

[breakout.pdf](#)

## bootloader

```
pi@raspberrypi ~/mchck/bootloader/usb-dfu $ sh -x flash.sh
+ ruby ../../toolchain/../../programmer/flash.rb name=mchck:dev=/dev/serial/by-id/usb-mchck.org_MC_H
Attaching debugger...
KINETIS: holding system in reset
KINETIS: releasing core from reset
done.
Programming 2600 bytes of firmware to address 0...
programming 0, 0 of 3072
programming 0x400, 1024 of 3072
programming 0x800, 2048 of 3072
done.
resetting...
done.
```

## blink

```
pi@raspberrypi ~/mchck/examples/blink $ sh -x flash.sh
+ ruby ../../toolchain/../../programmer/flash.rb name=mchck:dev=/dev/serial/by-id/usb-mchck.org_MC_H
Attaching debugger...
KINETIS: holding system in reset
KINETIS: releasing core from reset
done.
Programming 980 bytes of firmware to address 0xc00...
programming 0xc00, 0 of 1024
done.
resetting...
done.
```

## flashing script

```
pi@raspberrypi ~ $ cat mchck-swd-flash.sh
#!/bin/sh -xe

tail -f /var/log/kern.log &
pidof udevadm || sudo udevadm monitor &

cd mchck/bootloader/usb-dfu/
sh -x flash.sh
cd -
cd mchck/examples/blink/
sh -x flash.sh
cd -
```

## Arduino Teensy 3 compatibility

To be exact mchck can be compatible with Teensy 3 which in turn is Arduino compatible - <https://www.pjrc.com/teensy/schematic.html>

<https://github.com/mali1741/mchckduino>

DFU loader fixes: <http://hackaday.io/project/662-mcslck>

All pin names are from teensy, so you will have to remap them to names on mchck board in your head!

