizable platform for executing experiments on the effect of initial spatial distribution of *Dictostylium* cells in the environment and the spatiotemporal evolution of their colony aggregation response to starvation stress. A dual syringe tool was designed and included with this unit to facilitate the spatial control of both organisms and nutrients in experimental geometries. We are eagerly awaiting preliminary results from this first scientific application of Fab@Home.
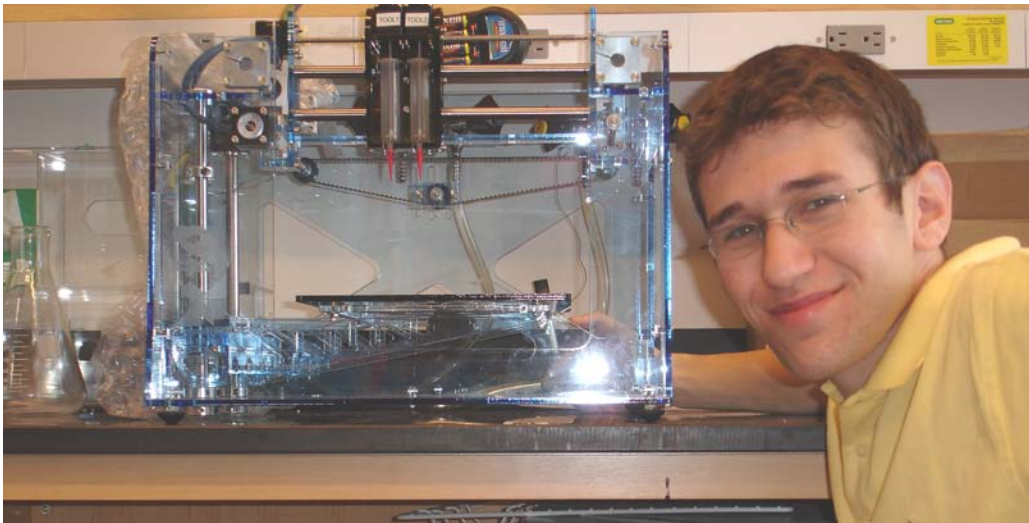

Figure 8. The second unit with a 2-syringe tool deployed by research student for Dictostylium research at Rockefeller University, NYC, NY

A third unit (Figure 9) has been assembled and is being used for ongoing debugging and development of the software. Minor hardware enhancements have been added to simplify the routing and strain-relieving of cables, and the hardware assembly process is becoming more routine.

The software has been improved to the point that bugs are no longer the most obvious limiting factor in performance. As Figure 9 shows, thin-walled, and multilayer parts are now achievable with RTV silicone rubber, but the part quality can be improved by more careful tuning of the deposition and machine control parameters.
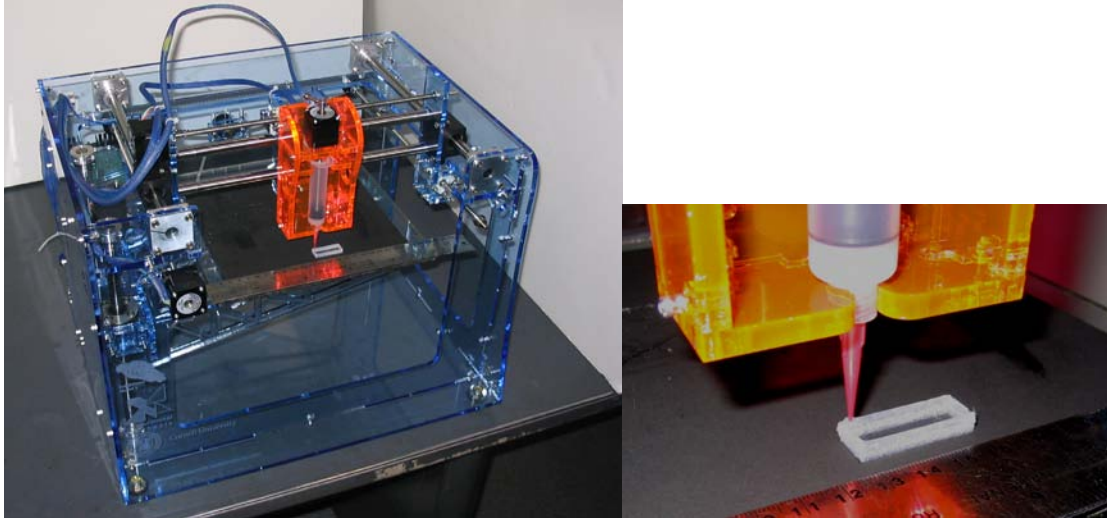
Figure 9. (left) The 3rd unit undergoing testing at Cornell University; (right) A successful part built by a Fab@Home using 1-part RTV silicone rubber

Future Work

Although we have reserved internet URL's for our website, we have only just begun the development of our user group website. The RepRap website [3] provides a nice example of the type of functionality we intend to implement in our website, including the user editability provided by a "wiki" framework. We are awaiting a first round of bids for outsourced manufacturing (MFG.COM) of the laser cut acrylic parts, which we will use to improve quality of our cost estimate for the kit. While writing software documentation and continuing improvements in the software, we will recruit volunteers at Cornell University to assemble units from parts in order to simplify, debug, and document the assembly procedure, parts lists, and source lists for the hardware. We also intend to have these volunteers and our first two users begin to populate the website with applications, tips, modifications, and bug reports in the interest of jump-starting a user community. Also in this interest, we are considering giving away a few more units to selected candidate users, such as the Soshanguve FabLab. In the longer term, we will begin to optimize the design to minimize cost, and enhance performance, perhaps developing design variants.

Conclusions

Few technologies can truly transform human society to a degree that computers have. The universal computer, from the early experimental machines of the '50s to the home computer today, has affected almost every aspect of human existence. It is not easy to foresee such disruptive technologies in advance, but it has not escaped many that solid freeform fabrication technologies bare many of the same traits. Like universal computation, universal manufacturing, embodied as today's freeform fabrication systems, has the potential to transform human society to a degree that few creations ever have.

In order to accelerate the spread and development of SFF applications and technology, and escape the "chicken and egg" paradox, we are developing an open-source, low-cost, personal SFF system kit, which we call "Fab@Home". The current kit

design has a parts cost of roughly $2000, requires only basic hobbyist tools and skills to assembly and use, and can be used to deposit almost any room temperature liquid or paste. We are also developing a website to foster a user community and to promote the exchange of ideas and improvements of the technology. We have thus far built three complete Fab@Home systems, and two of them have been delivered to users outside of Cornell University. The response of users, government officials, and the general public has been very positive, despite the problems associated with deploying prototype systems. Generating an initial wave of interest does not seem to be a concern – what remains to be seen is whether we can provide a system cheap enough and simple enough that for many people, the experience of obtaining and operating an SFF system does not overwhelm the experience of using and exploring the technology.

Bibliography

1. Gershenfeld N., (2005) *FAB: The Coming Revolution on Your Desktop – From Personal Computers to Personal Fabrication*, Basic Books
2. Lipson H. (2005) "Homemade: The future of Functional Rapid Prototyping", IEEE Spectrum, feature article, May 2005, pp. 24-31
3. Bowyer A., RepRap: The Replicating Rapid-Prototyper, http://reprap.org
4. Burns M., (1995) The Freedom to Create, in Technology Management, Volume 1, Number 4
5. Ceruzzi P.E., (2000) *A History of Modern Computing*, The MIT Press
6. Klein E.S., Vintage Machines, http://www.vintage-computer.com/altair8800.shtml
7. Blinkenlights Archaeological Institute, http://www.blinkenlights.com/pc.shtml
8. Wohlers T., (2006), Rapid Prototyping & Manufacturing State of the Industry, Wohlers Assoc.