BeagleBoard 101

Jason Kridner
jkridner@beagleboard.org

June 7, 2010

Archived at:
http://beagleboard.org/esc

Agenda

- Overview
  - Some BeagleBoard.org history
  - BeagleBoard-xM features
  - Classroom setup
  - Validating the BeagleBoard-xM (hands-on)
- Getting help from peer developers
  - How to ask for help
  - The BeagleBoard landscape
  - Resources for more information and support
- Play time (hands-on + Q&A)

What's in a name…

- Bring your own peripherals
- Entry-level cost ($149/179)
- ARM Cortex-A8 (superscalar)
- Graphics and DSP accelerated
- Linux and open source community
- Environment for innovators

Timeline

- Feb 2008: Rev A concept demonstrated at meeting with key kernel maintainers and open source developers at TIDC
- Jun 2008: Rev B launched broad availability with Digi-Key
- Mar 2009: First hands-on ESC BYOES training
- May 2009: Rev C doubles RAM to 256MB
- Jan 2010: Rev C4 bumps performance to 720MHz and resolves USB power supply stability issue
- Jun 2010: xM Rev A board demonstrated at hands-on ESC BYOES training

Community development

$149

Personally affordable

> 2,500 participants and growing

Wikis, blogs/RSS, promotion of community activity

Active & technical community

Freedom to innovate

Open access to hardware documentation

Android, Ubuntu, Angstrom, FFmpeg, MeeGo, Symbian, …

Opportunity to tinker and learn

Free software

Learn today. Design tomorrow.
ESC
Chicago • June 7 - 9, 2010

Why such an active community?

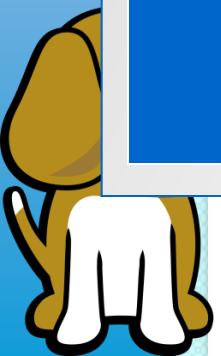
Affordable

Freedom to tinker at all levels

Lots of open starting points

Large and experienced community

Open ecosystem provides real options



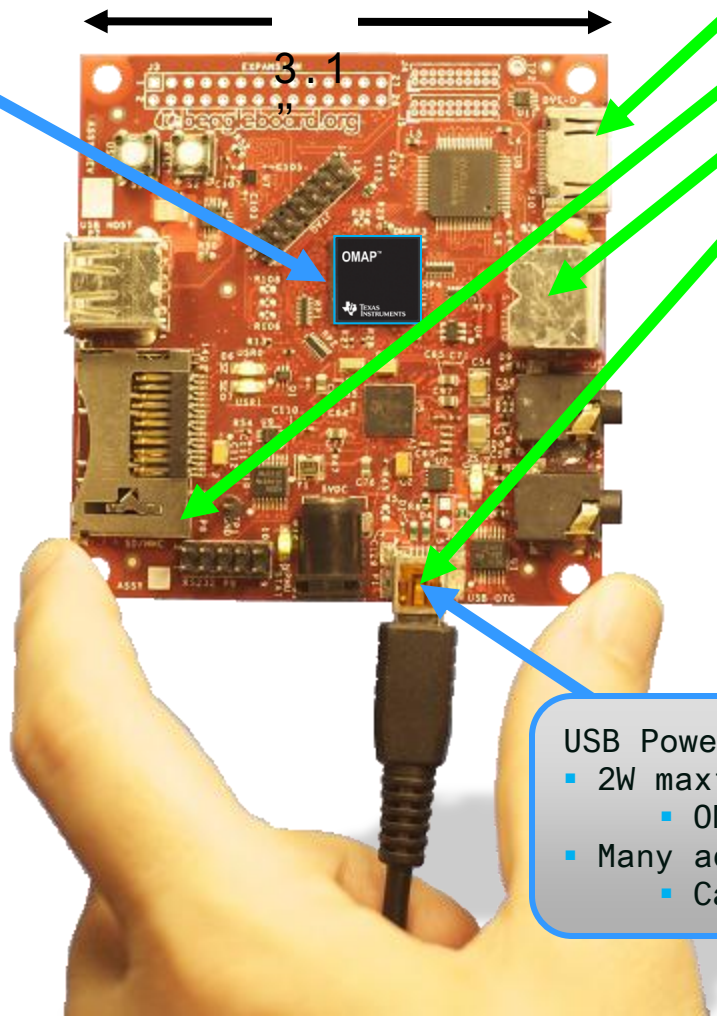
- $179 for same core processing
  - used in more expensive, yet popular, commercial products
- Focus on open source/hardware, DIY
  - Tens-of-thousands of boards sold exclusively in small quantities
  - All design, test, web, etc. materials shared
- Embedded high-level OS training
  - Ubuntu, Debian, Angstrom, Gentoo, WinCE, Symbian, QNX, and many others
- The BeagleBoard community shares
  - Over 150 registered projects on BeagleBoard.org
  - Part of the Google Summer of Code with 6 on-going projects to improve Linux, XBMC, and other open source
  - Average of around 5 articles or blog posts a day
  - Over 2,500 English-language mailing list subscribers with additional dedicated mailing lists in Japanese and Portuguese and numerous project oriented mailing lists in dozens of languages
  - Hundreds of followers on each of Facebook, Twitter, and LinkedIn
- Rich ecosystem using the design materials
  - Compatible or enhanced system-on-module/computer-on-module designs
    - See http://beagleboard.org/resources
  - Innovative mobile computers (TouchBook)
  - Radios (BeagleBrick)
  - Modular rapid prototyping development systems (Bug2.0)
  - And many add-ons…

Fast, low power, flexible expansion

OMAP3530 Processor
- 720MHz** Cortex-A8
  - NEON+VFPv3
  - 16KB/16KB L1$
  - 256KB L2$
- 430MHz C64x+ DSP
  - 32K/32K L1$
  - 48K L1D
  - 32K L2
- PowerVR SGX GPU
- 64K on-chip RAM

POP Memory
- 256MB* LPDDR RAM
- 256MB NAND flash

* Revisions before C had 128MB
** Revisions before C4 were 600MHz



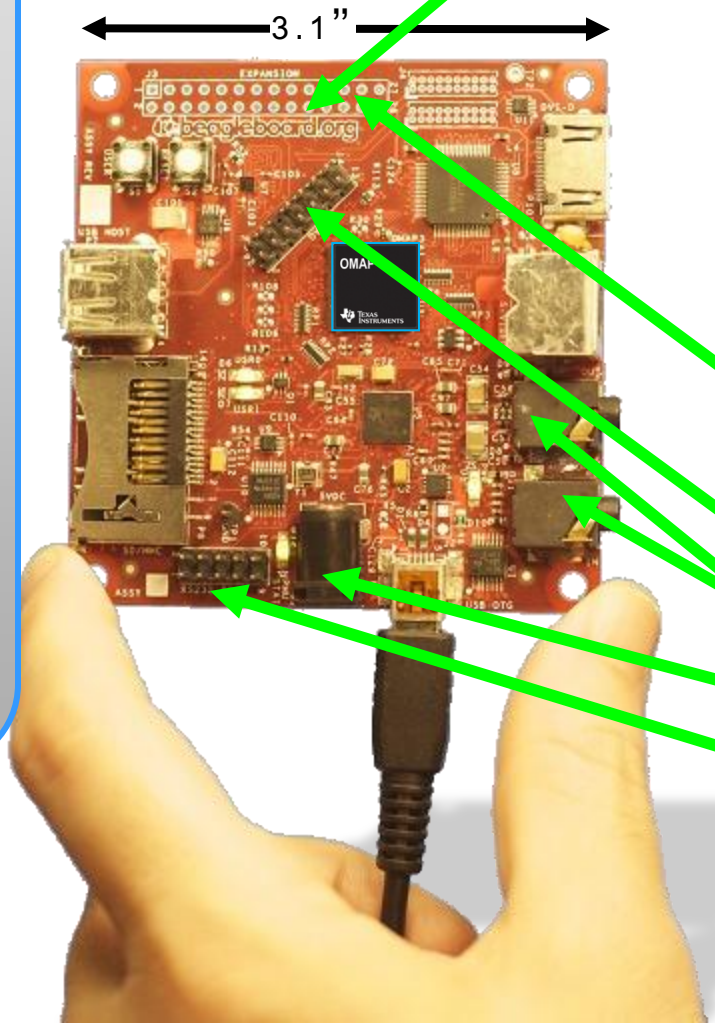
Peripheral I/O
- DVI-D video out
- SD/MMC+
- S-Video out
- USB 2.0 HS OTG
- I²C, I²S, SPI, MMC/SD
- JTAG
- Stereo in/out
- Alternate power
- RS-232 serial

USB Powered
- 2W maximum consumption
  - OMAP is small % of that
- Many adapter options
  - Car, wall, battery, solar, …

And more…

Collaboration at **BeagleBoard.org**
- Live chat 24/7 via IRC
- Links to project downloads

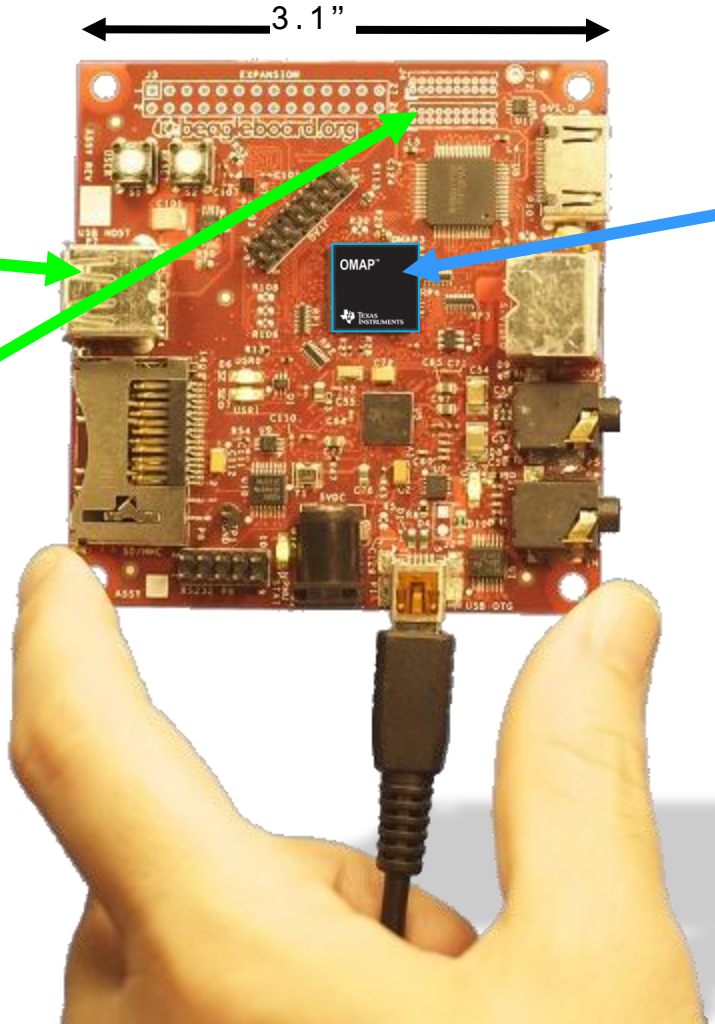
Other Features
- 4 LEDs
  - USR0
  - USR1
  - PMU_STAT
  - PWR
- 2 buttons
  - USER
  - RESET
- 4 boot sources
  - SD/MMC
  - NAND flash
  - USB
  - Serial



Peripheral I/O
- DVI-D video out
- SD/MMC+
- S-Video out
- USB HS on-the-go
- I²C, I²S, SPI, MMC/SD
- JTAG
- Stereo in/out
- Alternate power
- RS-232 serial

New for Revision C

Peripheral I/O
- USB HS/host-only
  (in addition to existing USB HS on-the-go)
- LCD expansion



256MB LPDDR RAM
(up from 128MB)

USB-powered BeagleBoard-xM unleashes community-oriented development

xM means Extra MHz and Extra MB

Breaks 1GHz barrier and offers 512 MB of memory for $179

$179

- 2,000 Dhrystone MIPS performance with ARM® Cortex™-A8
- 512MB POP memory enabling
  - Native builds of Ubuntu and other distros
  - More multitasking with complex apps like Firefox or OpenOffice.org
- Robust expansion with more direct connectivity without external hubs; on-board Ethernet and five USB 2.0 ports
- USB-powered board via low power processor integration
- Active and growing open source community at beagleboard.org

BeagleBoard-xM

Laptop-like performance

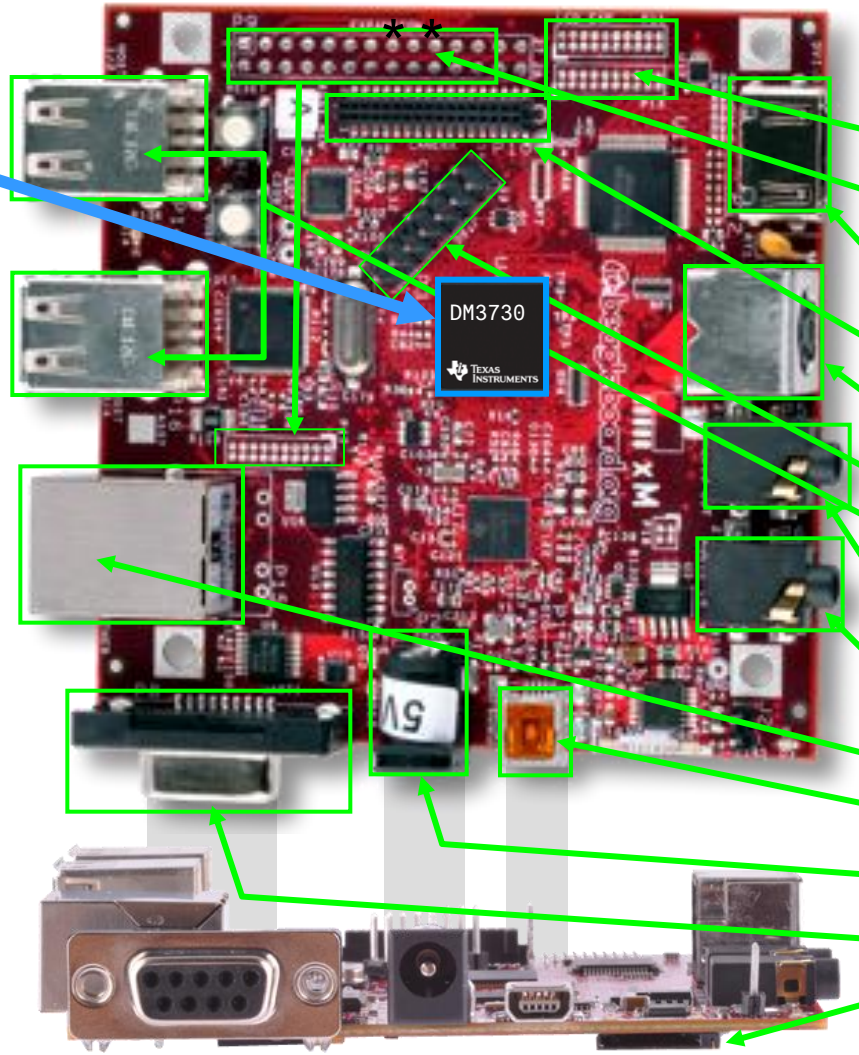
DM3730 processor (AM37x-compatibile)**

- 1GHz** superscaler ARM® Cortex ™-A8
- More than 2,000** Dhrystone MIPS
- Up to 20** Million polygons per sec graphics
- 512KB** L2$
- HD video capable C64x+™ DSP core

POP Memory**

- 512MB** LPDDR RAM

3.25”



Desktop-style USB peripherals and embedded style expansion

- LCD Expansion
- I²C, I²S, SPI, SD Expansion
- DVI-D
- Camera Header**
- S-Video
- JTAG
- USB 2.0 Hub**
  - 4-port
- Stereo Out
- Stereo In
- 10/100 Ethernet**
- USB 2.0 HS OTG*
- Alternate Power
- RS-232 Serial*
- microSD Slot*

* Supports booting from this peripheral
** Change between Rev C4 and BeagleBoard-xM

Cortex™-A8 : Block Level View

Dual issue In-Order Superscalar Pipeline

- 2.0 DMIPS/MHz

Support for ARMv7

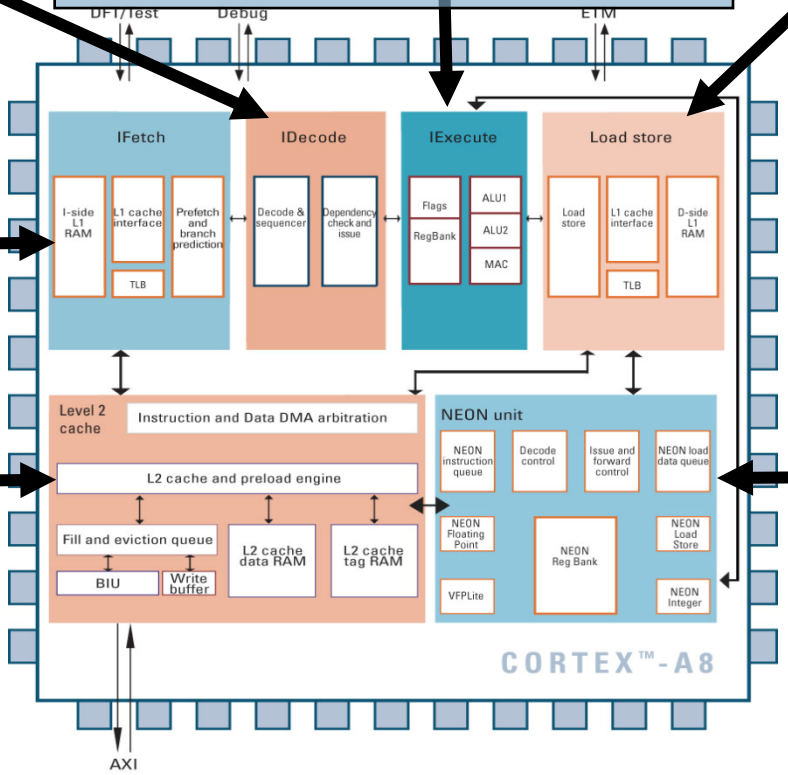
- Added new support for Thumb-2, Thumb-2EE(,Jazelle-RCT) and NEON

Advanced Dynamic Branch Prediction

- 95% accurate across industry benchmarks

Integrated L2 Cache 256KB (low latency/high BW i/f w/L1)

- Optimizes access to larger data sets and minimizes bus traffic

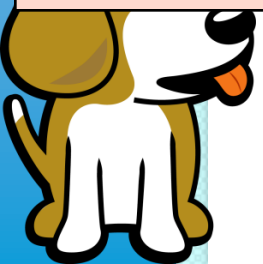


High speed Level 1 Caches (16KB)

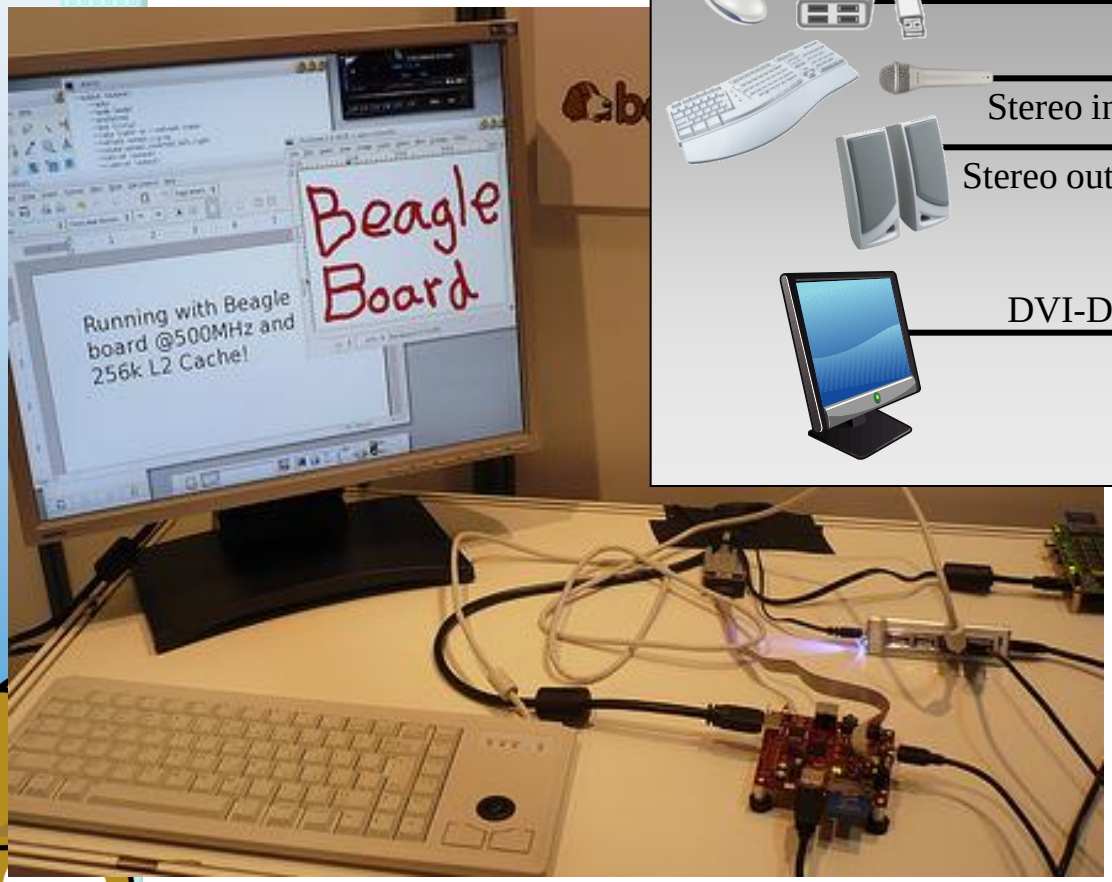
- Dual 32 entry memory translation TLB

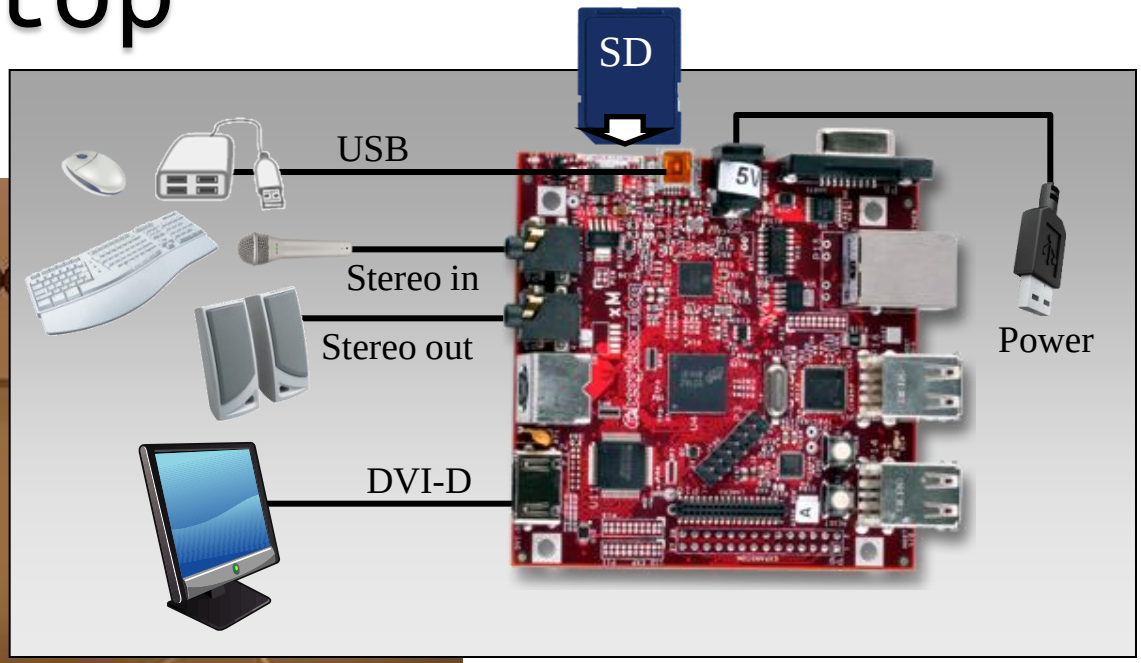
Integrated NEON co-processor for media and signal processing

- 2 to 4x performance improvement
- Integer and Floating Point support
- VFPv3-IEEE754 compliant (single and double precision) floating point support.



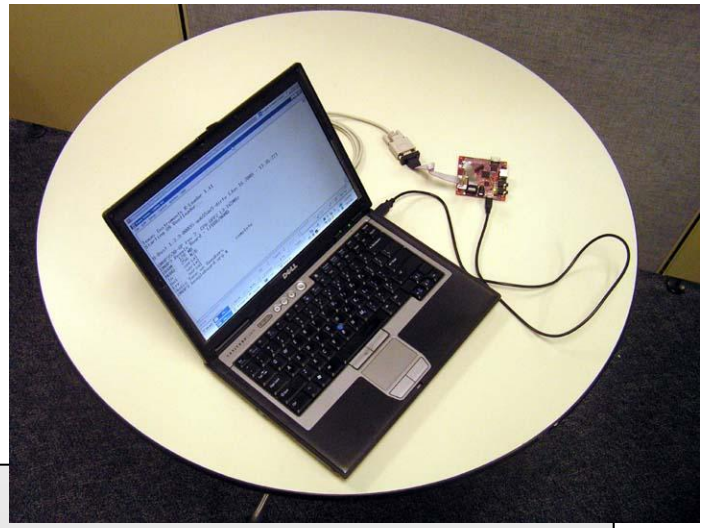
Use Beagle Board-xM like a desktop

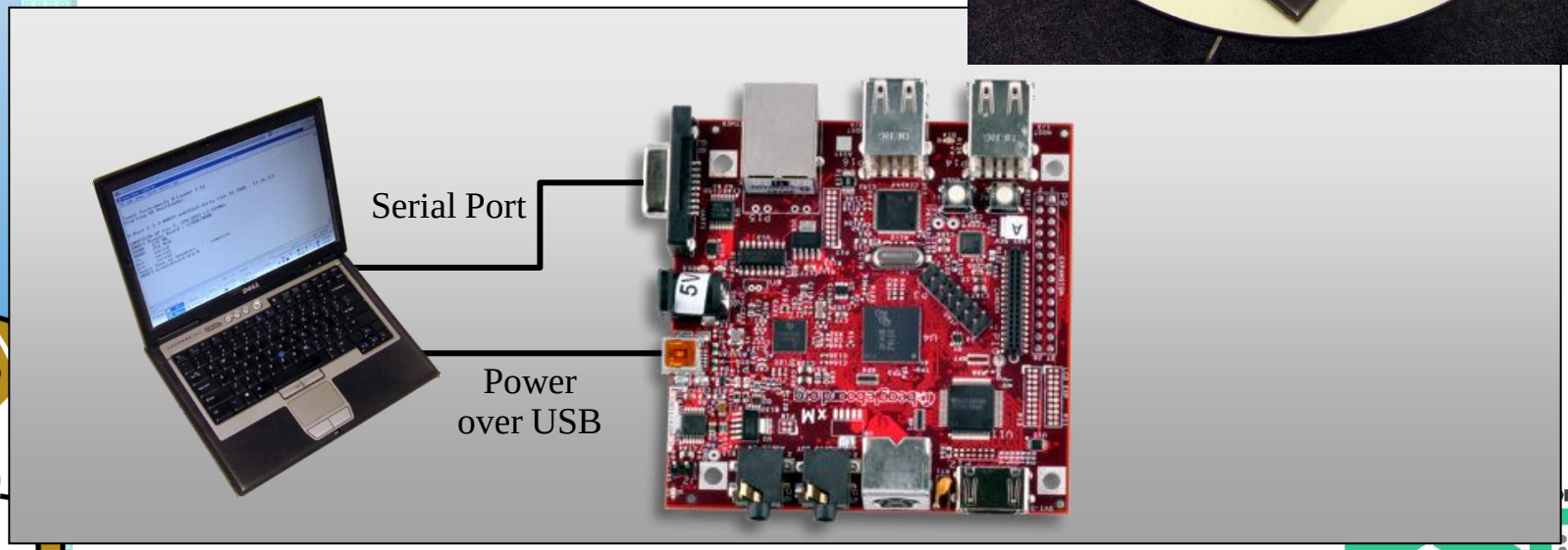




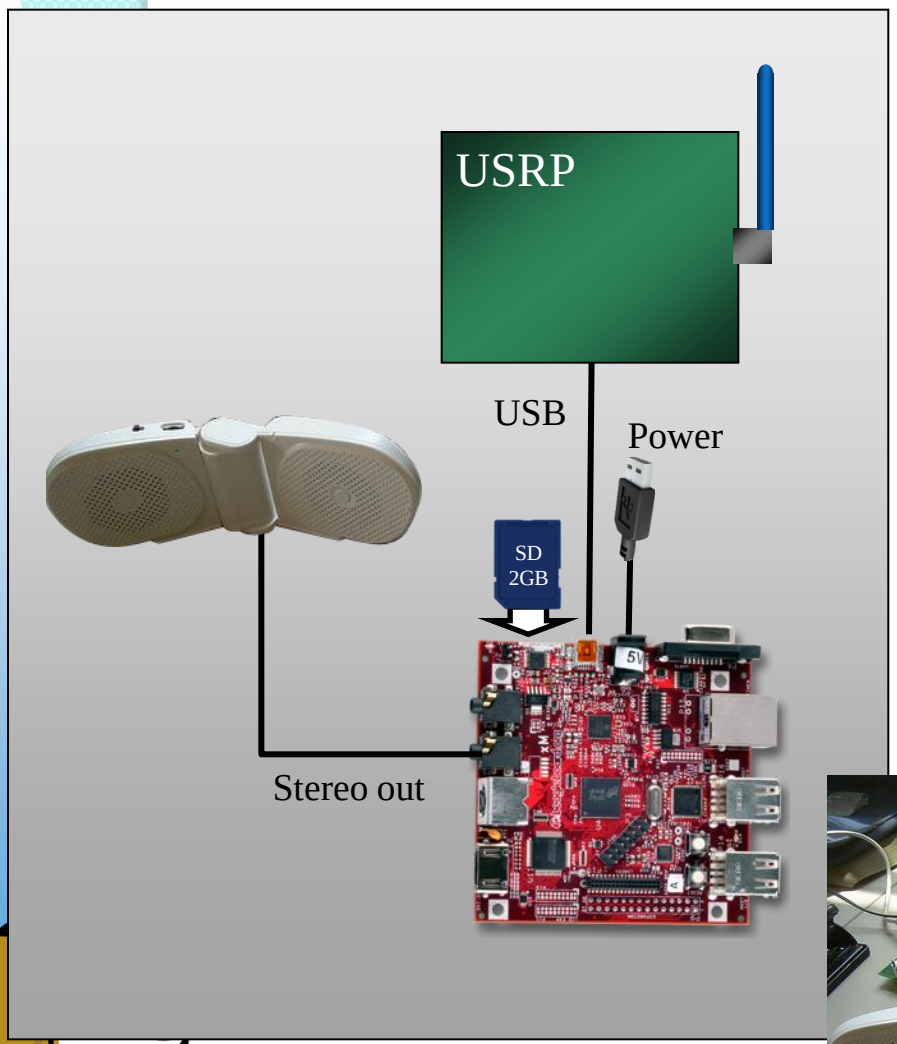
Note: Beagle Board can be powered from the alternate jack (as shown) or via USB

Take Beagle Board-xM anywhere & crank code on the go





Expand your Beagle Board-xM



Turn innovations into mass-produced products to share with the world

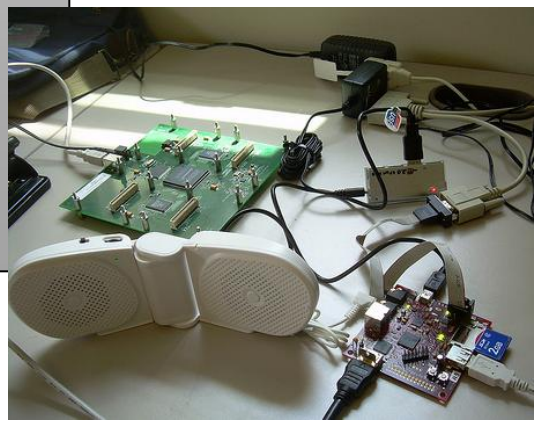


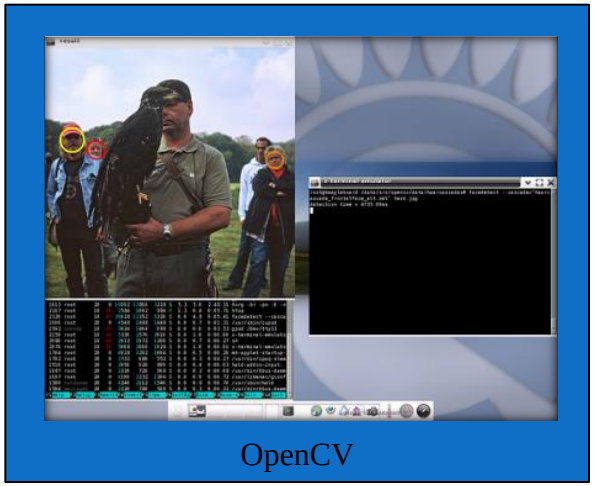
Photo by Philip Balister

Learn today. Design tomorrow.
ESC
Chicago • June 7 - 9, 2010

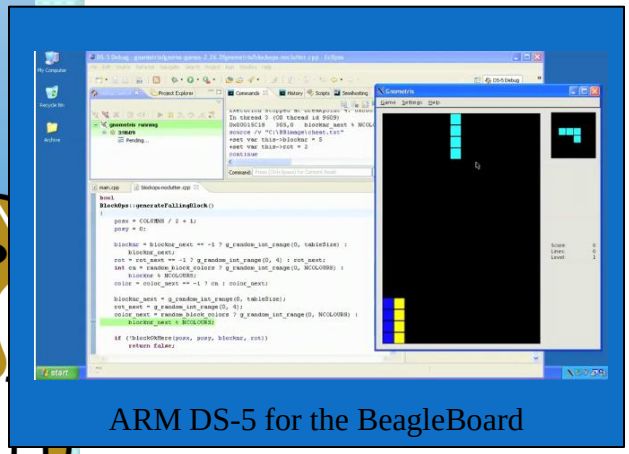
Open source, do-it-yourself, and pro developers embracing the BeagleBoard

FFmpeg BeagleBoard Video Wall

- Over 150 registered projects on http://beagleboard.org/project and hundreds of development activities on-going…
- Firefox
- Ubuntu 10.04
- Android
- Gnome
- Angstrom Distribution
- Gentoo
- WinCE
- QNX
- Flash
- TimeSys LinuxLink
- MontaVista MVL6 and Montabello
- RidgeRun SDK
- ARM DS-5 and ALIP
- Halcon machine vision
- BeagleBoard video wall (>1080p video)
- …



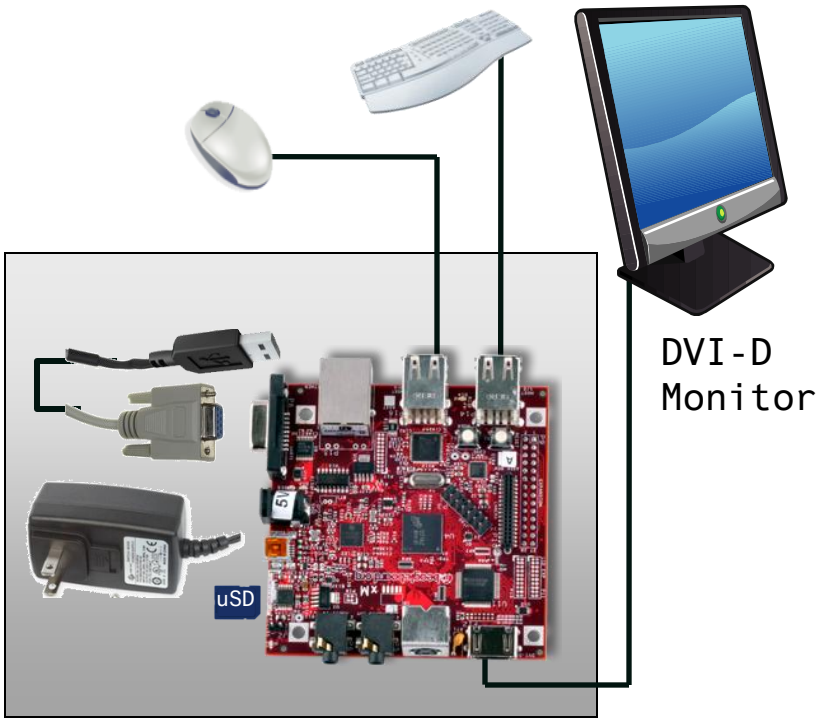
OpenCV



ARM DS-5 for the BeagleBoard

Classroom setup

Equipment at ESC Chicago



- For you to keep
  - BeagleBoard xM Rev A
  - microSD card
    - Content for each class
  - USB-to-serial adapter
  - 5V power adapter
- For use in the labs
  - DVI-D monitor and cable
  - USB keyboard and mouse

Desktop Computer Configuration

Boot-up

- Connect everything besides power
- Apply power
- “Matrix” application starts in about 45 sec
  - LED USR1 (D6) gives a “heartbeat”
  - Click “Exit” on the lower right
    - Exits to the GNOME desktop
- To shut down: System → Shutdown

Verifying the BeagleBoard-xM hardware

Hands-on

Verifying the hardware

- Code images, procedure, and sources are provided to verify the board functionality
  - Links to the diagnostics found at http://BeagleBoard.org/support
  - Includes bootloader, Linux kernel, and demo file system for testing
- These sources act as examples for software developers

I'm still updating the /support site to include all the information for the xM.

Get into the minimal ramdisk image

- Apply power WITHOUT the USER button
  - Applying power with the USER button pressed will result in the board not booting
- Press and hold the USER button within 3 seconds until one of the LEDs starts to blink
  - Timeout in u-boot.bin needs to occur, then USER button is sampled to load ‘user.scr’ rather than ‘boot.scr’
  - MLO → u-boot.bin → user.scr → ramdisk.gz / uImage
- Monitor should come to a login prompt
  - Use ‘root’ and press <ENTER> for password

Expect updates to the ramdisk image to include some example scripts that got missed in building the current release image.

Access the SD from the RAMDISK

- mkdir /proc
- mkdir /sys
- /etc/init.d/sysfs.sh
- mkdir -p /media/mmcblk0p2
- mount /dev/mmcblk0p2 /media/mmcblk0p2

Memory

testmem

- Boot minimal image
- opkg install memtester
- memtest 410M

LEDs

testleds

- export LED = “/sys/class/leds/beagleboard\:\:”
- echo “none” > ${LED}usr0/trigger
- echo “1” > ${LED}usr0/brightness
- echo “0” > ${LED}usr0/brightness
- echo “heartbeat” > ${LED}usr0/trigger

USER button

testuserbtn

- export GPIO = "/sys/class/gpio"
- echo "4" > $GPIO/export
- echo "in" > $GPIO/gpio4/direction
- cat $GPIO/gpio4/value

Read events

testhid

- Kernel documentation: http://git.kernel.org/?p=linux/kernel/git/torvalds/linux-2.6.git;a=blob;f=Documentation/input/input.txt
- opkg install evtest
- evtest /dev/input/event0
  - Press the “USER” button
  - ^C to exit
- evtest /dev/input/event4
  - Move the mouse
  - ^C to exit

Access monitor EDID

- cd /sys/bus; ls; cd
- cat /sys/bus/i2c/devices/3-0050/eeprom
- i2cdump -y 0x3 0x50 b
- decode-edid
- fbset

This looks a bit different now and requires installation of the i2c-tools

USB OTG and EHCI

- cd /sys/bus/usb/devices
- ls
- cat usb1/speed
- cat usb1/1-2/1-2.2/manufacturer
- cd
- lsusb

Networking

- Copy linux.inf from SD card to host and connect Beagle
- ifconfig
- nano /etc/networking/interfaces
- ifdown usb0; ifup usb0
- ifconfig
- Configure your host using linux.inf
- ping 192.168.123.1
- VNC
  - x11vnc &
  - Connect with your VNC viewer from your host
- Synergy
  - Start Synergy server on your host
  - synergyc --daemon --restart 192.168.123.1

Getting help from your peers

Understanding the basics of Linux

How to ask for help

http://catb.org/~esr/faqs/smart-questions.html

- Know the on-line resources
- Know the on-line community
- Know the manual
- Listen to the answers
- Pay for performance
- Share the answers you find

Order of resources

1. Search beagleboard.org, eLinux.org, the mailing list archive, and IRC logs
2. Read and search BBSRM_latest.pdf
3. Check the http://beagleboard.org/faq link
4. Search the web
5. Try something
   - Gives you some perspective on what to ask
6. Ask on IRC and be patient/polite
   - Doesn't disrupt everyone
7. Mailing list
   - Individual developers will go away if load isn't shared

The community perspective

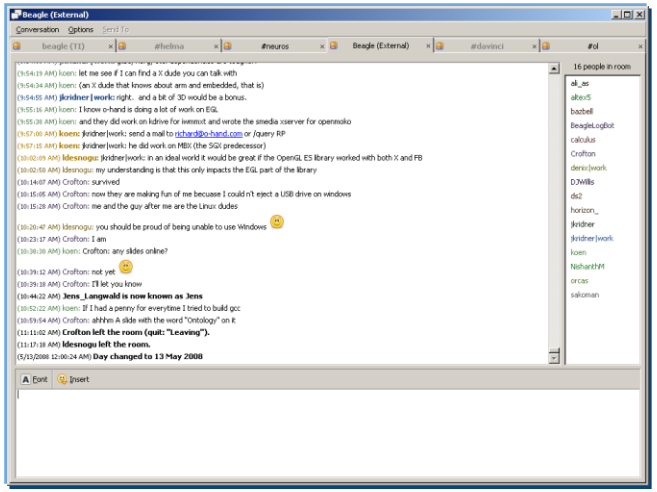
- Earn respect by saying what you've done and how you've tried to find an answer
  - Where did you search?
  - What did you try on the board?
- You aren't entitled to an answer
  - Show that you are willing to work for it and the community will feel you are a part of it
  - Impatience implies that your time is more valuable than others in the community

Chat, mail, forums, blogs, and wikis!

- All exist because they all solve different problems
- Chat allows you to know someone's listening
  http://beagleboard.org/chat or #beagle on irc.freenode.net
  - Great for beginner questions and rapid coordination
- Mail allows you to reach almost anyone
  http://groups.google.com/group/beagleboard
  - Brings larger group into the conversation
  - Provides you with a personal log in your inbox
- Forums helps get the threads organized
  https://community.ti.com/forums/32.aspx (minimal activity to avoid disrupting community critical mass)
- Blogs provide emphasis, filtering, and timeliness
  http://beagleboard.org/news and http://beagleboard.blogspot.com
- Wikis enable inputs to become documentation
  http://eLinux.org/BeagleBoard and http://code.google.com/p/beagleboard/wiki

Chat on IRC

- http://webchat.freenode.net
  - #beagle: discussion of the BeagleBoard
  - #gst-ti: discussion of GStreamer with TI DSP components
  - #ubuntu-arm: discussion of Ubuntu on ARM processors
  - #rowboat: discussion of Android on OMAP & Sitara devices
  - #linux-omap: discussion of OMAP Linux kernel
  - #davinci: discussion of TI DaVinci products
  - #neuros: discussion of Neuros open source devices
- IRC clients
  - http://beagleboard.org/chat
  - http://pidgin.im
  - http://www.mirc.com
  - http://en.wikipedia.org/wiki/List_of_IRC_clients
  - http://www.ircreviews.org/clients



Angstrom and Open Embedded

- Angstrom is what you are running
- OE is a build tool
  - Used by RidgeRun, Mentor Graphics, MontaVista, and many others
  - Builds many distributions besides Angstrom
- http://www.angstrom-distribution.org/

Ubuntu

- Most popular Linux distribution
- Has support for the BeagleBoard with a netbook installer
- Builds all packages natively

Android

- Uses (most) of the Linux kernel, but own versions of user-space applications
- Runs applications within a virtual machine
- At least half-a-dozen companies provide commercial support for Android on the BeagleBoard
- Rowboat is the one endorsed by TI
- 0xdroid (0x1ab) and Embinux are also interesting and free

MeeGo

- The combination of Moblin and Maemo
- Maemo was the first of the two and started on OMAP processors
- Good support on the BeagleBoard with demonstrations directly from the Linux Foundation
- Initially focused on Internet Tablets and Netbooks
- Very interesting for automotive infotainment

Gentoo

- Builds every package from source
- The Linux distribution the BeagleBoard.org web server runs
- Builds ARM applications both natively and cross
- This might be a useful source
  - https://www.slashorg.net/48-Gentoo-port-for-BeagleBoard.html

QNX

- Check out Foundry27
- Has a free version to test out on the BeagleBoard
  - http://beagleboard.org/project/QNX+Neutrino+on+OMAP/

Symbian

- Initial open source release was on the BeagleBoard
  - http://developer.symbian.org/wiki/index.php/BeagleBoard_Quick_Start
  - http://beagleboard.org/project/symbian/

WinCE

- Several commercial ports available from various providers
- Possible to adapt TI's EVM WinCE to BeagleBoard
  - http://beagleboard.org/project/evmonbeagle/

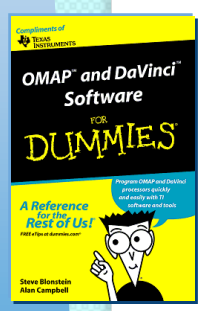
Participating in the community

- Joining the herd of cats
  http://lwn.net/talks/elc2007
- Building Community for your open source project
  http://www.eclipsecon.org/2006/Sub.do?id=268
- Video of Greg Kroah-Hartman on the Linux kernel
  http://www.linuxelectrons.com/news/linux/16774/greg-kroah-hartman-linux-kernel
- Sending kernel patches upstream
  http://wiki.omap.com/index.php?title=Patch_upstream_sending

Baseline tools and software

http://beagleboard.org/resources

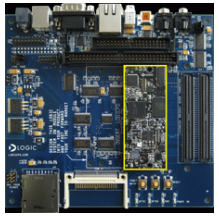
- Hardware verification procedure (http://beagleboard.org/support)
  - GPL x-load, u-boot, Linux kernel, and demo distro for validation
  - Code images, procedure, and sources are provided to verify the board functionality
- GPL ARM GNU compiler collection (GCC)
  - Code Sourcery version 2009q1 is the latest supported by TI
    - Runs on Linux/Windows and generates ARMv7/Thumb2
  - Angstrom version is utilized in ESC training and demo image on xM
- Access to C6000 with compilers and open source software
  - Free TI C6000 compiler for non-commercial use
    - x86-Linux hosted (ARM hosted version in evaluation)
  - GPL GCC compiler in progress (http://linux-c6x.org)
  - C6Run (DSPEasy) project to simplify development model
  - BSD/GPL DSP/Link interface software
- Free 3D graphics libraries (OpenGLES 2.0)
- Free production audio/video codecs for the DSP



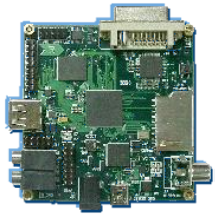
Some hardware options

http://wiki.omap.com/index.php?title=OMAP3_Boards

- TI/Mistral OMAP35x EVM
- Nokia Internet Tablets
- LogicPD OMAPZoom
- Gumstix Overo
- Analogue & Micro Cobra3530
- Cogent CSB740



LogicPD OMAP35x Dev. Kit / Medical EVM
5.75" x 6.25"

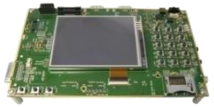


Mini Board
3" x 3"

LogicPD OMAP34x Mobile Development Kit
3.8" x 6.3" x .95"

Gumstix Overo

Beagle Board
3" x 3"



OMAP3 5x EVM
4.25" x 7"

OMAP34x SDP
8.5" x 11"

Not to scale.
Approximate size noted (in inches)

Many tools options

http://focus.ti.com/dsp/docs/dspplatformscontenttp.tsp?sectionId=2&familyId=1525&tabId=2224

Many OS vendors are OMAP35x not listed here

Tool / Top features	Debug	Compile	Other
	Low-level ARM and DSP	Low-level ARM (ARMv7) and DSP (NEON roadmap)	Power-aware debug
RealView® Tools by ARM®	Low-level ARM	Application-level ARM (ARMv7, NEON)	
LAUTERBACH	Low-level and app ARM and DSP	None	Extensive trace
Green Hills® SOFTWARE, INC.	Low-level and app ARM and DSP	Low-level ARM	Trace
CodeSourcery	Linux application debug	Linux kernel/app ARM (ARMv7, NEON)	

•Cortex-A8 uses ARMv7 instructions

Additional third party information: here