GreatFET: Making GoodFET Great Again

Michael Ossmann
Great Scott Gadgets

Abstract

The open source GreatFET project provides a new general-purpose hardware interface for debugging, troubleshooting, and connecting to external devices. The platform improves upon the popular GoodFET in several ways, offering greater capabilities at a reduced cost while continuing important GoodFET traditions.

1 Introduction

Hardware hackers, developers, reverse engineers, and tinkerers often require the ability to connect a general-purpose computer to a target device. Many specialized tools are available for such purposes. These include debuggers, programmers, logic analyzers, and other test instruments.

In recent years, general-purpose interfaces such as the Bus Pirate [8] have become popular. These tools provide connectivity to wired communication protocols such as SPI or I^2C and sometimes support limited debugging or logic analysis capabilities.

A popular use for general-purpose interface tools is to read from or write to flash memory chips found on a target device [2]. Another use is to monitor inter-chip communications on a target. These are but two of many applications made possible at a low cost by general-purpose interface tools.

1.1 GoodFET

GoodFET was one of the first general-purpose interface tools popular in the information security community. After starting out as a debugger for MSP430 targets [4], the open source hardware platform quickly gained general-purpose interface capabilities after its developer, Travis Goodspeed, was inspired by the Bus Pirate [3].

More than twenty variants of the GoodFET hardware platform were developed by Goodspeed and friends over a five year period starting in 2009. These included platforms with integrated special-purpose interfaces such as wireless transceivers or target USB controllers. Some were designed for use as electronic conference badges.

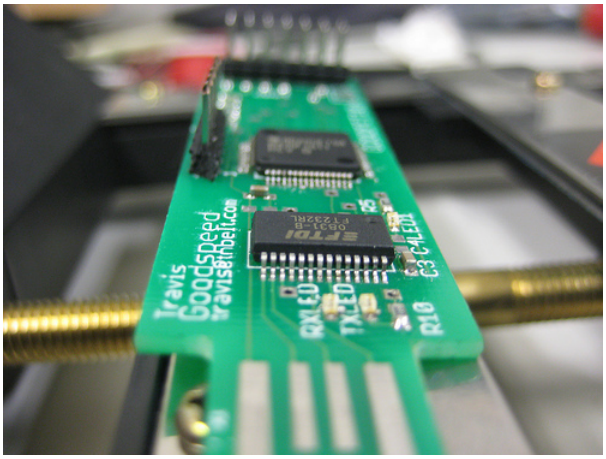

Figure 1: an early GoodFET design

GoodFET has enabled a surge of security research presented by Goodspeed, myself, and others since the project's inception. The scope of this research has ranged from whimsical modifications of children's toys [6] to the discovery of vulnerabilities in USB host stacks [1] to important new techniques for the monitoring of wireless communication systems [5].

1.2 GoodFET Drawbacks

Though the proliferation of hardware designs serves as an indicator of the project's success and popularity, it is unfortunate that a user must acquire or assemble a separate platform to implement each special function. The hardware fragmentation also has resulted in complex software that is difficult to maintain.

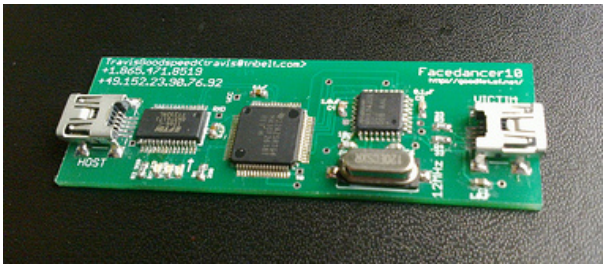

Figure 2: Facedancer, a GoodFET variant with target USB interface

The selection of microcontroller is another disadvantage of GoodFET. Most GoodFET designs use a 16-bit MSP430 microcontroller that is needlessly expensive. Originally chosen to maintain compatibility with firmware for a commercial debugger, the component has the great advantage that its manufacturer, Texas Instruments, makes free samples available. This is a boon for individuals building one or two boards for personal use, but the volume pricing is a hindrance to anyone hoping to manufacture large quantities. The other major chip on GoodFET, a USB-to-serial interface from FTDI, is also expensive for mass production, especially considering the availability of microcontrollers with integrated USB device controllers.

Another drawback of GoodFET is performance. Interfaces such as SPI are implemented by bit-banging and do not take advantage of higher speed peripherals available on the microcontroller. Perhaps the best example of performance limitations is Facedancer, a popular GoodFET variant that includes a secondary USB device controller for probing USB hosts. The Facedancer's target USB port implements Full Speed USB, ostensibly a 12 Mbps interface, but the connection between the USB device controller and the microcontroller is a 1 Mbps bottleneck.

To a great extent, each GoodFET hardware design was made to demonstrate one particular application with little regard for performance or for mass production. The only manufacturability consideration was the ease of hand-assembly by end users.

2 GreatFET

While GoodFET has experienced a decline in development over the past two years, the platform remains a popular and useful tool. I started the GreatFET project [7] in an effort to develop a successor that can continue the GoodFET tradition for years to come.

A key goal of GreatFET is to overcome deficiencies of GoodFET designs, particularly making the platform

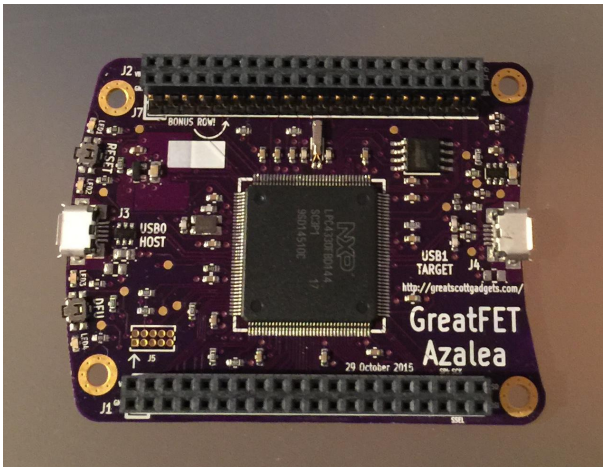

Figure 3: GreatFET One, formerly known as GreatFET Azalea

cost effective for mass production. GreatFET One, the first GreatFET hardware design, replaces the combination of MSP430 and FTDI USB chips with a single, lower cost, higher performance microcontroller, NXP's LPC4330. This 32-bit ARM Cortex-M4 operates with a CPU clock speed up to 204 MHz and features a Hi-Speed (480 Mbps) USB interface. Originally conceived as a way to save cost, the performance benefits of a replacement microcontroller significantly influenced the GreatFET architecture.

GreatFET One features not one but two USB ports. The secondary port, a Full Speed USB controller also built-in to the LPC4330, enables connectivity to target USB hosts or devices in the manner of Facedancer but with improved performance. This functionality allows GreatFET One to replace both GoodFET and Facedancer, the most popular GoodFET variant.

3 Neighbors

Perhaps the greatest improvement of GreatFET over GoodFET is an expansion interface that allows a wide variety of add-on boards to be plugged in to GreatFET One. These expansion boards, called "neighbors" in honor of Goodspeed's well-known appreciation of neighborliness, implement special functions in a modular fashion, eliminating the need for main-board variants that contributed to the complexity of the GoodFET project.

For example, the GoodFET project included The Next Hope Badge, a design manufactured only once as a conference badge featuring an NRF24L01+ wireless transceiver IC. The GreatFET project instead includes a simple neighbor with the same wireless transceiver and without the need to duplicate the microcontroller and

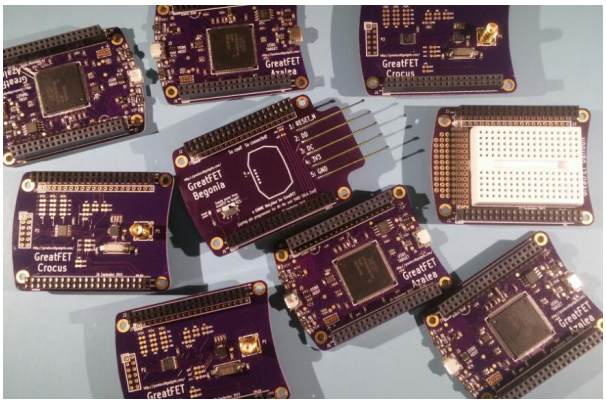

Figure 4: GreatFET One with an assortment of prototype neighbors

USB interface present on GreatFET One.

Because GreatFET One includes a large expansion interface of 80 pins (plus a 20 pin bonus row for optional use), a tremendous array of neighbors can be supported in the future. I have designed a few and plan to design more. I've also documented the neighbor interface [9] so that others may design their own neighbors compatible with GreatFET One.

4 Honoring GoodFET Traditions

The popularity of GoodFET has been greatly enhanced over the years by Goodspeed's habit of handing out bare Printed Circuit Boards (PCBs) for free or at a low cost. Indeed, a GoodFET PCB given to me at a conference was my first surface mount assembly project, encouraging me to delve into hardware hacking and electronics design.

Though the GreatFET One PCB will be somewhat more expensive than any GoodFET design, the tradition of hand-assembly will continue. I designed GreatFET One so that it could feasibly be assembled with a soldering iron, and I have made revisions specifically to improve the hand-assembly process. A higher performance platform with a larger number of parts, GreatFET One takes longer to hand-assemble than GoodFET, but it is still possible to do with a soldering iron in a single session.

An important feature of GoodFET, particularly for users who assemble their own boards, is that it has a built-in bootloader accessible over USB. This allows someone to build a board and immediately install firmware without requiring an additional piece of hardware (such as another GoodFET) to program it. While many microcontroller platforms have a chicken-and-egg problem for someone assembling their very first board, GoodFET neatly solves this problem by using a microcontroller with a built-in bootloader. GreatFET honors this tradition by making the LPC4330's USB bootloader available with the push of a button.

The GoodFET project is published under a BSD license, enabling hassle-free use of the software and hardware designs. I'm continuing this tradition by releasing GreatFET under a similarly permissive license.

5 Conclusion

The GreatFET project aims to succeed GoodFET by making possible all the popular functions of GoodFET while enabling great things that could not be done before. I hope that GreatFET, like GoodFET, will enlighten people who hadn't previously realized how accessible hardware hacking is. With enhanced performance, similar or lower cost, and a well documented expansion interface, GreatFET One is poised to serve as a base platform for future information security researchers, electronics hobbyists, and all sorts of curious folk. The great GoodFET tradition will continue with GreatFET.

6 Acknowledgments

I thank Travis Goodspeed for creating GoodFET, inspiring me to pursue electronics, and repeatedly demonstrating how a simple tool can open the door to wonderful and sophisticated research. Thank you to Dominic Spill for firmware and other contributions to GreatFET. Thank you to Taylor Streetman for assembly of prototypes. I thank the many neighbors, too numerous to name, who have shared thoughts and ideas I've incorporated into this and other open source projects.

References

[1] BRATUS, S., AND GOODSPEED, T. Facedancer USB: Exploiting the Magic School Bus. *Recon* (2012). https://recon.cx/2012/schedule/events/237.en.html.

[2] CZARNY, J., AND RIGO, R. Analysis of an encrypted HDD. *SSTIC* (2015). https://www.sstic.org/2015/presentation/hardware_re_for_software_reversers/.

[3] GOODSPEED, T. GoodFET. http://goodfet.sourceforge.net/.

[4] GOODSPEED, T. Improving the MSP430 FET. *Travis Goodspeed's Blog* (2009). http://travisgoodspeed.blogspot.com/2009/03/improving-msp430-fet.html.

[5] GOODSPEED, T. Promiscuity is the nRF24L01+'s Duty. *Travis Goodspeed's Blog* (2011). http://travisgoodspeed.blogspot.com/2011/02/promiscuity-is-nrf24l01s-duty.html.

[6] GOODSPEED, T., AND OSSMANN, M. Real Men Carry Pink Pagers. *ToorCon* (2010). https://www.youtube.com/watch?v=WGU30mF_dgM.

[7] GREAT SCOTT GADGETS. GreatFET. http://greatscottgadgets.com/greatfet/.

[8] LESNET, I. Bus Pirate.
http://dangerousprototypes.com/docs/Bus_Pirate.

[9] OSSMANN, M. How to Design a Neighbor.
https://github.com/greatscottgadgets/greatfet/wiki/How-to-Design-a-Neighbor.