

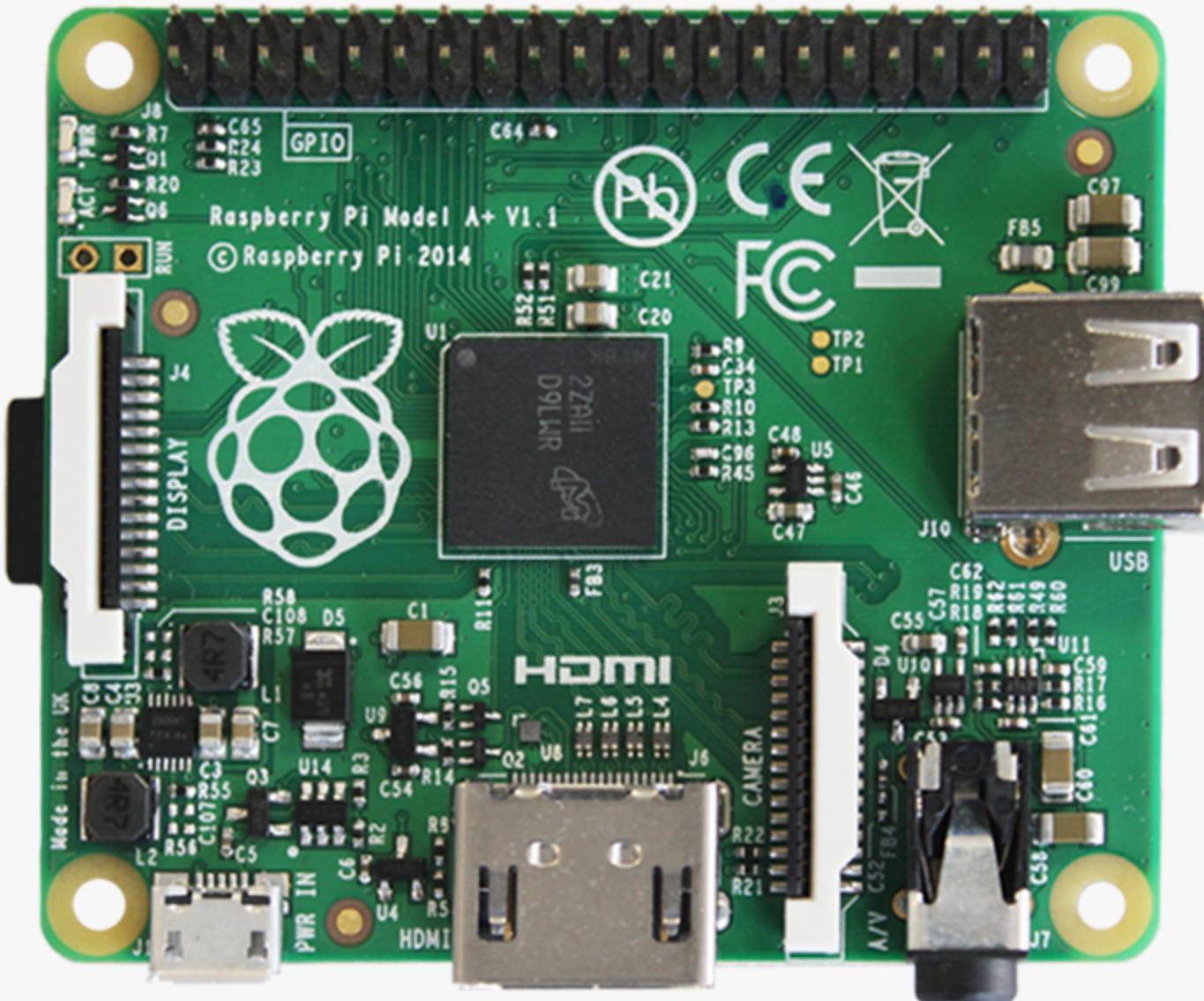


<http://quinnndunki.com/blondihacks/?p=1347>

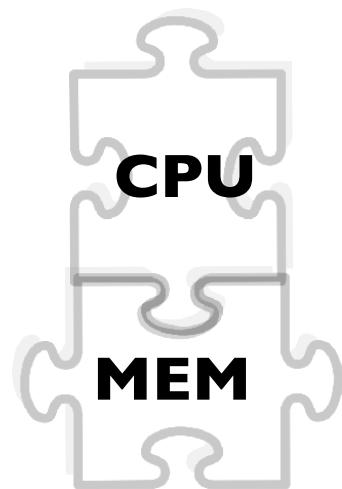
Goal I

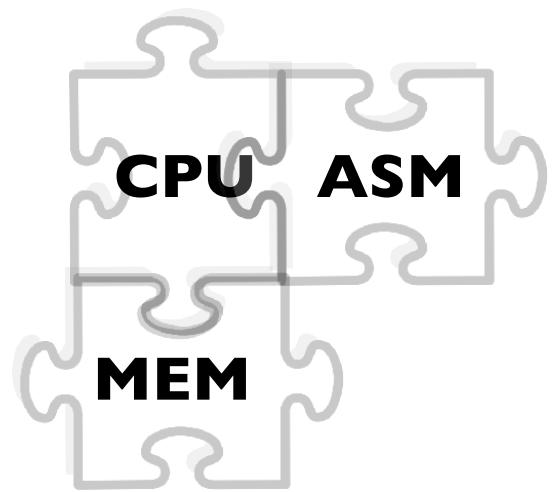
Understand How Computers Work

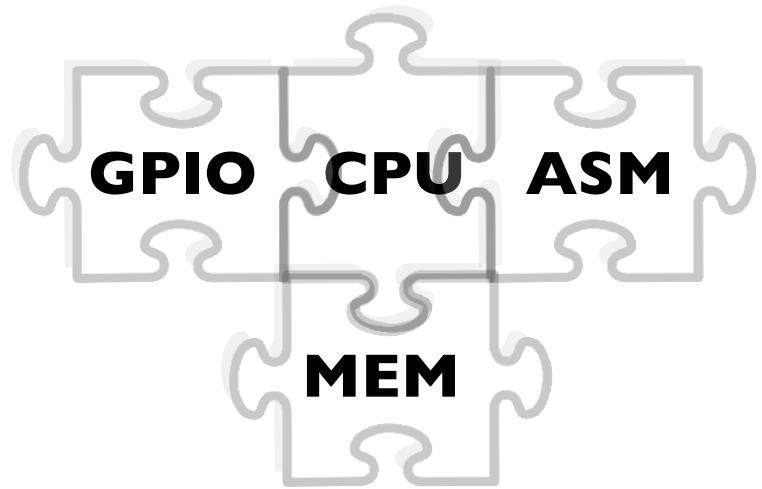
Computer Systems from the Ground Up

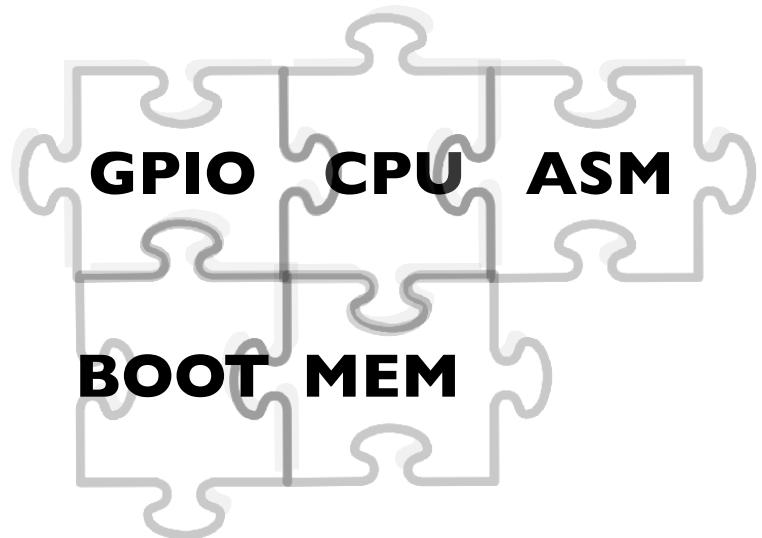


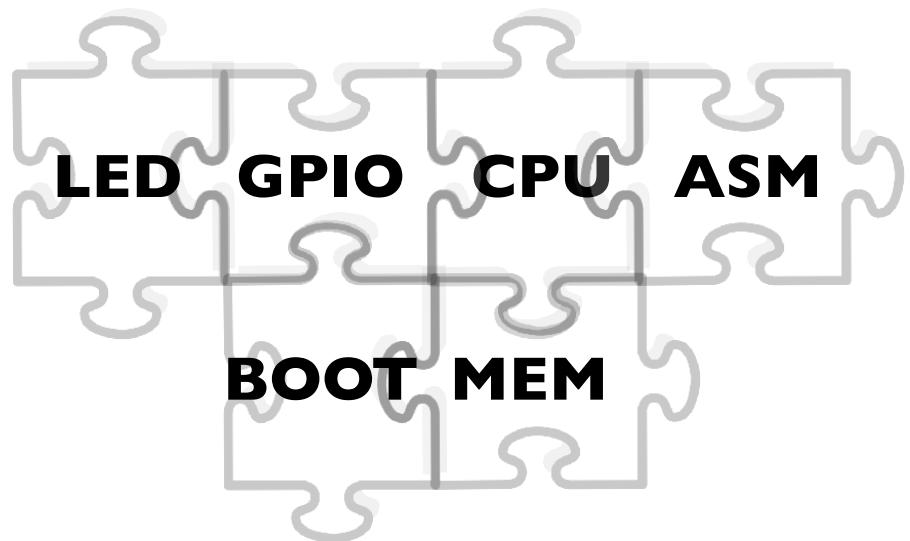
Bare Metal on the Raspberry Pi

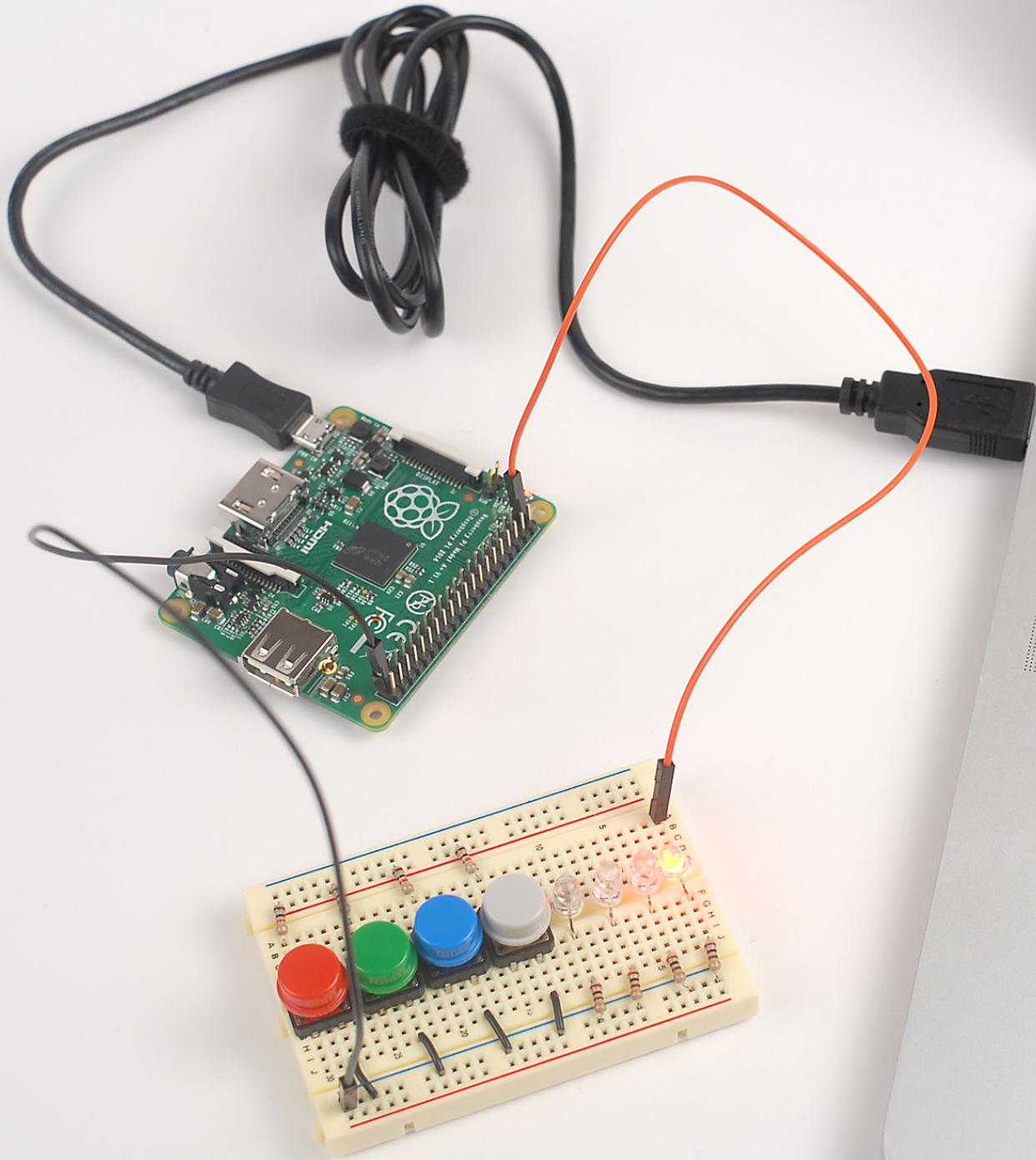


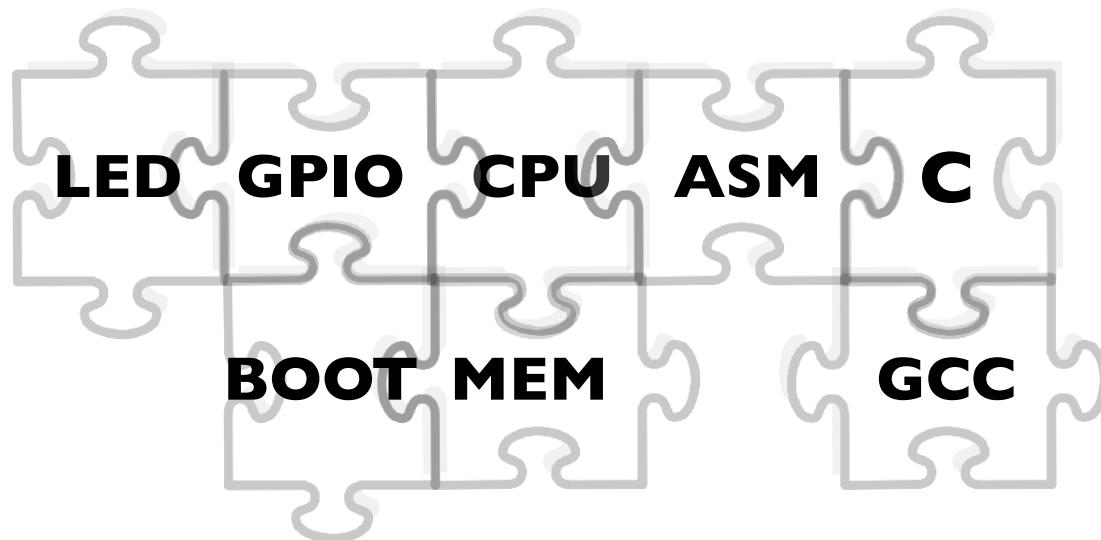


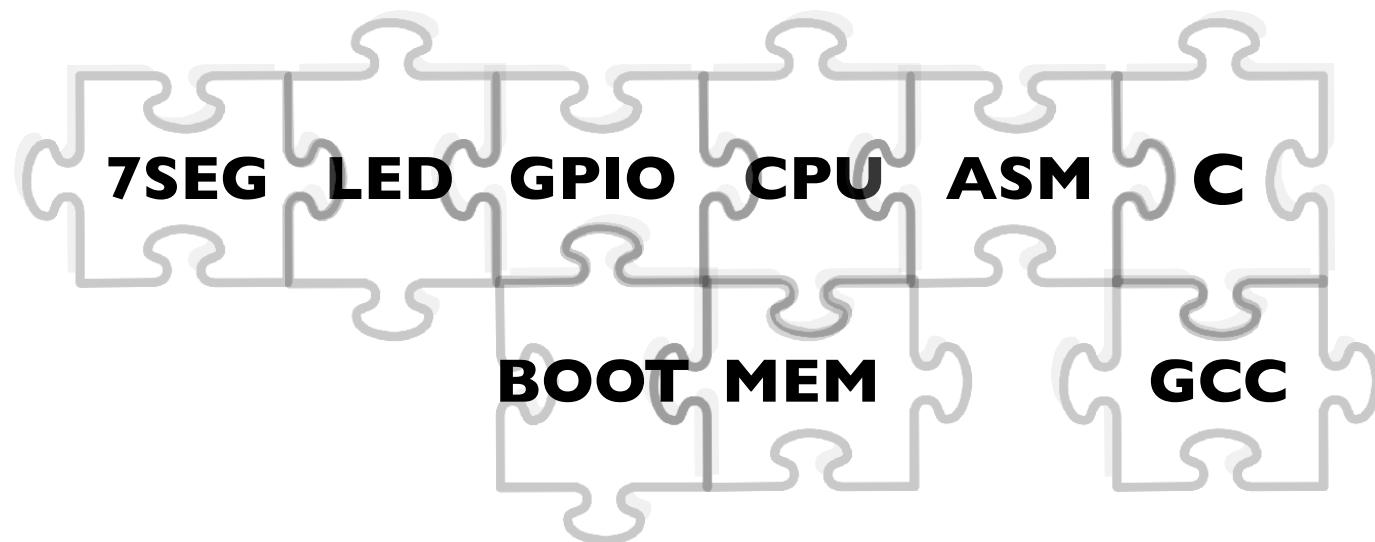


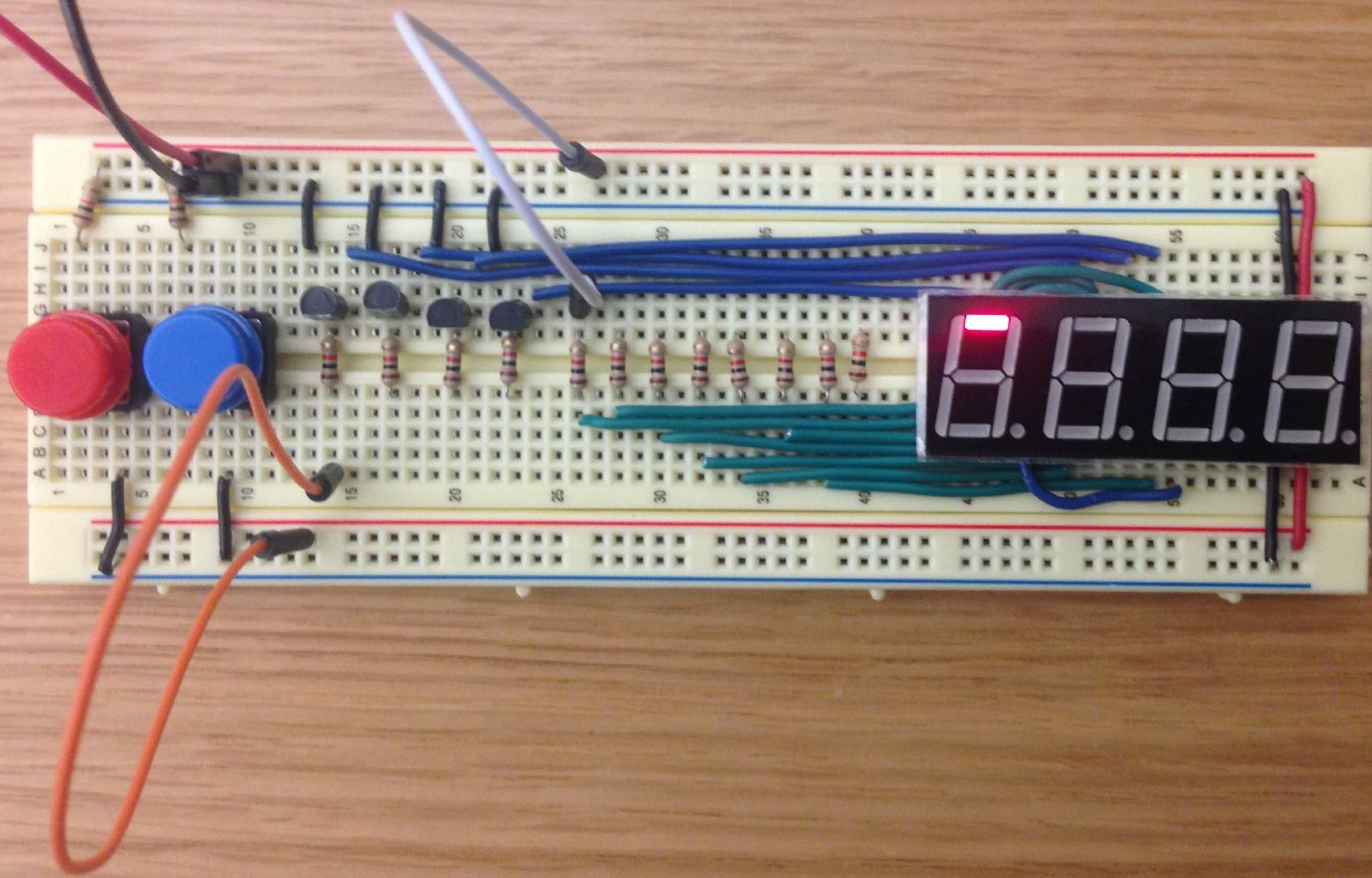












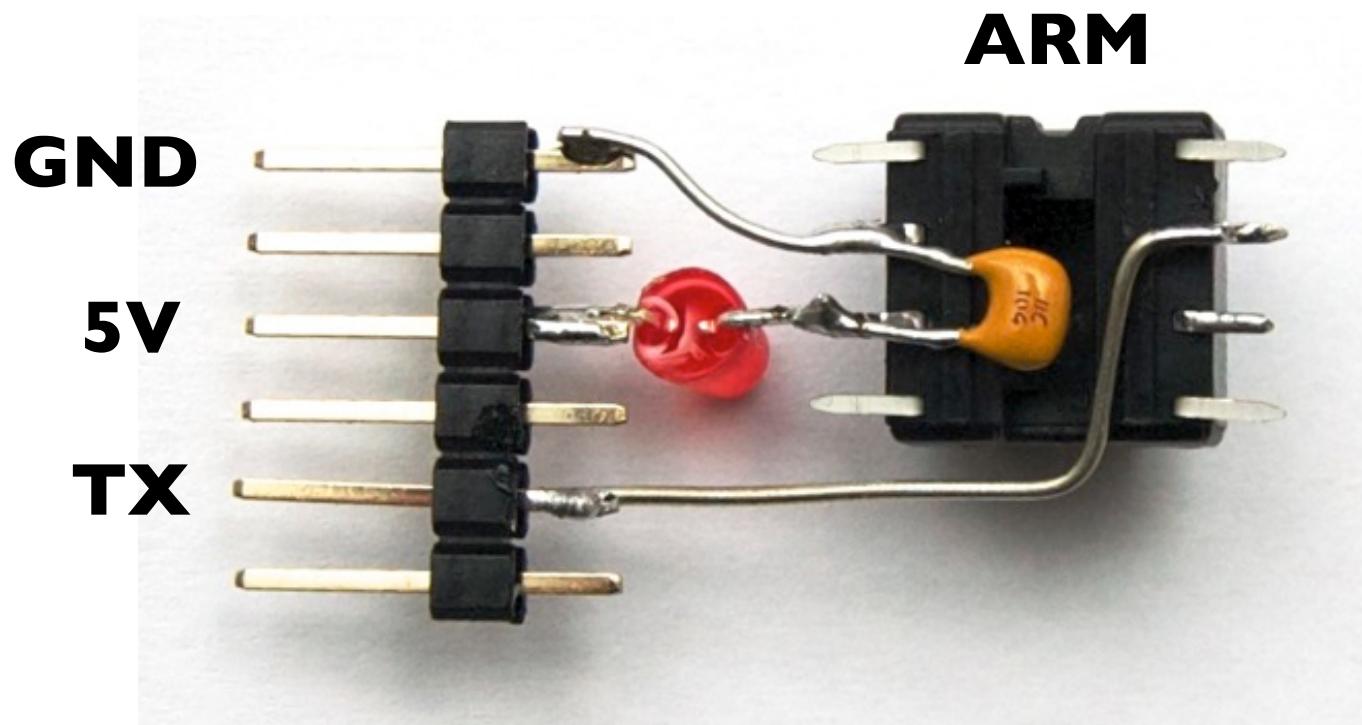
Simple and Clear

Functionality

is

Beautiful

Jean-Claude Wippler



www.jeelabs.org

```

/* match: search for regexp anywhere in text */
int match(char *regexp, char *text) {
    if (regexp[0] == '^')
        return matchhere(regexp+1, text);
    do { /* must look even if string is empty */
        if (matchhere(regexp, text))
            return 1;
    } while (*text++ != '\0');
    return 0;
}
/* matchhere: search for regexp at beginning of text */
int matchhere(char *regexp, char *text) {
    if (regexp[0] == '\0')
        return 1;
    if (regexp[1] == '*')
        return matchstar(regexp[0], regexp+2, text);
    if (regexp[0] == '$' && regexp[1] == '\0')
        return *text == '\0';
    if (*text != '\0' && (regexp[0] == '.' || regexp[0] == *text))
        return matchhere(regexp+1, text+1);
    return 0;
}
/* matchstar: search for c*regexp at beginning of text*/
int matchstar(int c, char *regexp, char *text) {
    do { /* a * matches zero or more instances */
        if (matchhere(regexp, text))
            return 1;
    } while (*text != '\0' && (*text++ == c || c == '.' ));
    return 0;
}

```

Ken Thompson

Regular Expression

ch

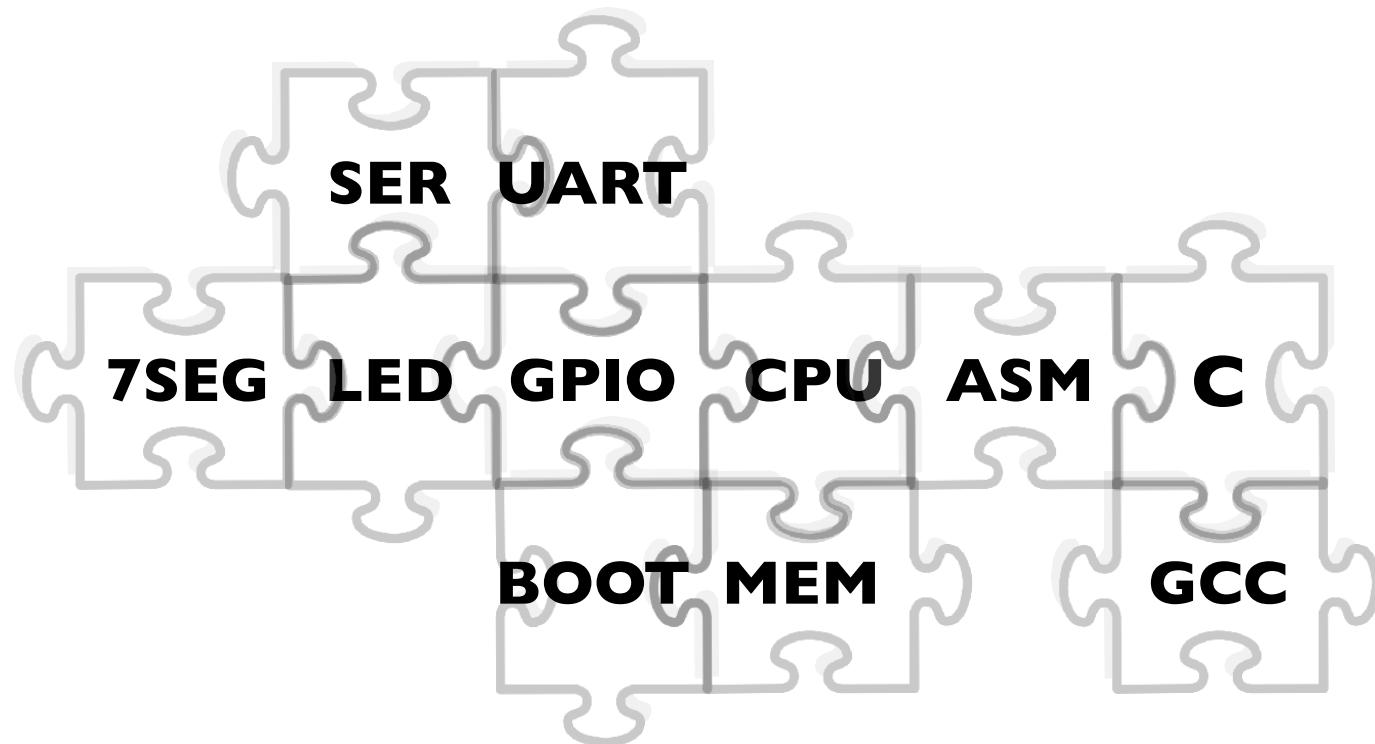
.

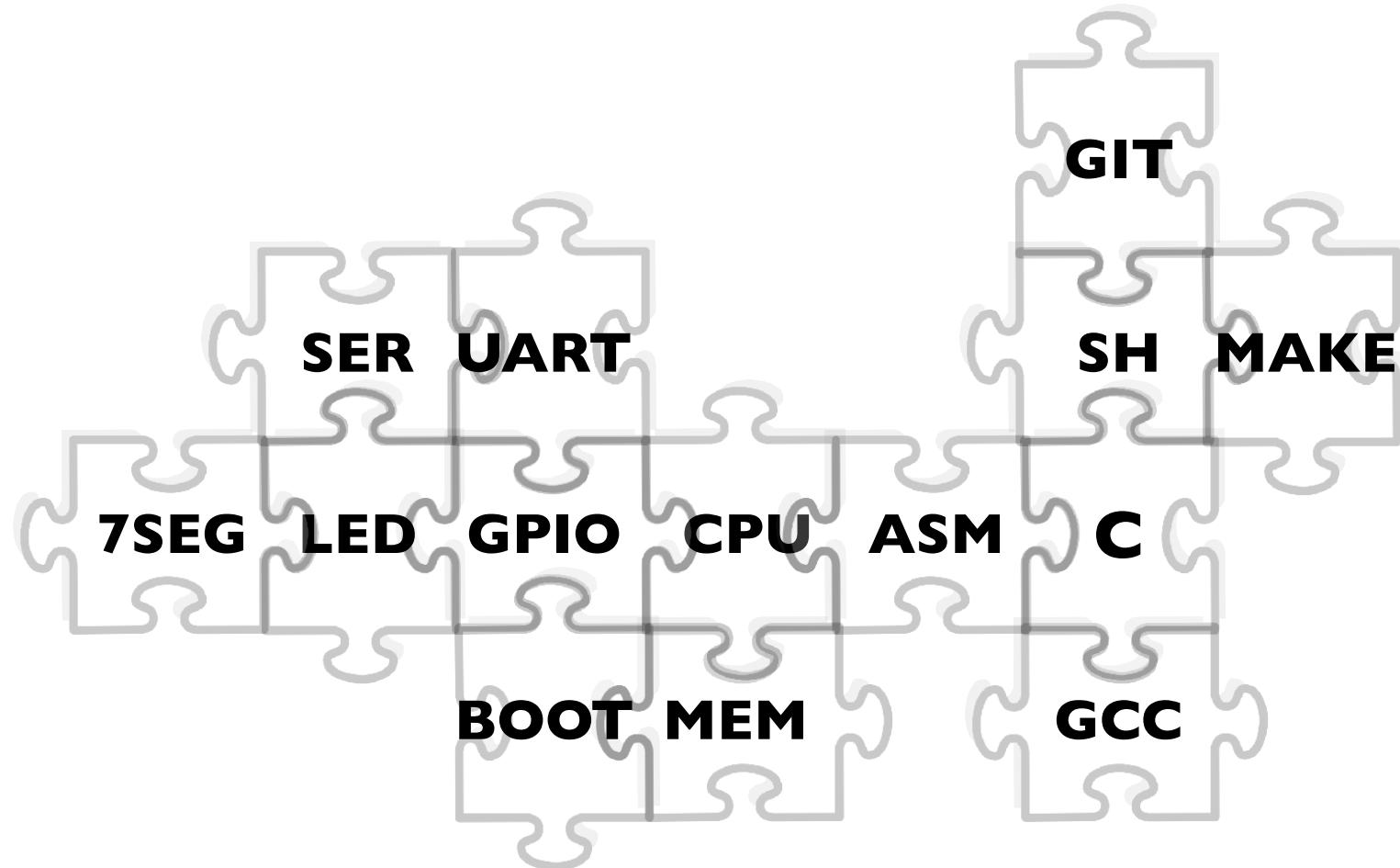
*

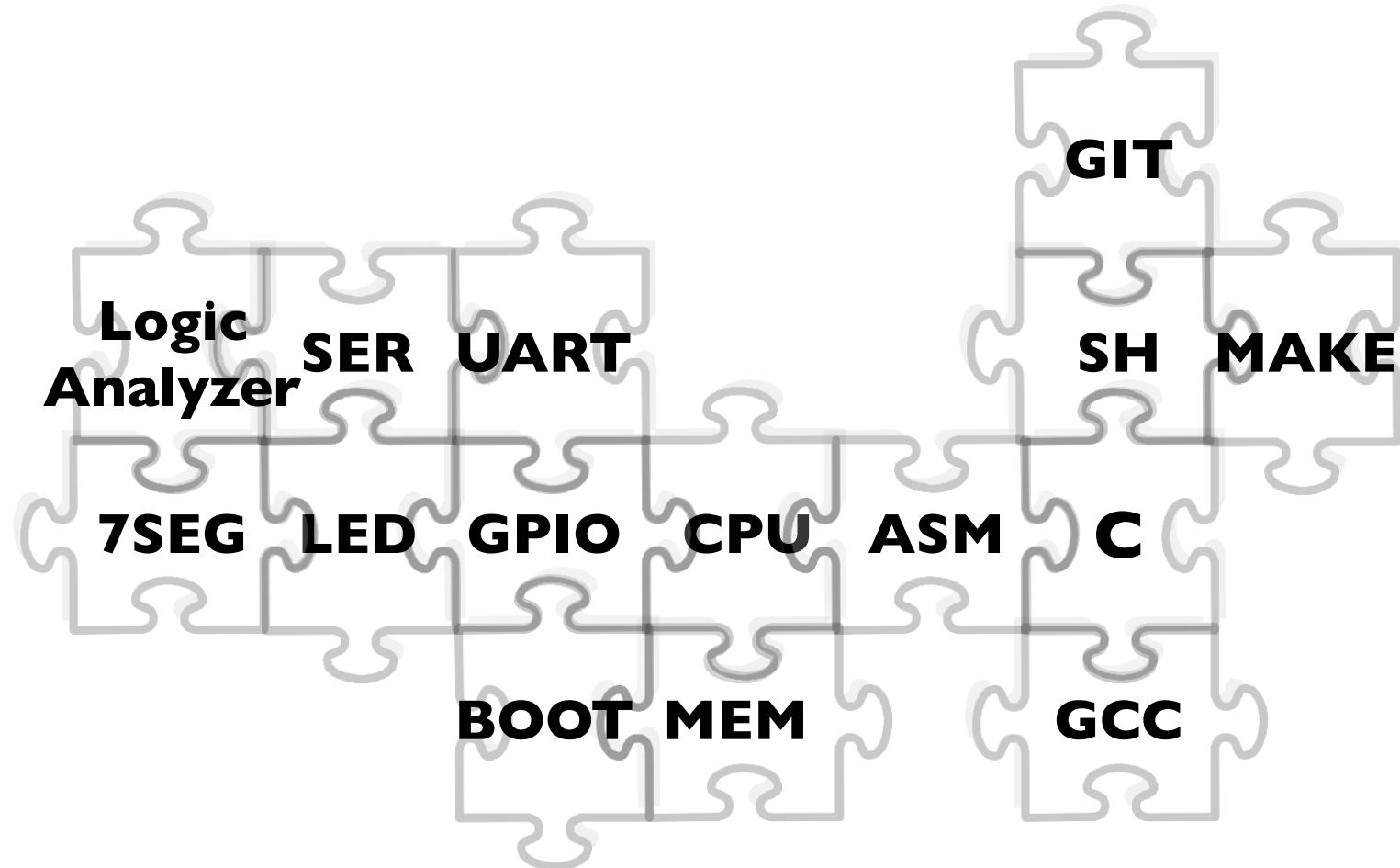
^

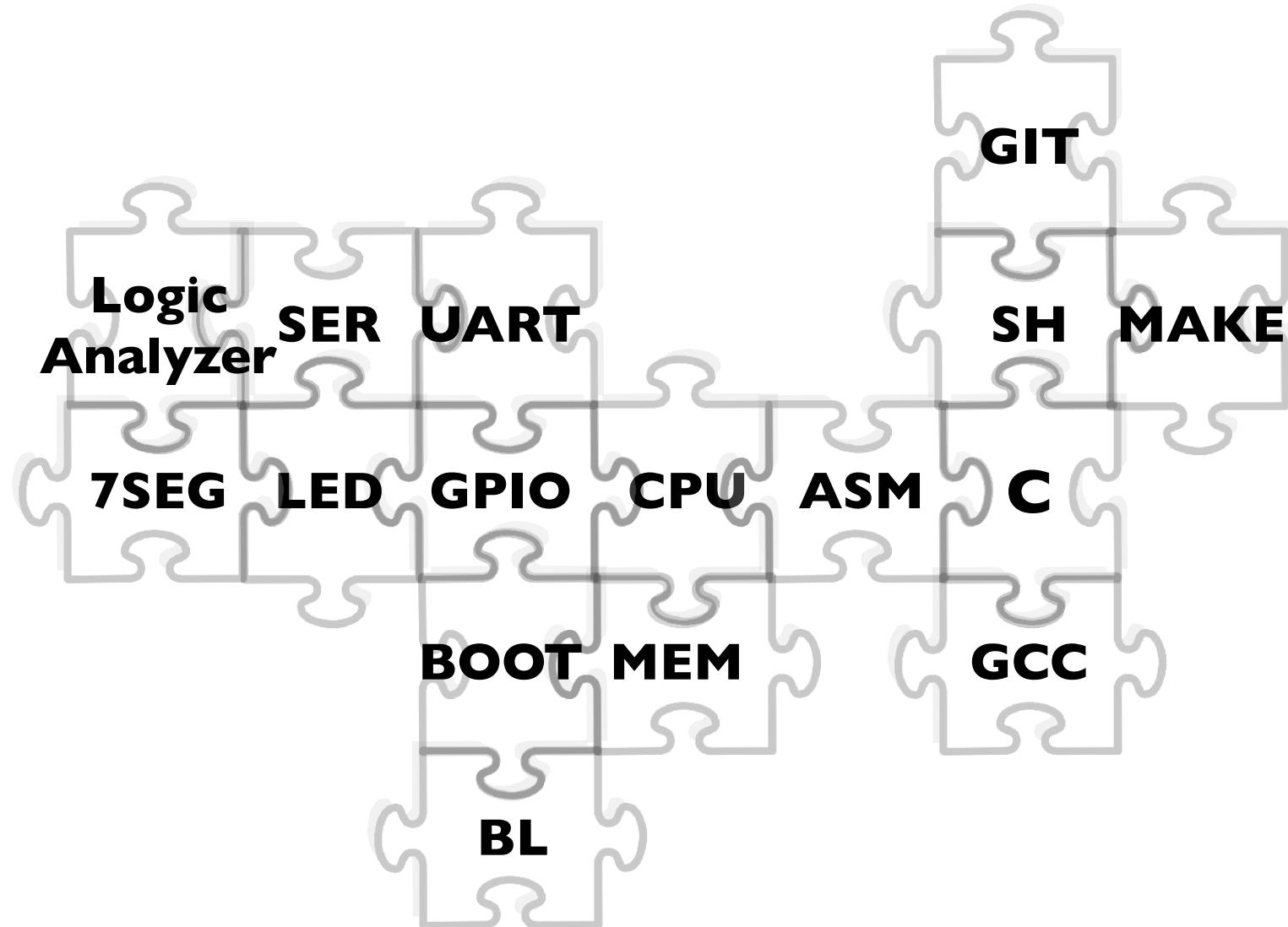
\$

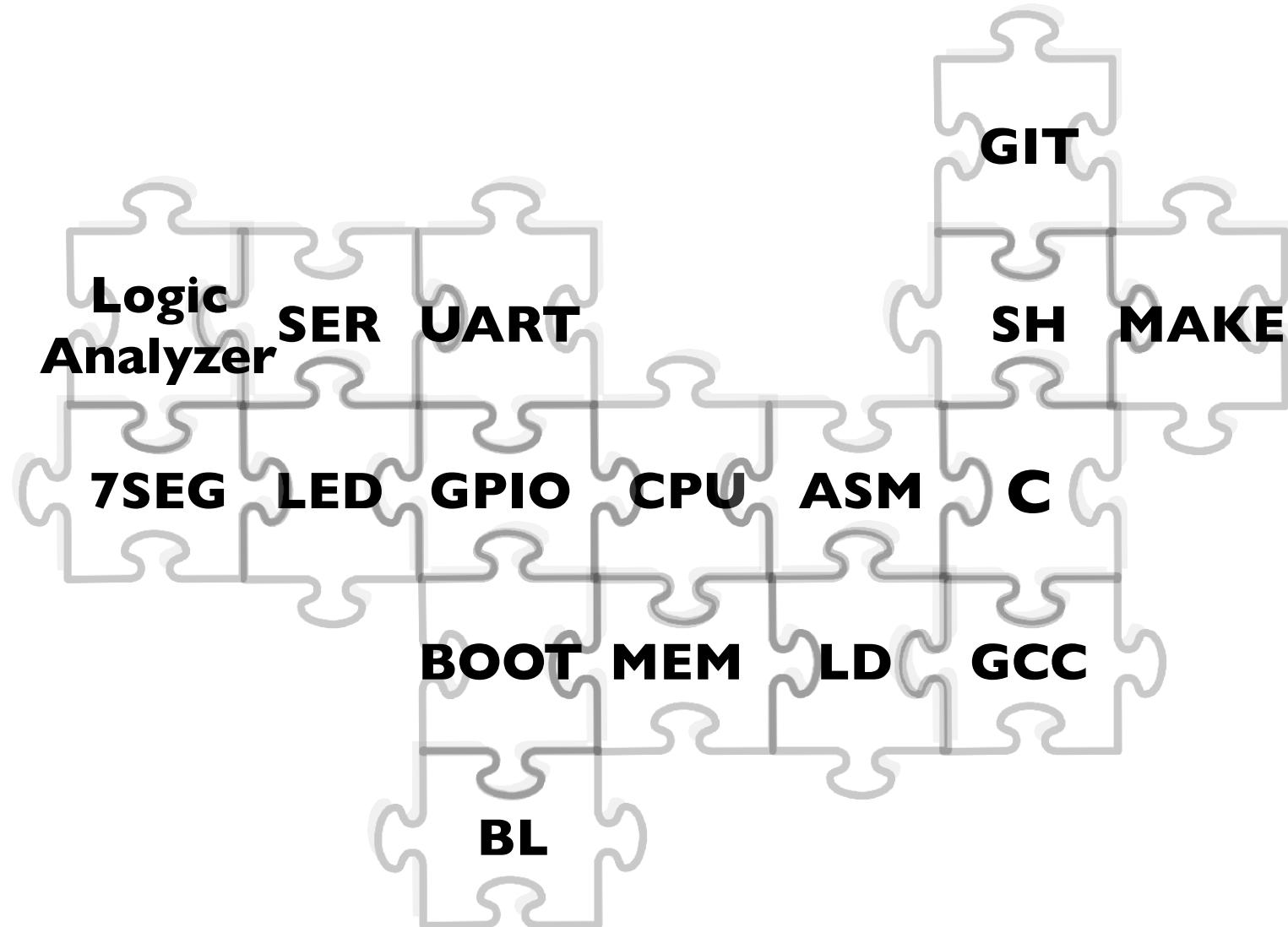
Read Beautiful Code

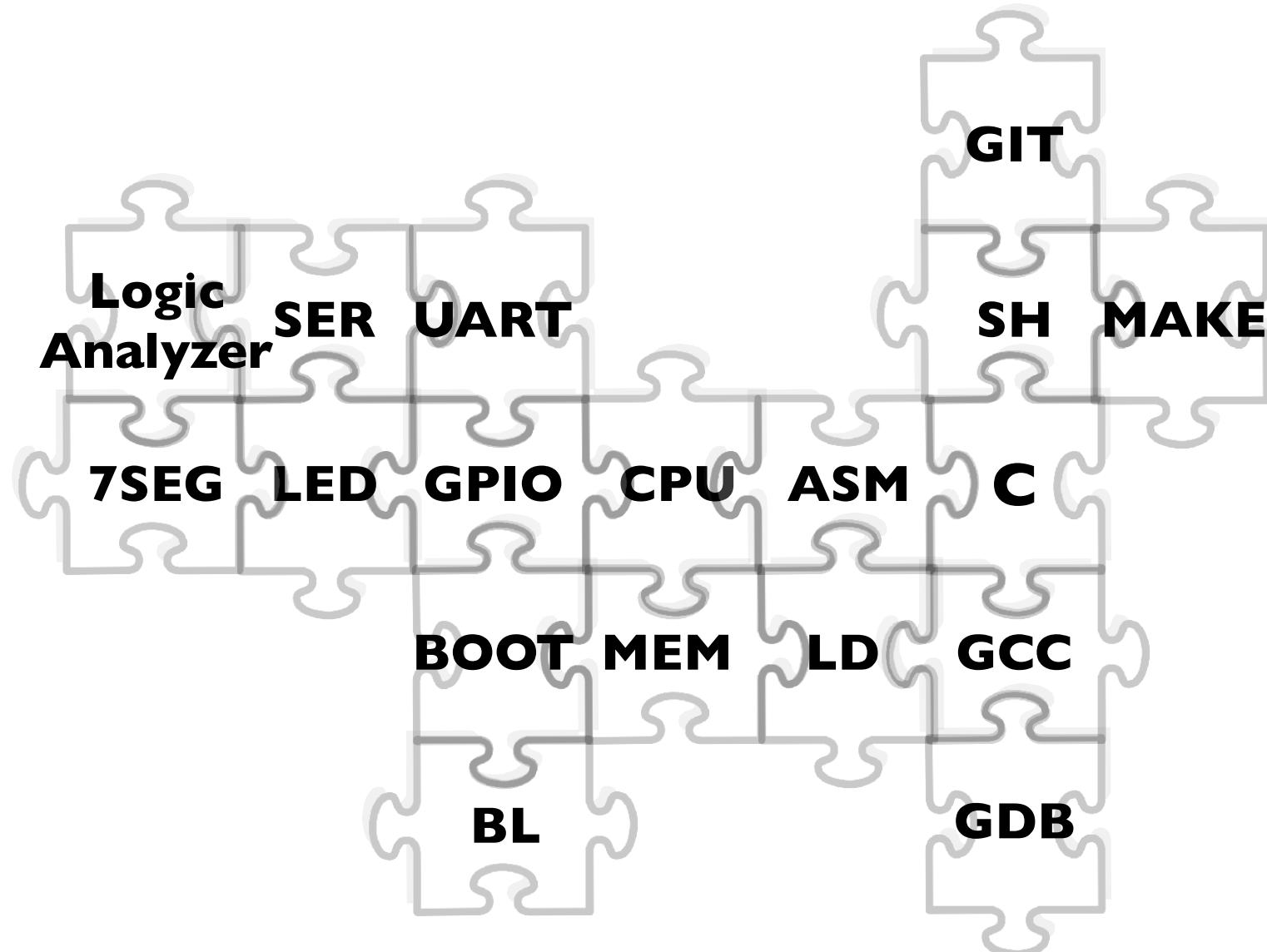


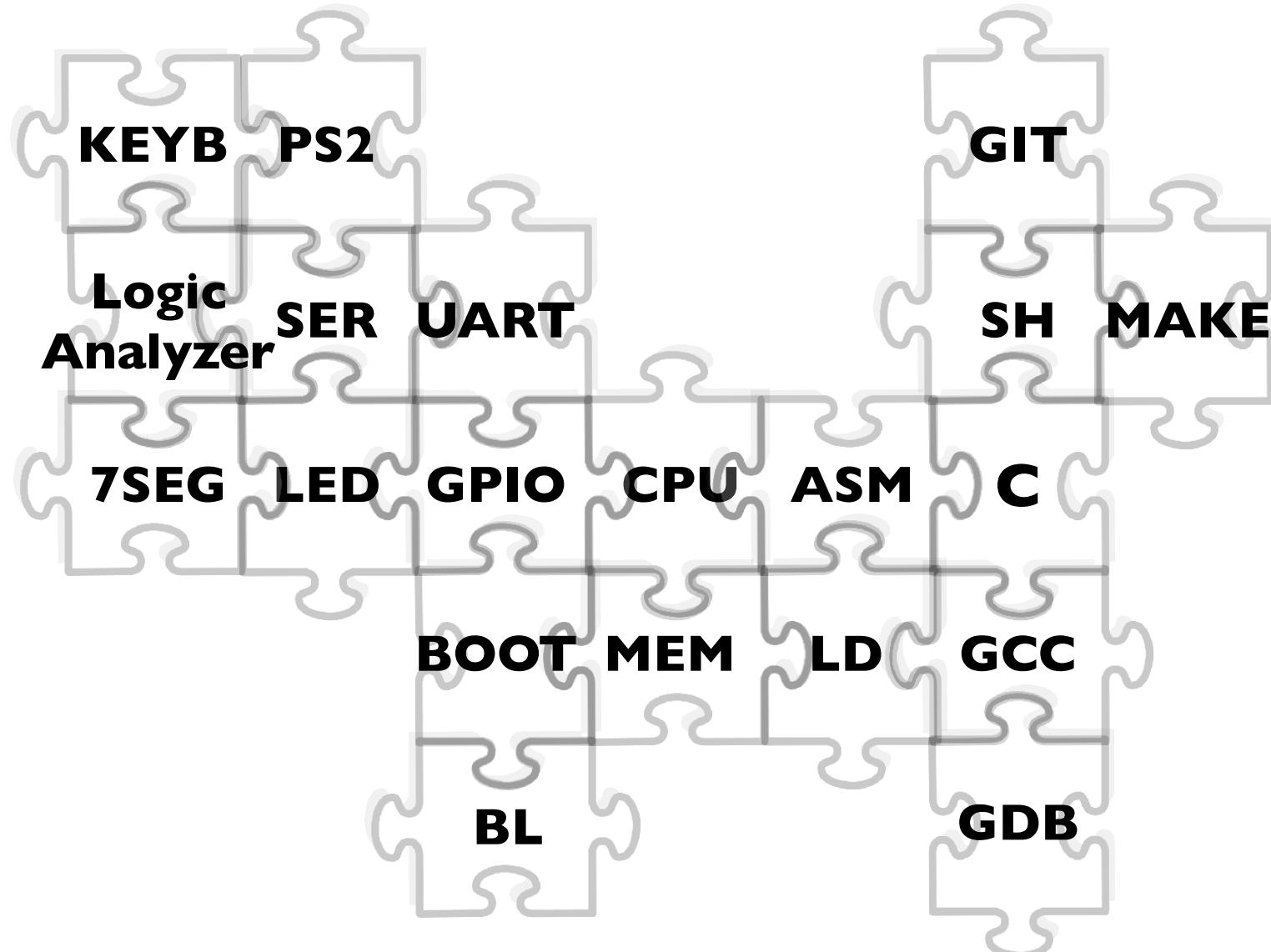


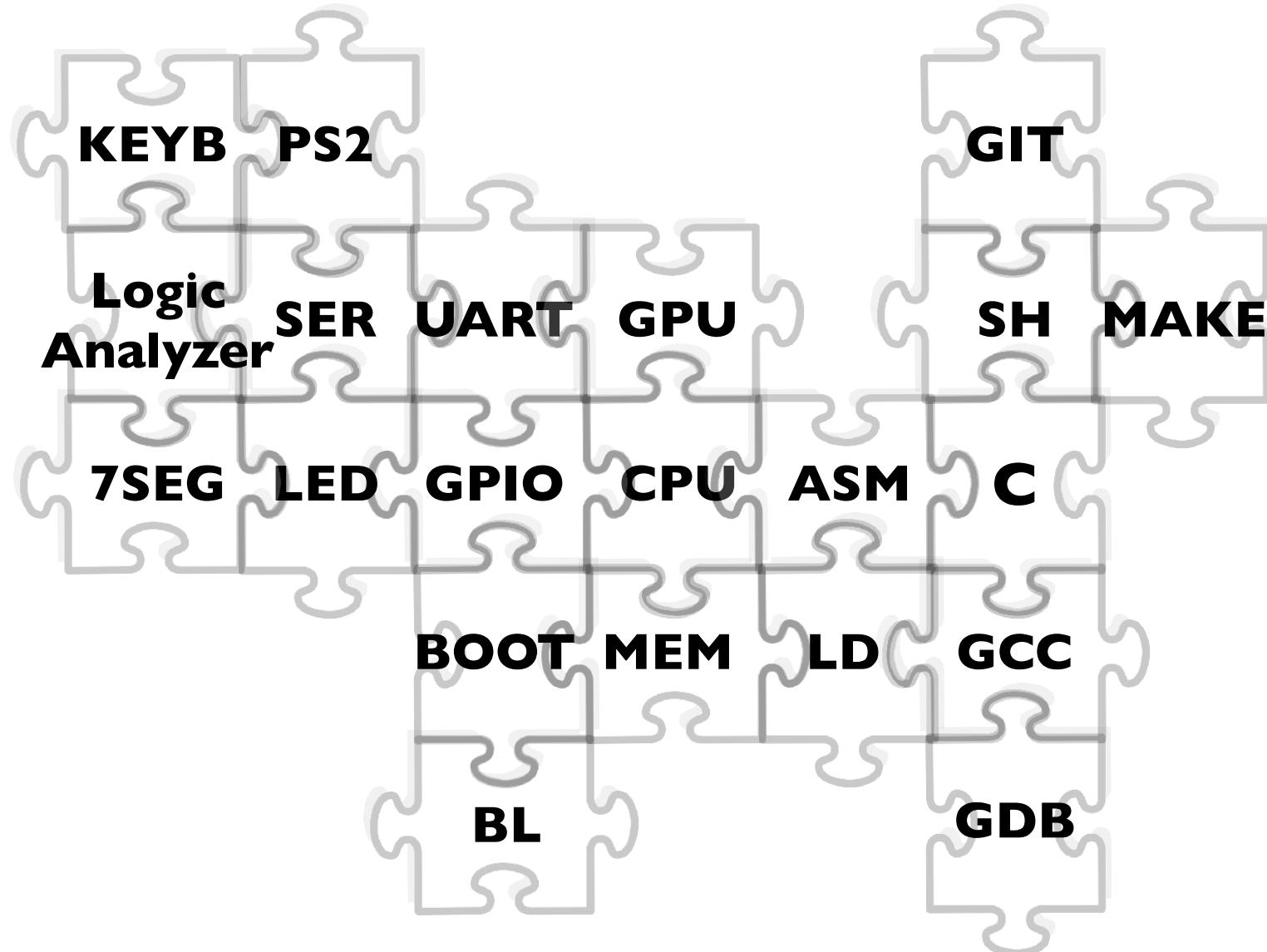




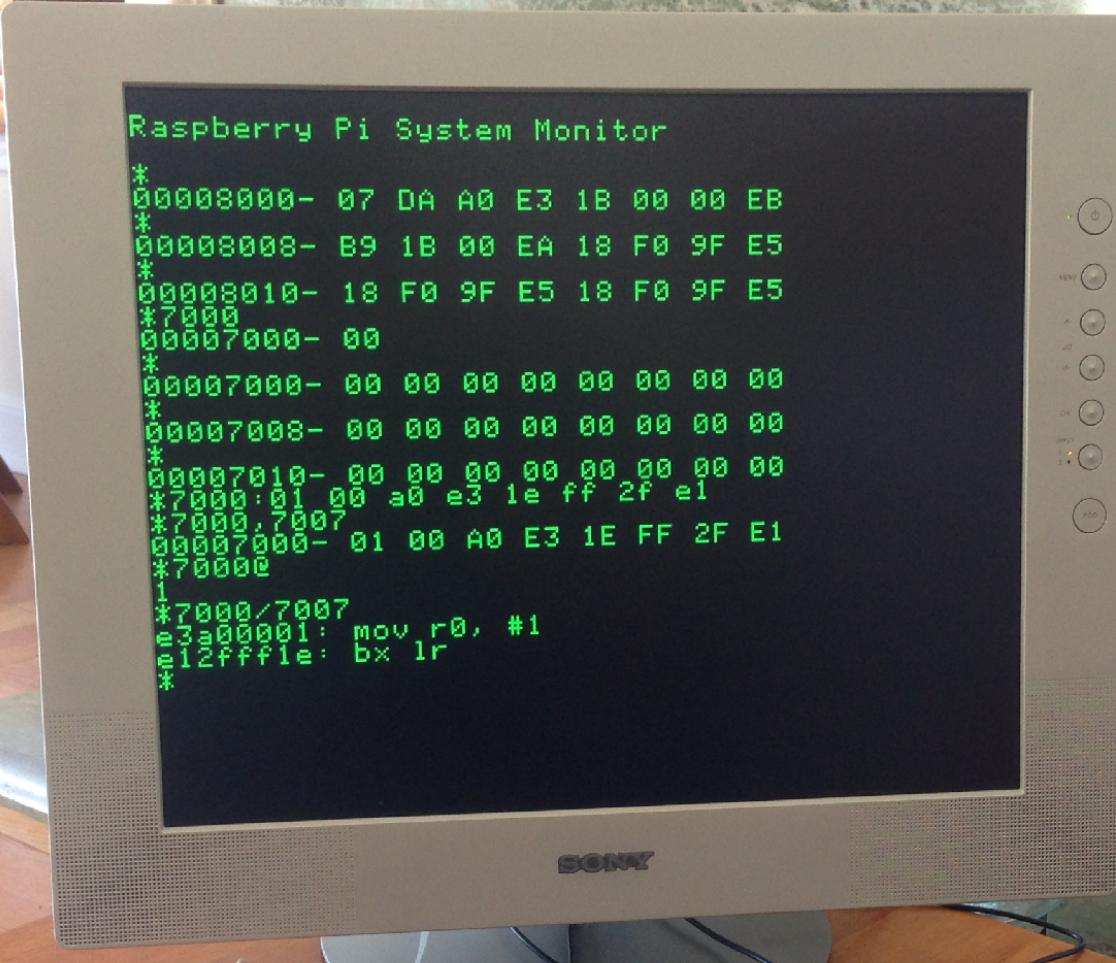
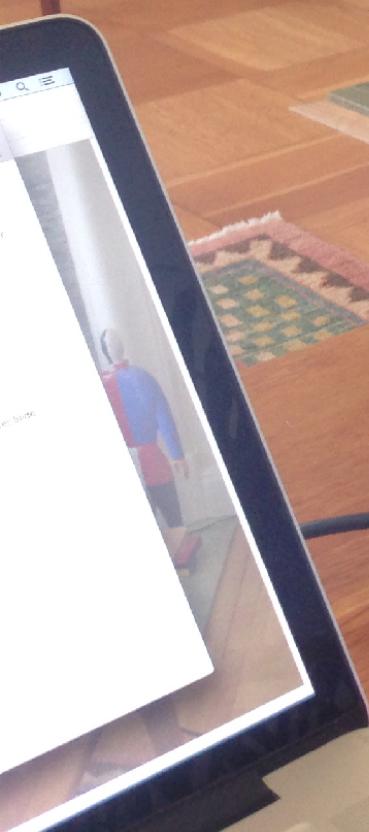






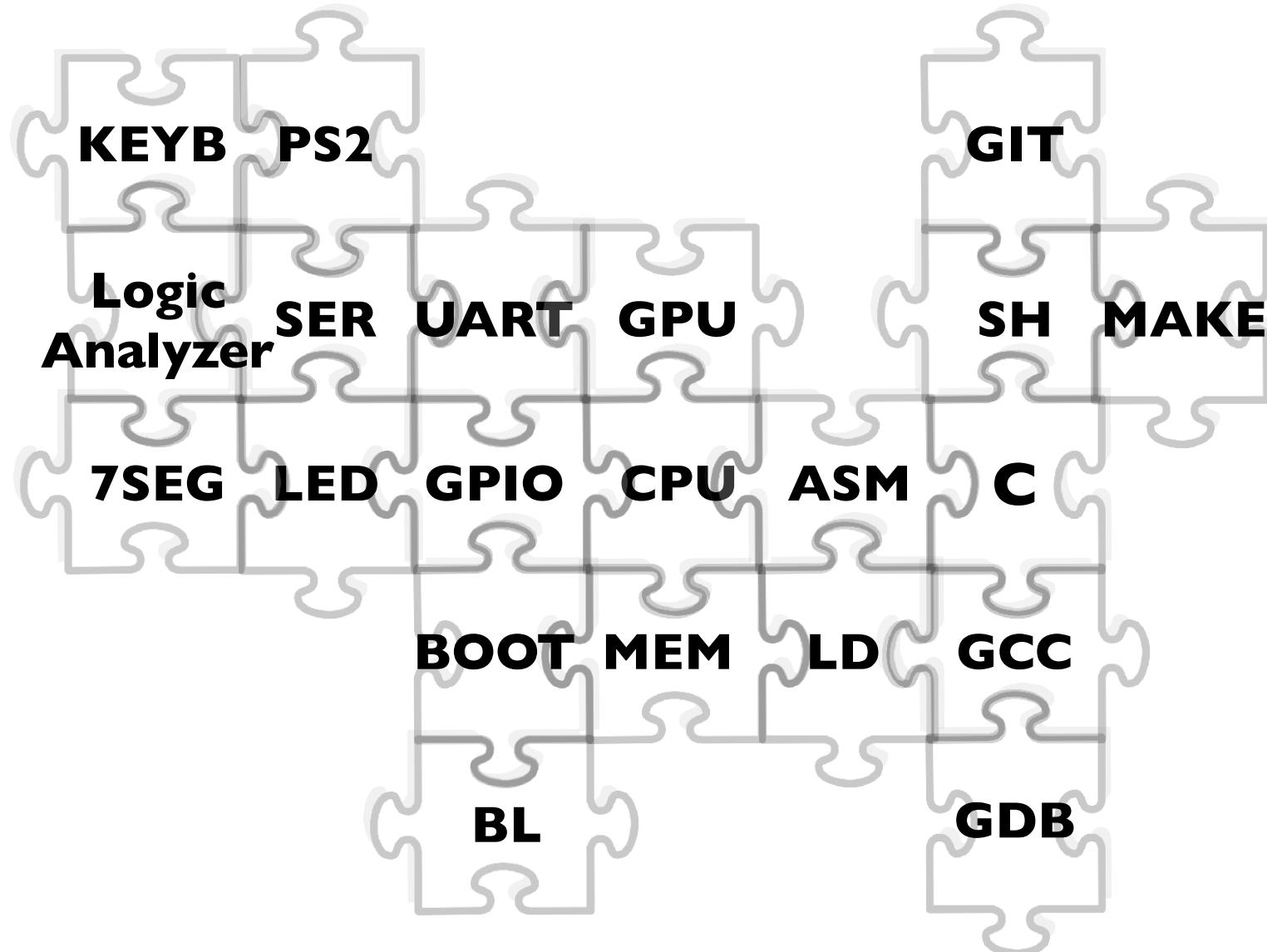


```
Raspberry Pi System Monitor  
*  
00008000- 07 DA A0 E3 1B 00 00 EB  
*  
00008008- B9 1B 00 EA 18 F0 9F E5  
*  
00008010- 18 F0 9F E5 18 F0 9F E5  
*7000  
00007000- 00  
*  
00007000- 00 00 00 00 00 00 00 00  
*  
00007008- 00 00 00 00 00 00 00 00  
*  
00007010- 00 00 00 00 00 00 00 00  
*7000:01 00 a0 e3 1e ff 2f e1  
*7000,7007  
00007000- 01 00 A0 E3 1E FF 2F E1  
*70000  
1  
*7000/7007  
e3a00001: mov r0, #1  
e12fffffe: bx lr  
**
```

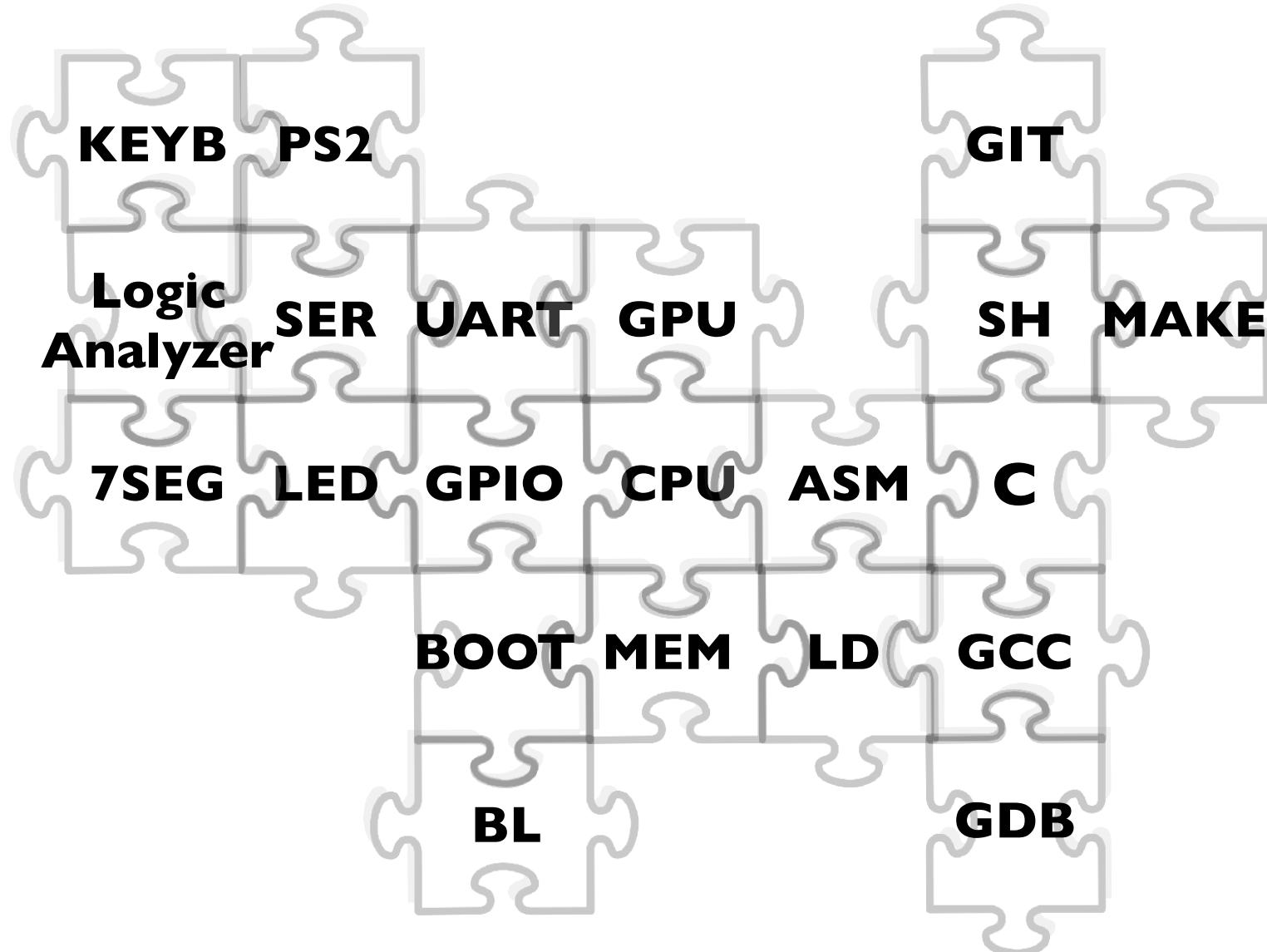


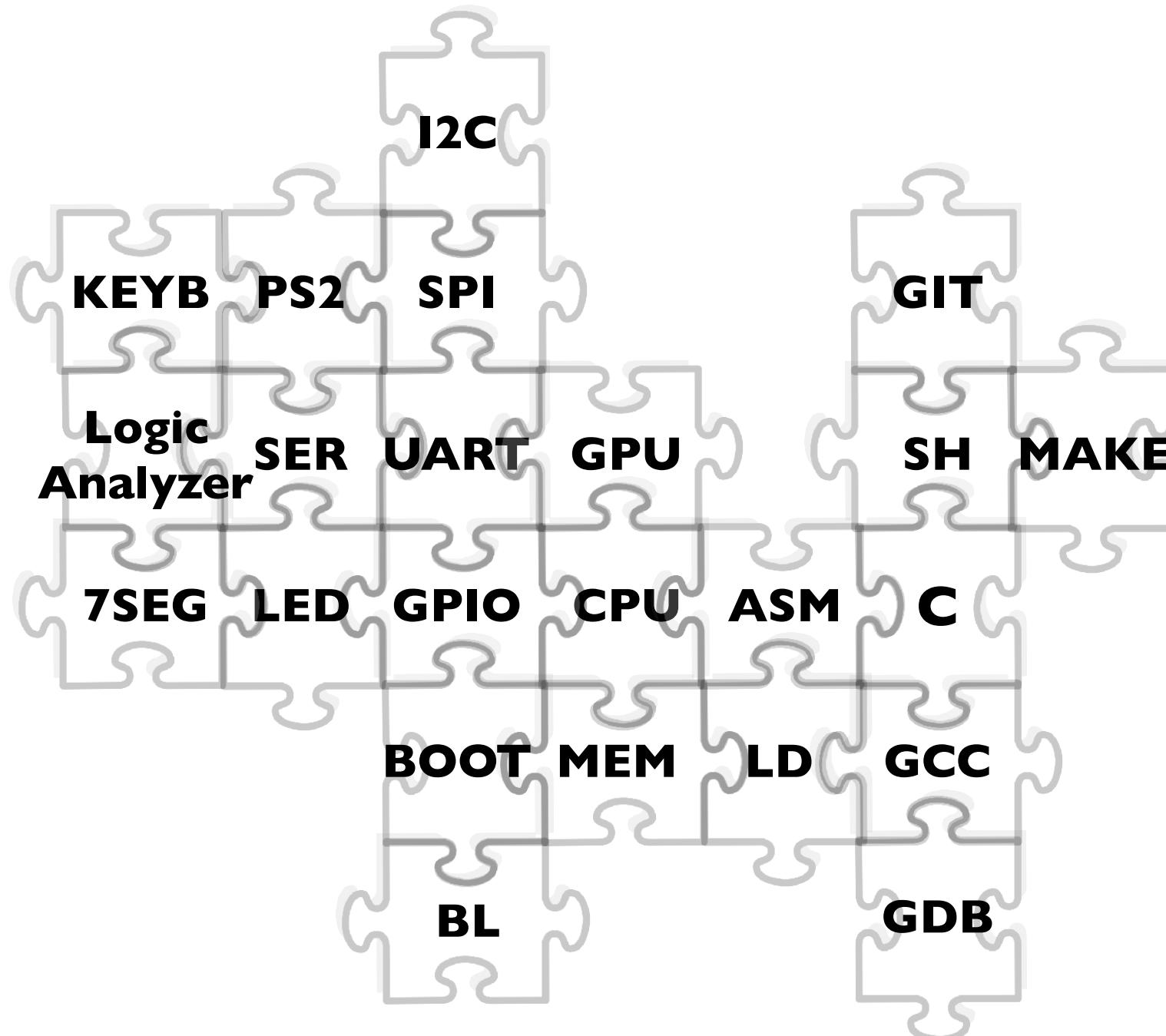
* *
* APPLE II *
* SYSTEM MONITOR *
* *
* COPYRIGHT 1977 BY *
* APPLE COMPUTER, INC. *
* *
* ALL RIGHTS RESERVED *
* *
* S. WOZNIAK *
* A. BAUM *
* *

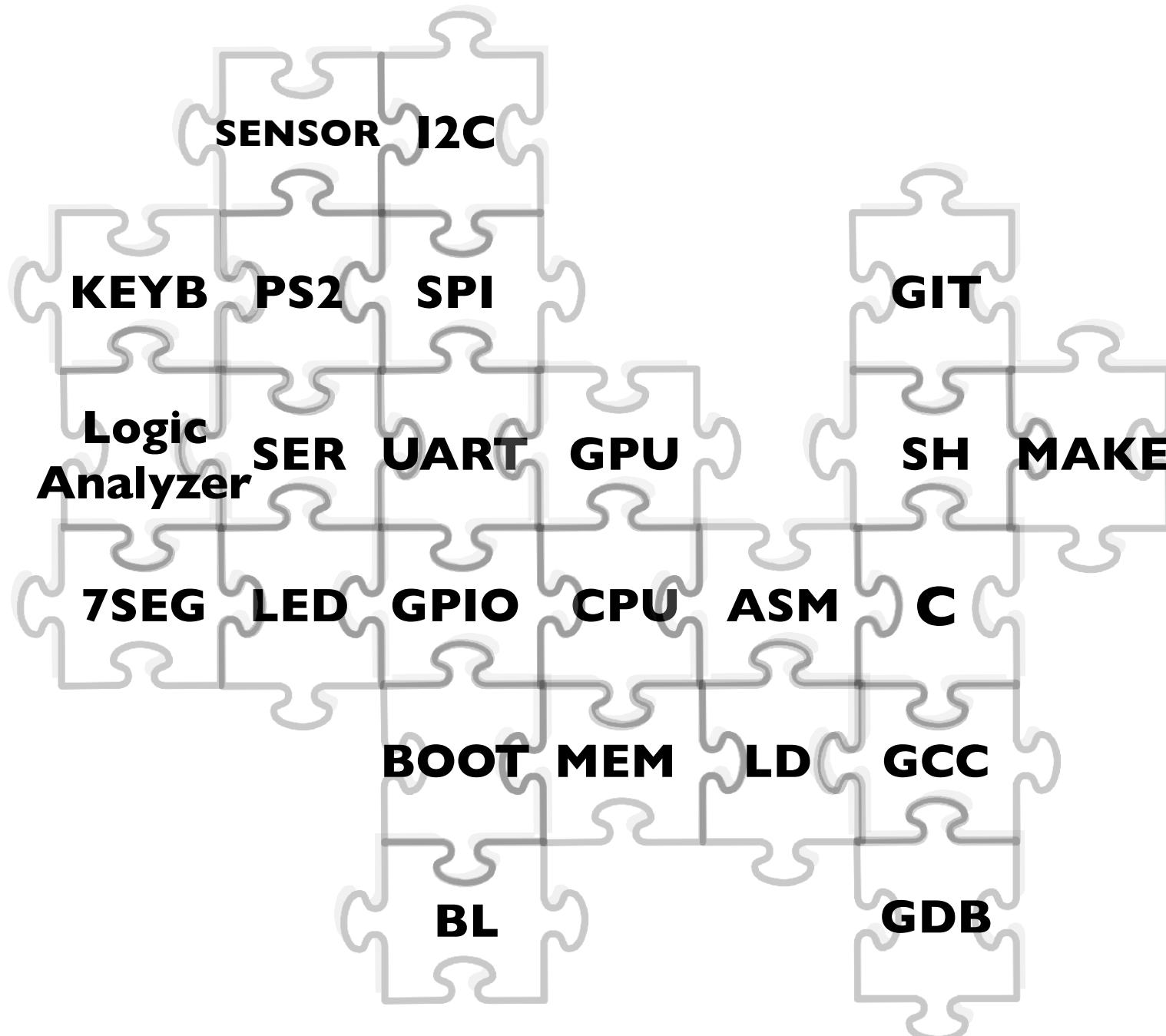
	TITLE	"APPLE II SYSTEM MONITOR"
LOC0	EPZ	\$00
LOC1	EPZ	\$01
WNDLFT	EPZ	\$20
WNDWDTH	EPZ	\$21
WNDTOP	EPZ	\$22
WNDBTM	EPZ	\$23
CH	EPZ	\$24
CV	EPZ	\$25
GBASL	EPZ	\$26
GBASH	EPZ	\$27
BASL	EPZ	\$28
BASH	EPZ	\$29
BAS2L	EPZ	\$2A
BAS2H	EPZ	\$2B
H2	EPZ	\$2C
LMNEM	EPZ	\$2C
RTNL	EPZ	\$2C
V2	EPZ	\$2D
RMNEM	EPZ	\$2D
RTNH	EPZ	\$2D
MASK	EPZ	\$2E

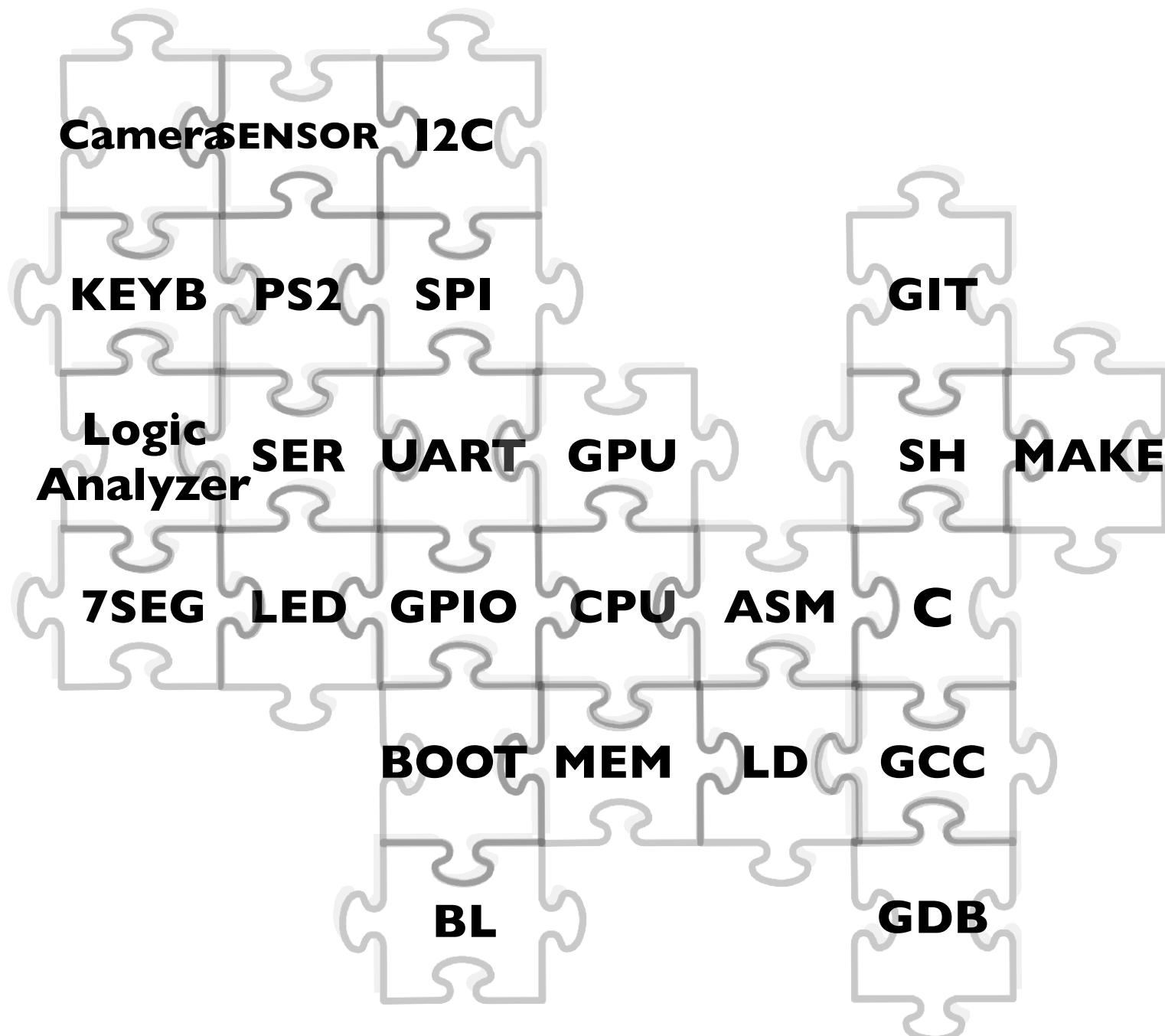


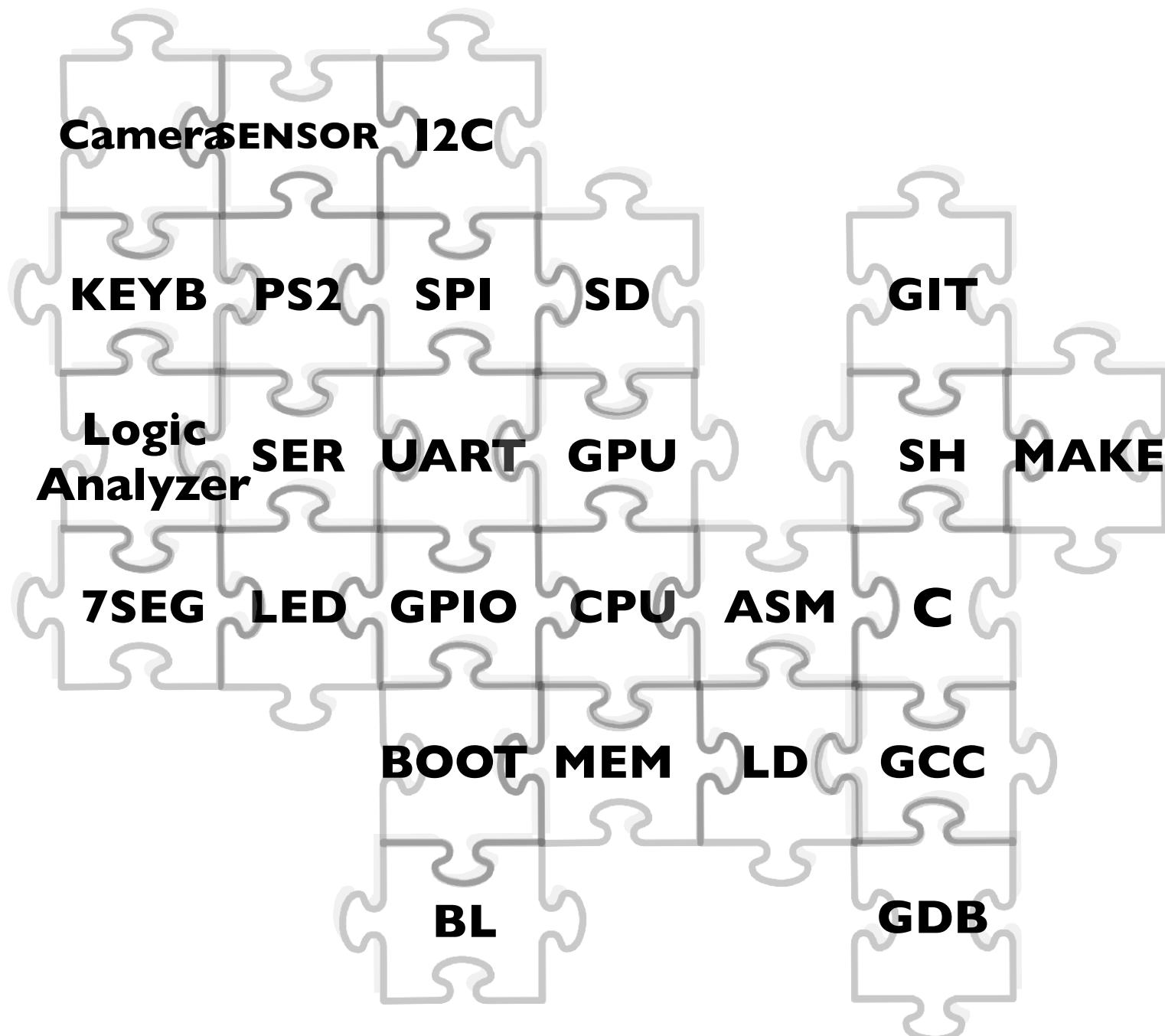
But It Keeps Growing!

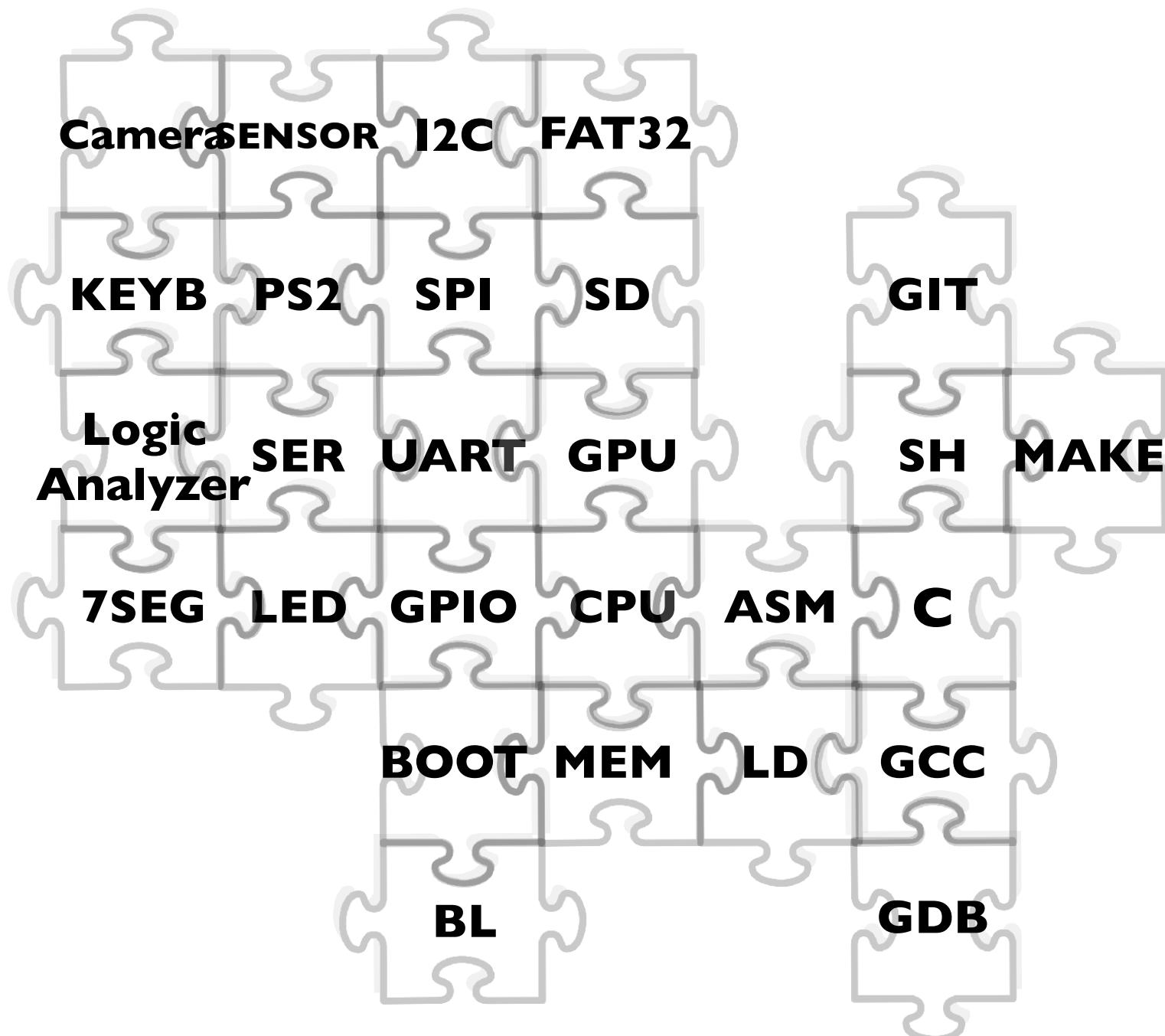


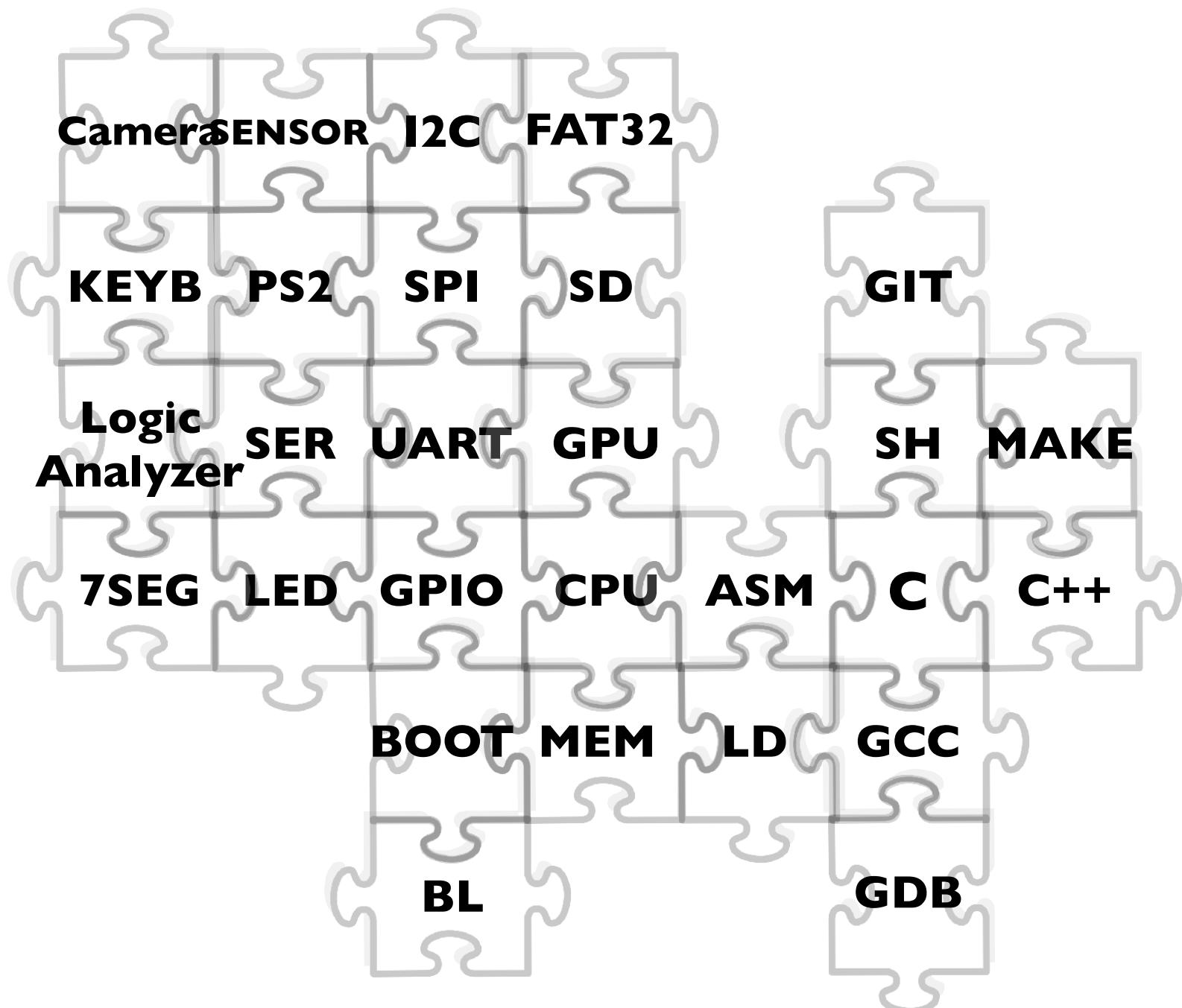


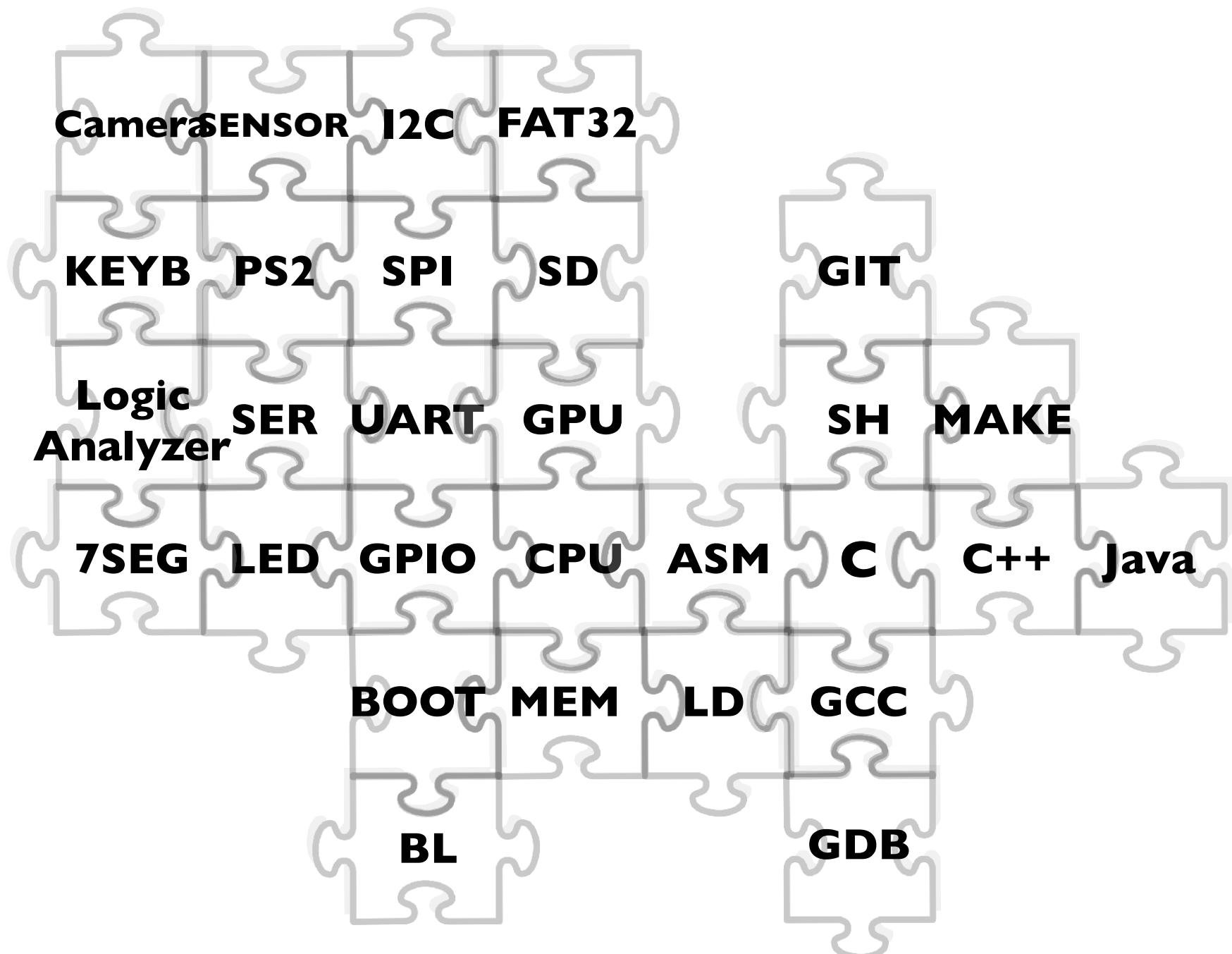


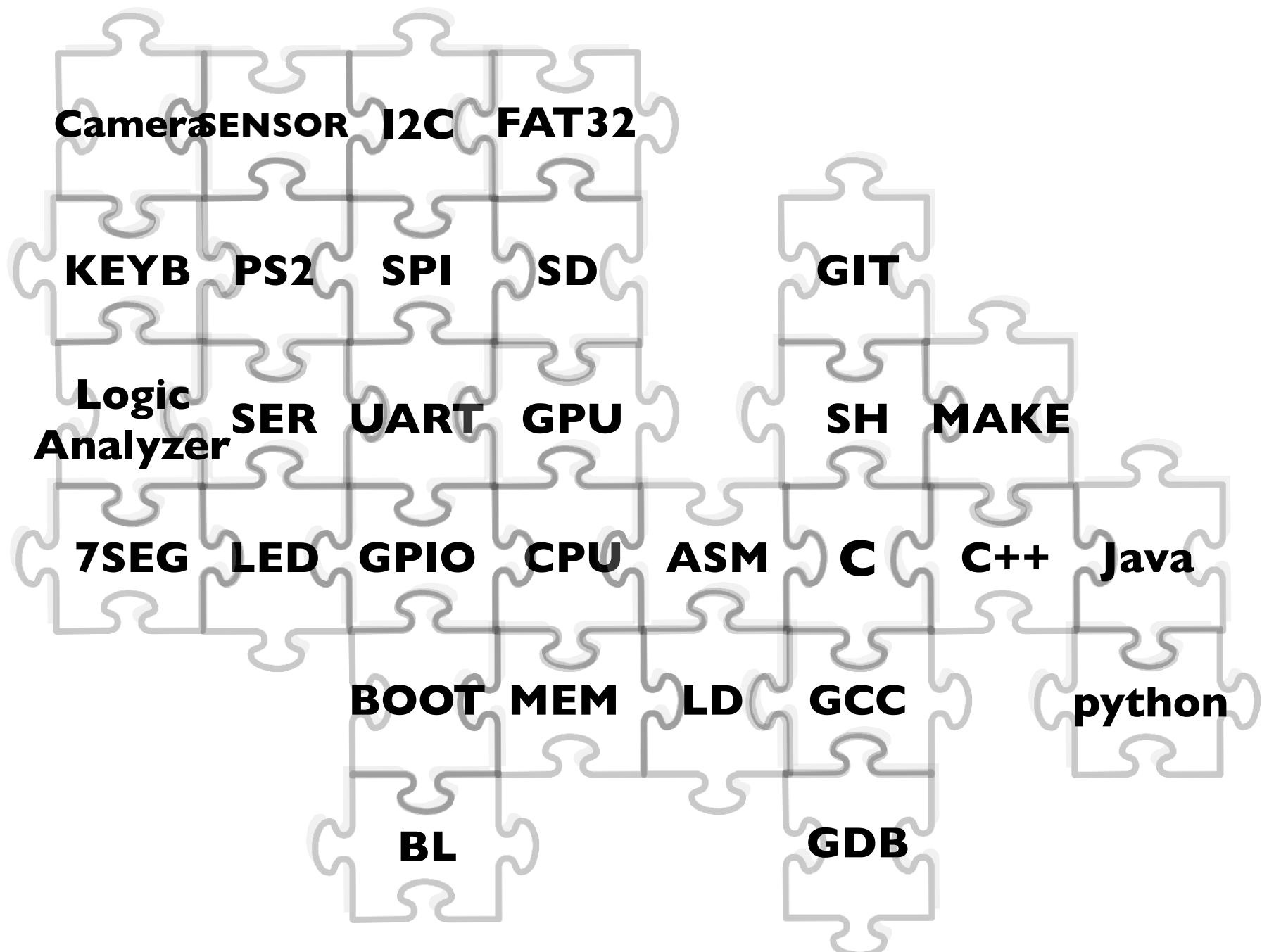






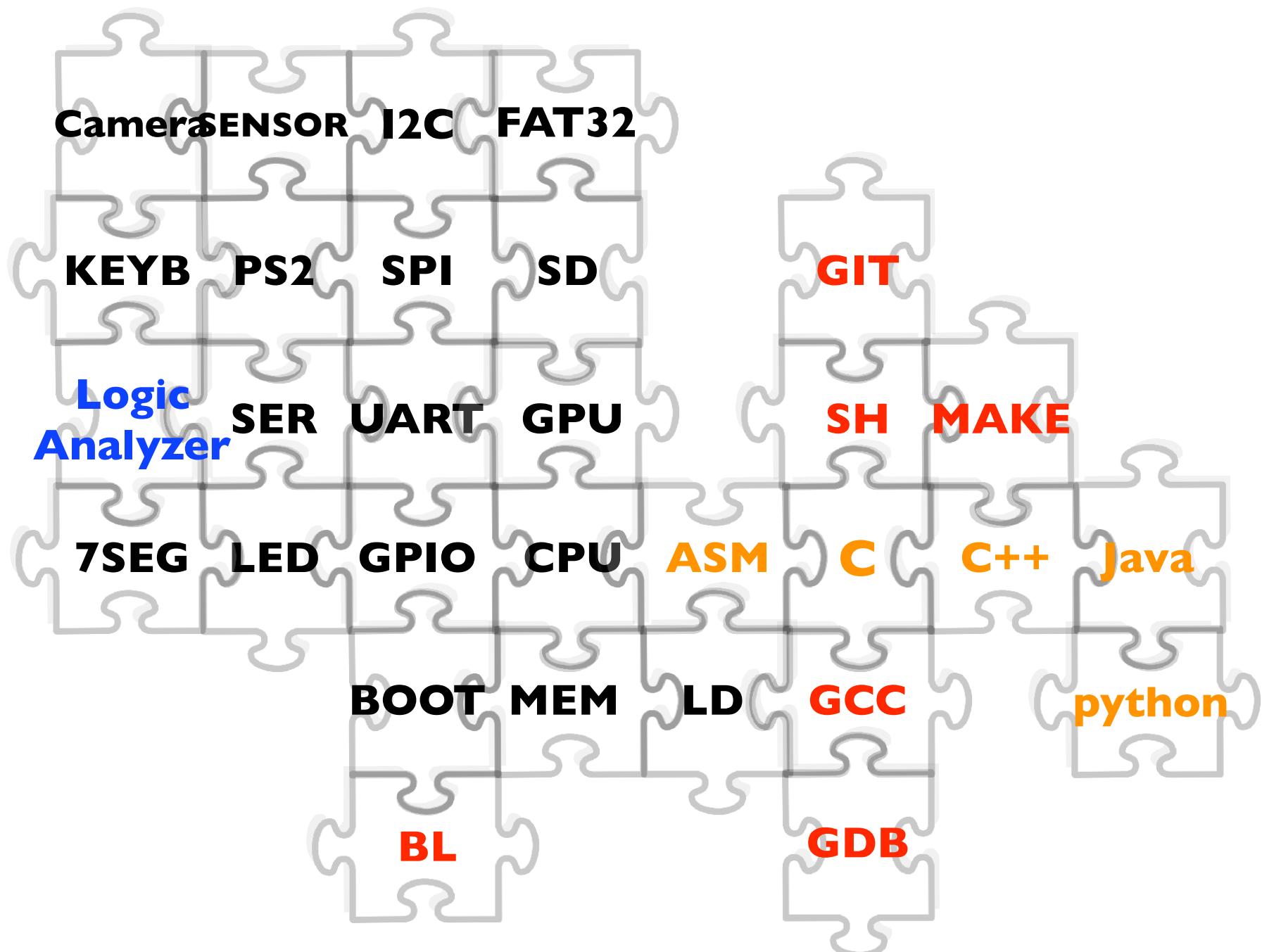


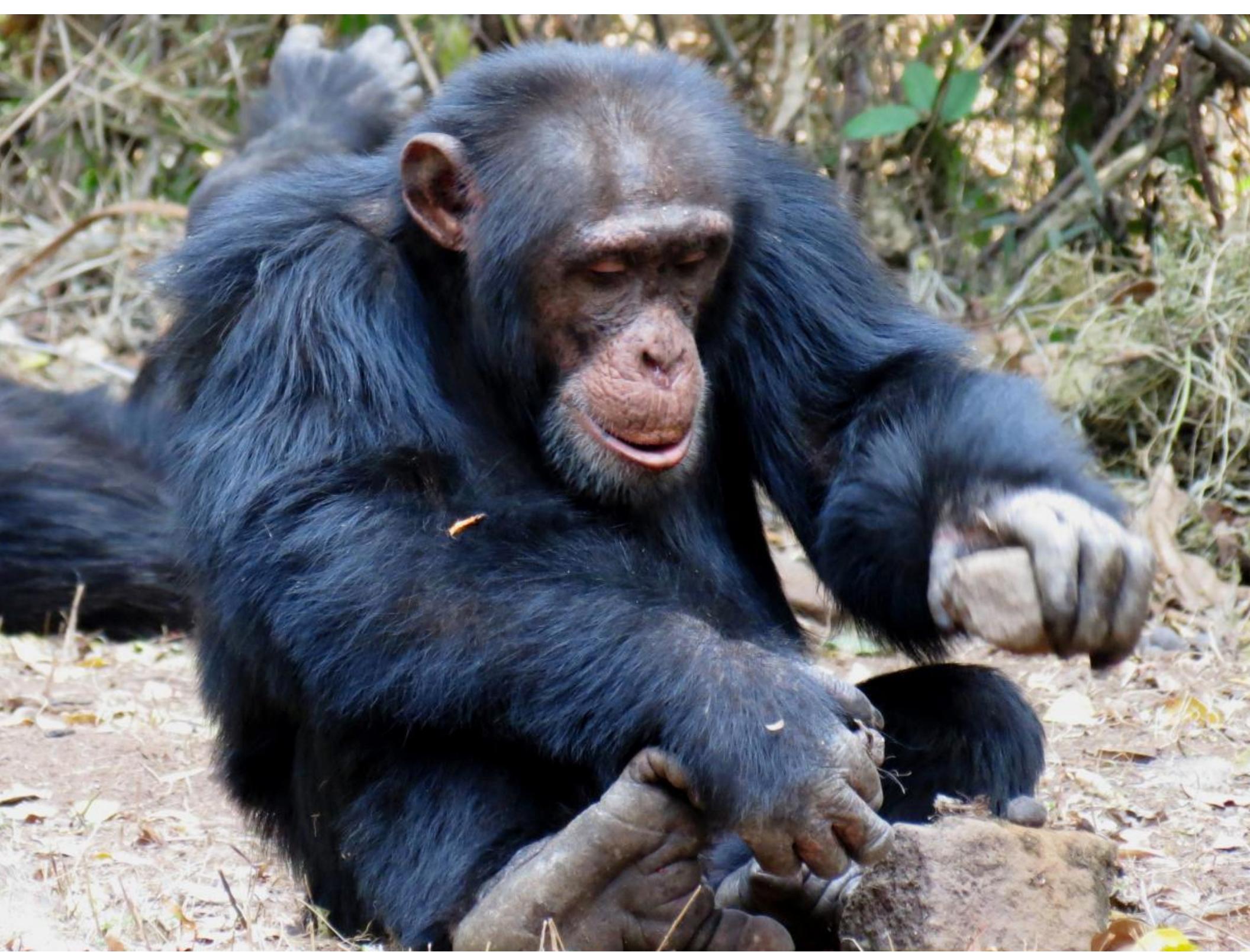




Goal 2

Master Your Tools









The Essential Set of Skills

Debugging, Testing, and Troubleshooting

Engineering Habits

Test, test, test, and test some more; Test as you go

Always start from a working state, take small steps

Visibility is key (printf, logic analyzer, gdb)

Methodical, not random. Search for cause of problem, not solution. Don't throw darts!

Fast prototyping, embrace automation, 1-click build

Don't be frustrated by bugs, relish the challenge, take frequent breaks.

Organized Development Environment



<http://amhistory.si.edu/juliachild/>



To invent, you need a good imagination and a pile of junk
Thomas Edison

Be a Maker and a Doer

Become an Engineer

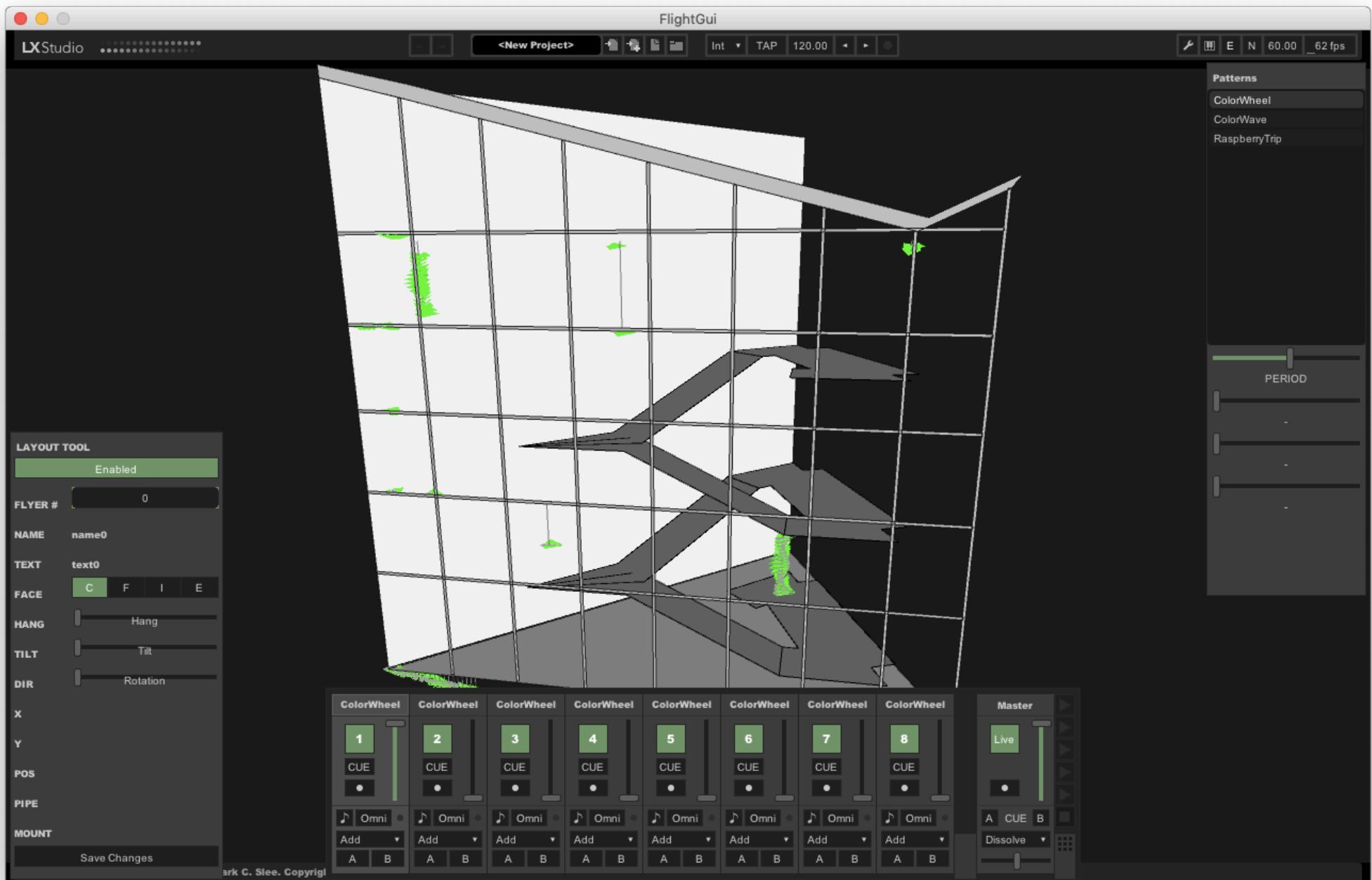
EE125 Interactive Light Sculpture

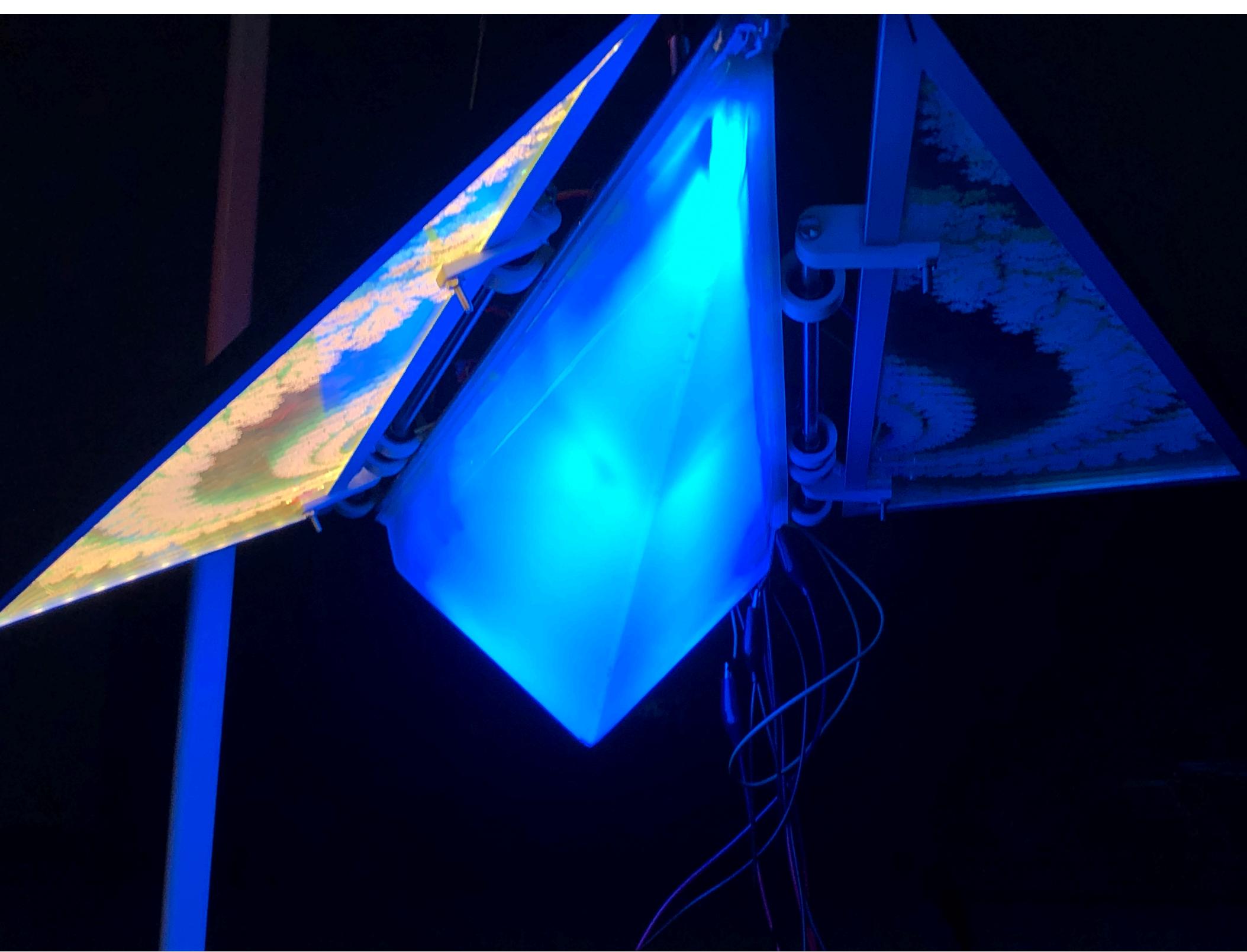
Make Something Beautiful

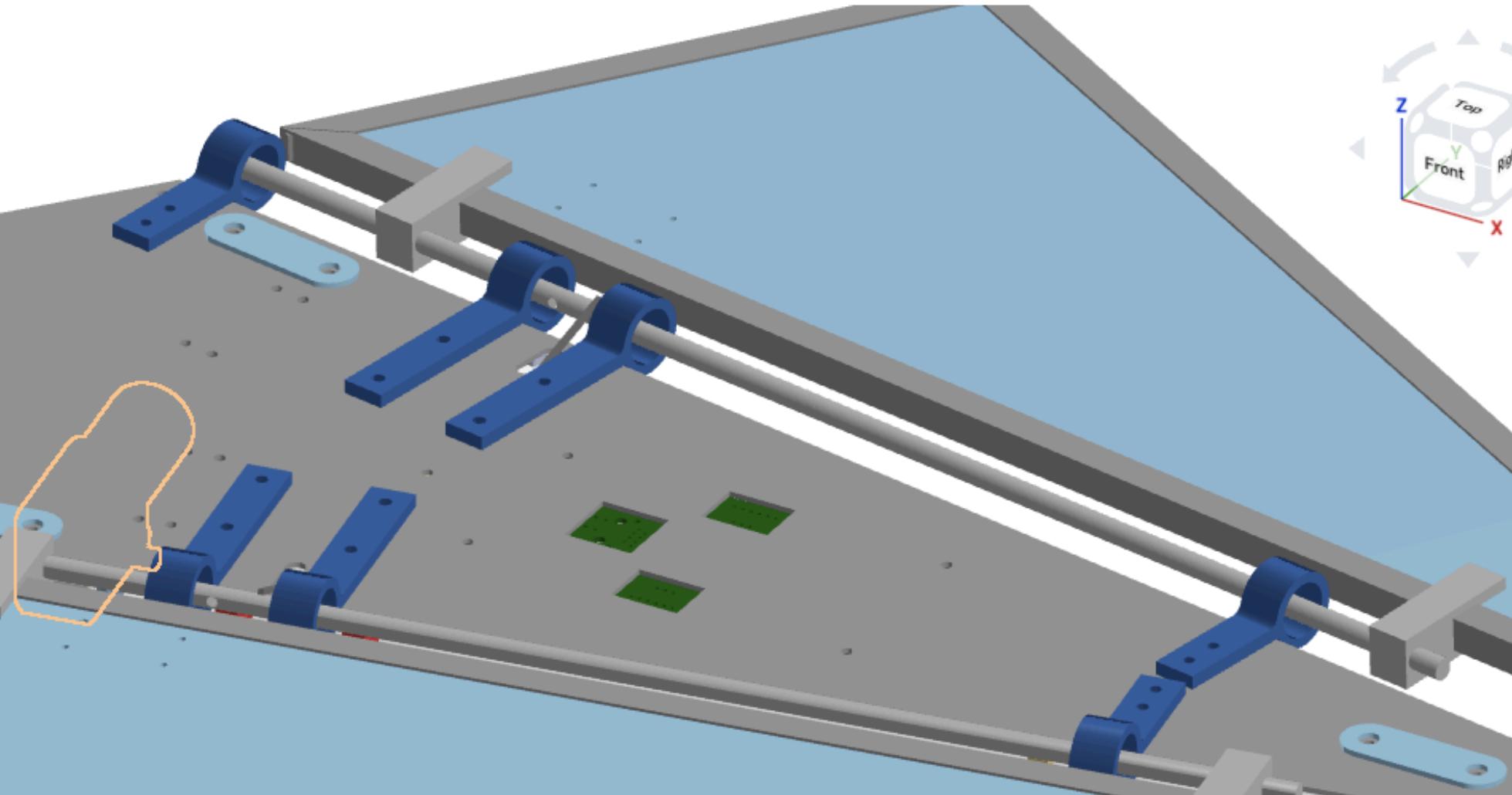


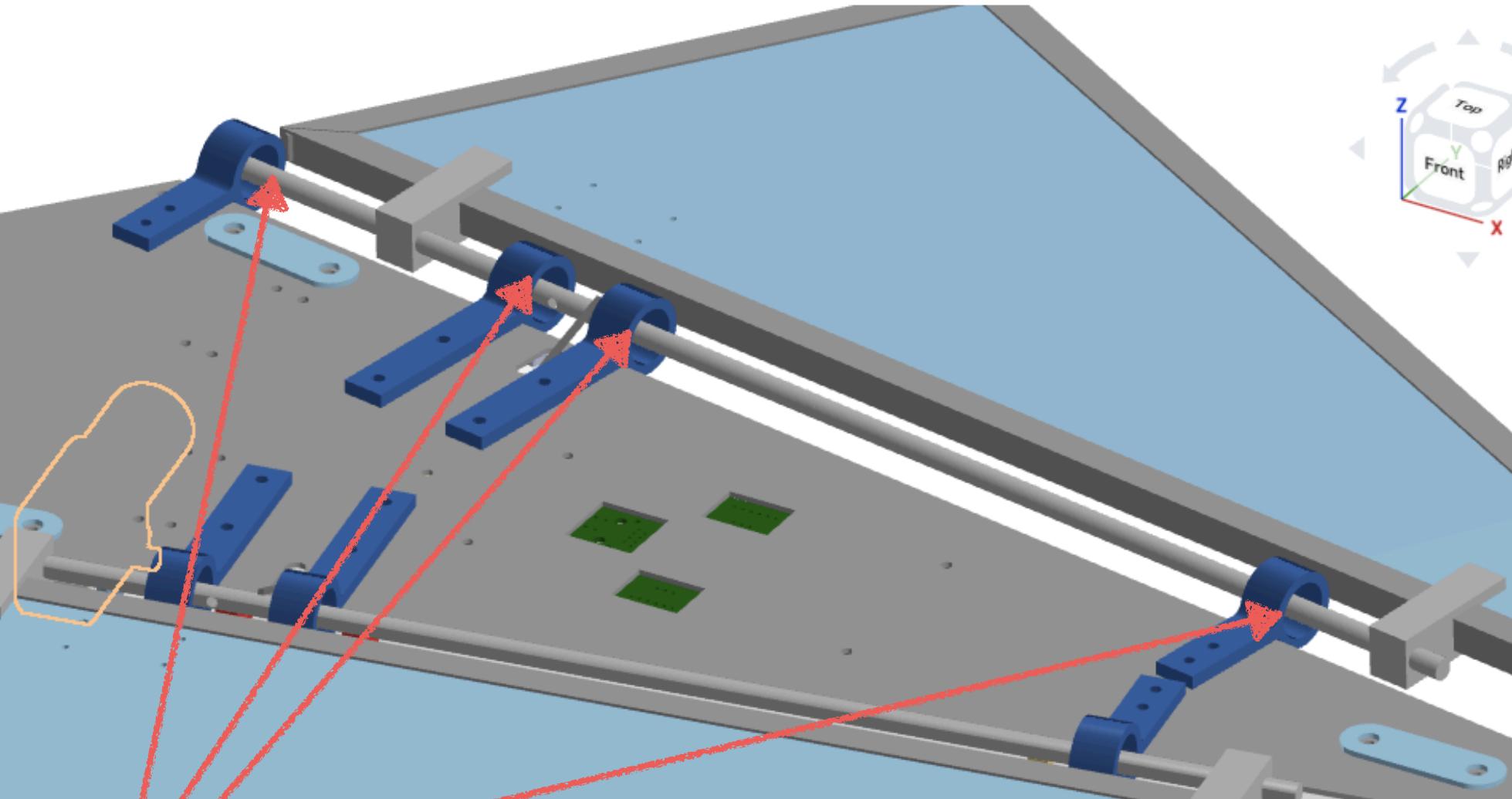
Help Us Celebrate Our 125th Anniversary!

This year the EE Department celebrates its 125th anniversary. As part of the celebration we are building a beautiful interactive light sculpture to be installed in the David Packard Electrical Engineering Building staircase.









Each

In stock
\$5.98 Each
60355K503

ADD TO ORDER

\$50 for just the bearings!

Dry-Running Sleeve Bearings

High-Load Dry-Running Sleeve Bearings



A metal shell adds strength.

Light Duty Dry-Running Sleeve Bearings



Use where high loads and speeds are not required.

High-Temperature Dry-Running Sleeve Bearings



Graphite provides a layer of dry lubrication that can operate in high-temperature environments.

Ultra-Low-Friction Dry-Running Sleeve Bearings



Use these bearings in applications with frequent starts and stops because they operate with very little friction.

Food Industry Dry-Running Sleeve Bearings



Made of FDA listed materials for use in food applications, these bearings also withstand caustic and washdown environments.

Chemical-Resistant Dry-Running Sleeve Bearings



These bearings stand up to a wider range of chemicals than other dry-running sleeve bearings.

Multipurpose Dry-Running Sleeve Bearings



Reinforced with polyester fabric, these plastic bearings have load and speed capabilities comparable to metal bearings.

Water-Resistant Dry-Running Sleeve Bearings



Good for underwater applications, these bearings won't swell or warp when wet.

Harsh Environment

Ball Bearings

Ball Bearings



These steel ball bearings handle higher loads than stainless steel and plastic bearings.

High-Load Ball Bearings



Wider than our standard ball bearings, these rugged bearings are good for handling heavy loads at low speeds.

Permanently Lubricated Ball Bearings



A solid polymer lubricant surrounds the balls, eliminating the need for additional lubrication. Because it's solid, the lubricant blocks out water and dust, extending the life of the bearing.

Light Duty Ball Bearings



Use these bearings where high speed and precision are not required.

Precision Ball Bearings



For greater accuracy and higher speeds, these bearings are made to tighter tolerances than standard ball bearings.

Self-Aligning Ball Bearings



With a spherical raceway and two rows of balls, these bearings compensate for shaft misalignment.

Hex-Bore Ball Bearings



Mount these bearings onto a hex shaft.

Electrically Insulated Ball Bearings



Good for use in electric motors and power generators, these bearings have ceramic balls that insulate against stray current to prevent damage to the bearing.

One-Way Locking Ball Bearing Clutches



Internal locking

Stainless Steel Ball Bearings



Bearings are 440C stainless steel for good corrosion resistance.

Permanently Lubricated Stainless Steel Ball Bearings

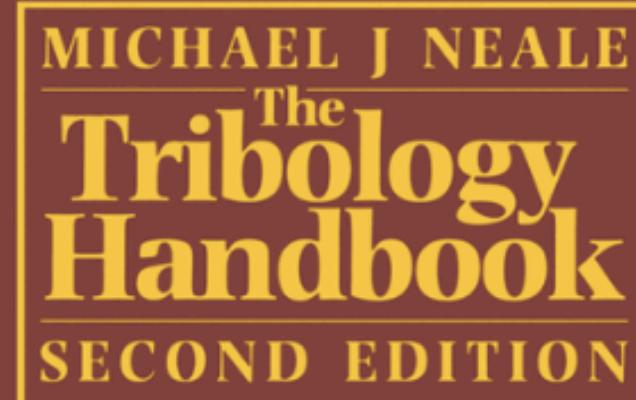


A solid polymer

Ultra-Corrosion-Resistant Stainless Steel Ball Bearings



These 316 stainless



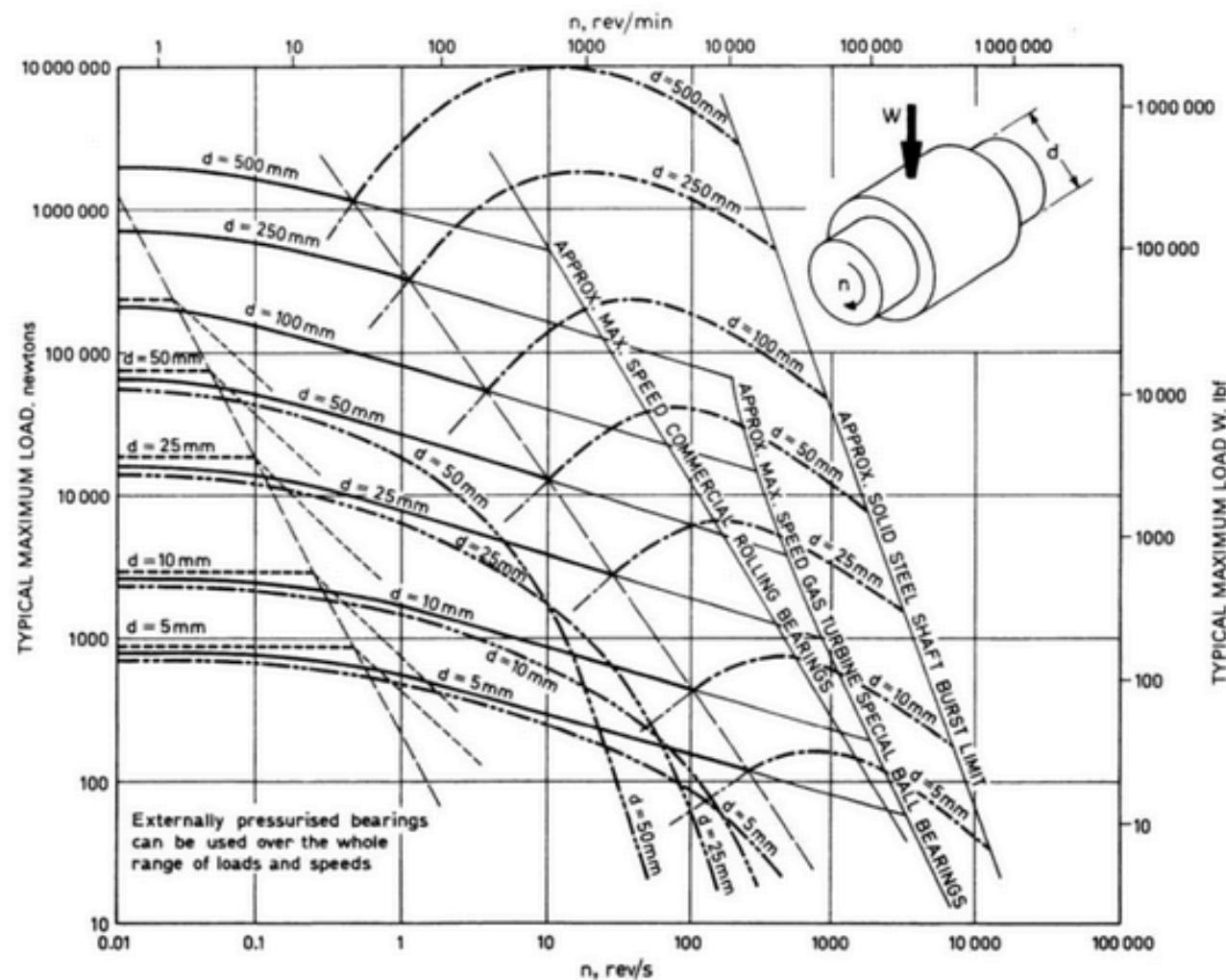
Tribology

From Wikipedia, the free encyclopedia



It has been suggested that [Draft:Green tribology](#) be merged into this article. ([Discuss](#)) Proposed since May 2020.

Tribology is the science and engineering of interacting surfaces in relative [motion](#). It includes the study and application of the principles of [friction](#), [lubrication](#), and [wear](#). Tribology is highly interdisciplinary. It draws on many academic fields, including [physics](#), [chemistry](#), [materials science](#), [mathematics](#), [biology](#), and [engineering](#). People who work in the field of tribology are referred to as *tribologists*.^[1]



Rubbing plain bearings in which the surfaces rub together.
The bearing is usually non-metallic.



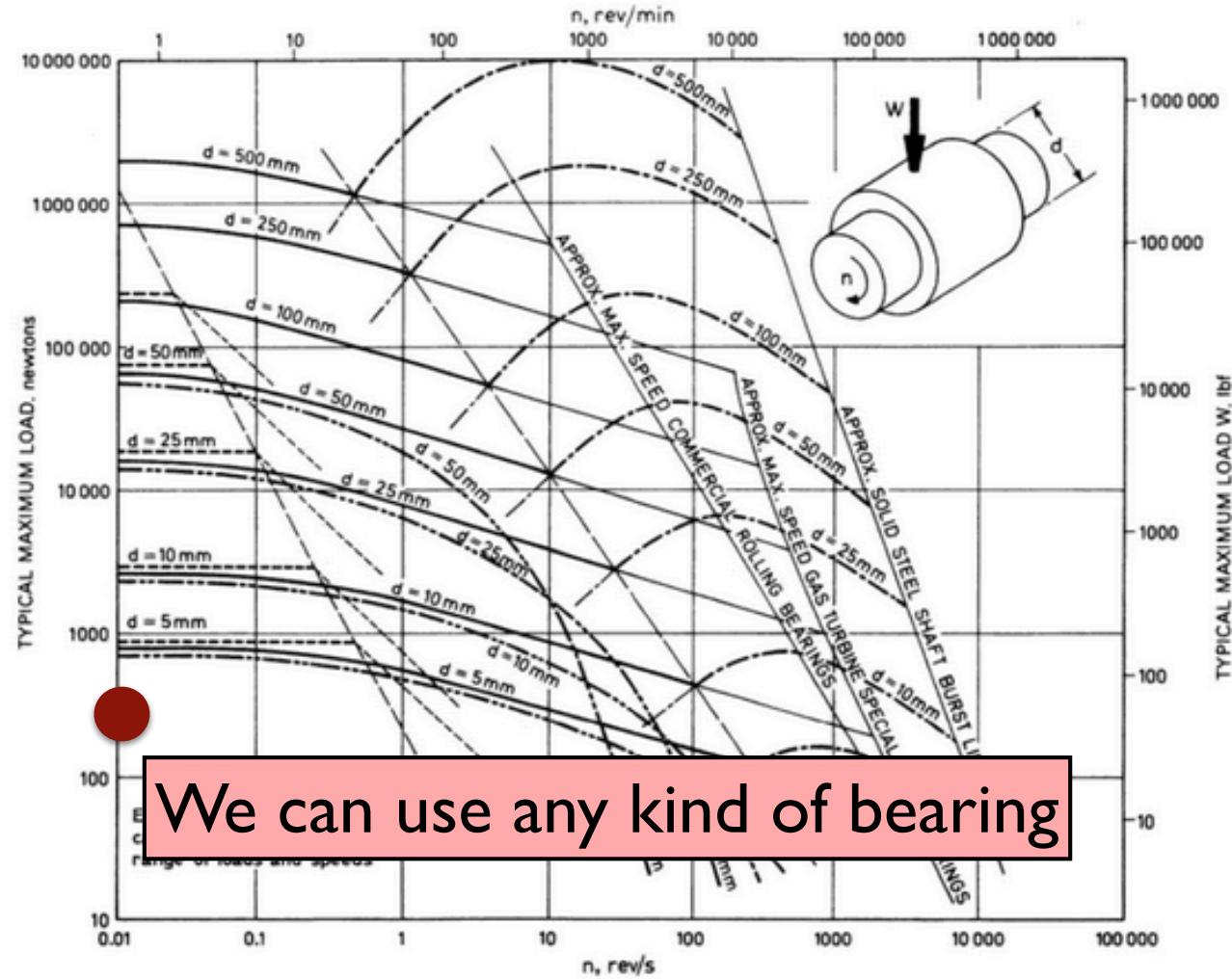
Plain bearings of porous metal impregnated with a lubricant.



Rolling bearings. The materials are hard, and rolling elements separate the two moving components.



Fluid film plain bearings. A hydrodynamic pressure is generated by the relative movement dragging a viscous fluid into a taper film.



Rubbing plain bearings in which the surfaces rub together.
The bearing is usually non-metallic.



Plain bearings of porous metal impregnated with a lubricant.



Rolling bearings. The materials are hard, and rolling elements separate the two moving components.



Fluid film plain bearings. A hydrodynamic pressure is generated by the relative movement dragging a viscous fluid into a taper film.

<https://www.hackster.io/>

<https://www.electronicshub.org/iot-project-ideas/>

<https://electronicsforu.com/iot-projects-ideas>

**[https://makezine.com/category/technology/
wearables/](https://makezine.com/category/technology/wearables/)**

<http://blondihacks.com/>

Linux and Beyond

Operating Systems

Multiple processes (scheduling)

Memory management (virtual memory)

Share peripherals (devices)

File systems and storage

Networking

Protect/secure processes from each other

Modes

		Privileged modes				
		Exception modes				
User	System	Supervisor	Abort	Undefined	Interrupt	Fast interrupt
R0	R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7	R7
R8	R8	R8	R8	R8	R8	R8_fiq
R9	R9	R9	R9	R9	R9	R9_fiq
R10	R10	R10	R10	R10	R10	R10_fiq
R11	R11	R11	R11	R11	R11	R11_fiq
R12	R12	R12	R12	R12	R12	R12_fiq
R13	R13	R13_svc	R13_abt	R13_und	R13_irq	R13_fiq
R14	R14	R14_svc	R14_abt	R14_und	R14_irq	R14_fiq
PC	PC	PC	PC	PC	PC	PC

CPSR	CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
		SPSR_svc	SPSR_abt	SPSR_und	SPSR_irq	SPSR_fiq

△ indicates that the normal register used by User or System mode has been replaced by an alternative register specific to the exception mode

Modes

		Privileged modes				
		Exception modes				
User	System	Supervisor	Abort	Undefined	Interrupt	Fast interrupt
R0	R0	R0	R0	R0	R0	R0
R1	R1	R1			R1	R1
R2	R2	R2	nc	nc	R2	R2
R3	R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7	R7
R8	R8	R8	R8	R8	R8	R8_fiq
R9	R9	R9	R9	R9	R9	R9_fiq
R10	R10	R10	R10	R10	R10	R10_fiq
R11	R11	R11	R11	R11	R11	R11_fiq
R12	R12	R12	R12	R12	R12	R12_fiq
R13	R13	R13_svc	R13_abt	R13_und	R13_irq	R13_fiq
R14	R14	R14_svc	R14_abt	R14_und	R14_irq	R14_fiq
PC	PC	PC	PC	PC	PC	PC
CPSR	CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
		SPSR_svc	SPSR_abt	SPSR_und	SPSR_irq	SPSR_fiq

△ indicates that the normal register used by User or System mode has been replaced by an alternative register specific to the exception mode

Modes

		Privileged modes				
		Exception modes				
User	System	Supervisor	Abort	Undefined	Interrupt	Fast interrupt
R0	R0	R0	R0	R0	R0	R0
R1	R1	R1			R1	R1
R2	R2	R2	nc	nc	R2	R2
R3	R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7	R7
R8	R8	R8	R8	R8	R8	R8_fiq
R9	R9	R9	R9	R9	R9	
R10	R10	R10	R10	R10	R10	
R11	R11	R11	R11	R11	R11	R11_fiq
R12	R12	R12	R12	R12	R12	R12_fiq
R13	R13	R13_svc	R13_abt	R13_und	R13_irq	R13_fiq
R14	R14	R14_svc	R14_abt	R14_und	R14_irq	R14_fiq
PC	PC	PC	PC	PC	PC	PC
CPSR	CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
		SPSR_svc	SPSR_abt	SPSR_und	SPSR_irq	SPSR_fiq



indicates that the normal register used by User or System mode has been replaced by an alternative register specific to the exception mode

Modes

		Privileged modes				
		Exception modes				
User	System	Supervisor	Abort	Undefined	Interrupt	Fast interrupt
R0	R0	R0	R0	R0	R0	R0
R1	R1	R1			R1	R1
R2	R2	R2	nz	nz	R2	R2
R3	R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7	R7
R8	R8	R8	R8	R8	R8	R8_fiq
R9	R9	R9	R9	R9	R9	
R10	R10	R10	R10	R10	R10	
R11	R11	R11	R11	R11	R11	R11_fiq
R12	R13	R12	R12	R12	R12	R12_fiq
R13	R13	R13_svc	R13_abt	R13_und	R13_irq	R13_fiq
R14	R14	R14_svc	R14_abt	R14_und	R14_irq	R14_fiq
PC	PC	PC	PC	PC	PC	PC
CPSR	CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
		SPSR_svc	SPSR_abt	SPSR_und	SPSR_irq	SPSR_fiq



indicates that the normal register used by User or System mode has been replaced by an alternative register specific to the exception mode

User vs Supervisor Mode

User mode can't do a lot of things ...

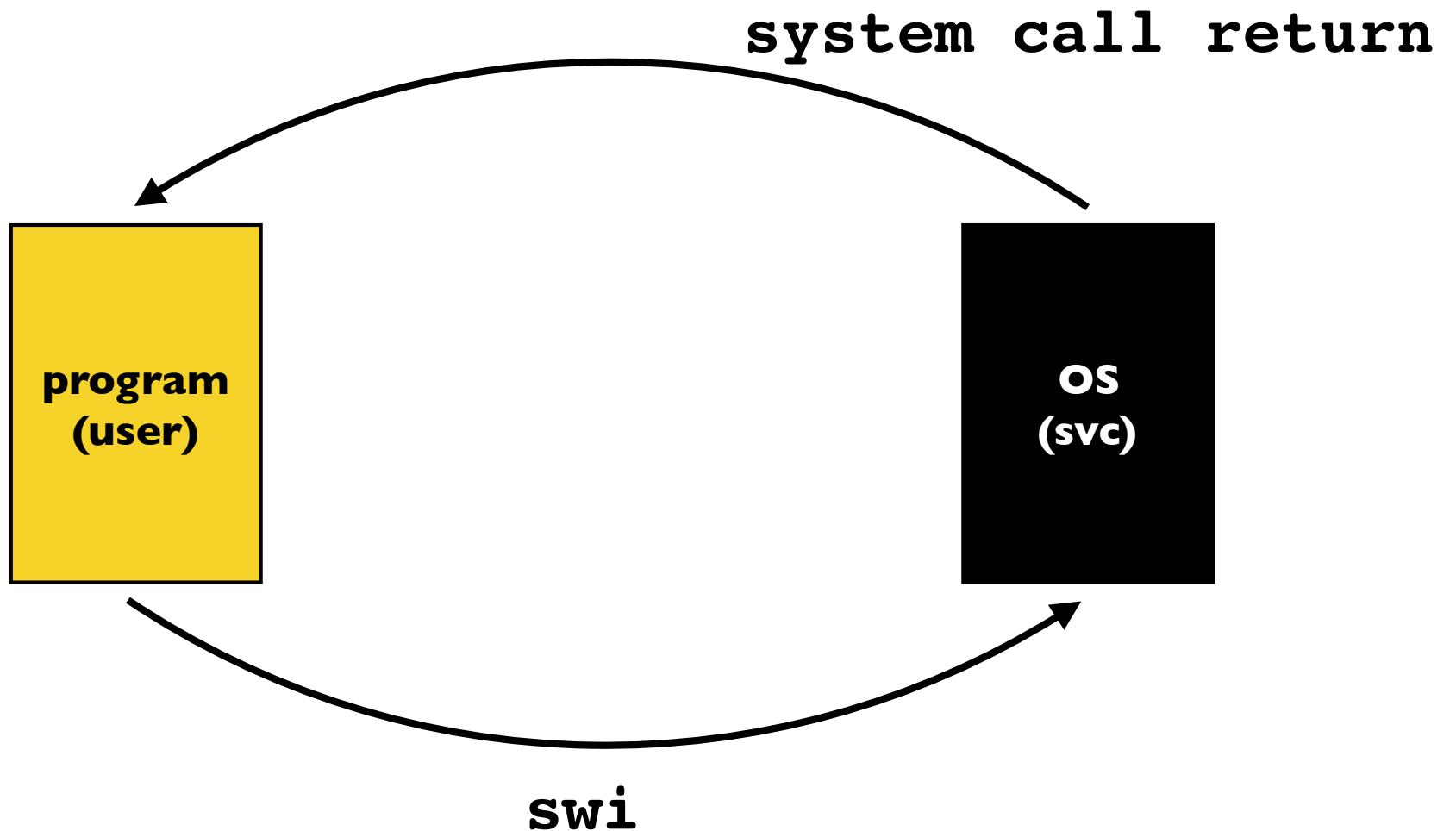
- 1. Modify CPSR (change modes)**
- 2. Access configuration registers**
- 3. Access IO registers/memory**

Hardware throws an exception if user mode program does any of these things

Enables operating system to keep processes from interfering with each other

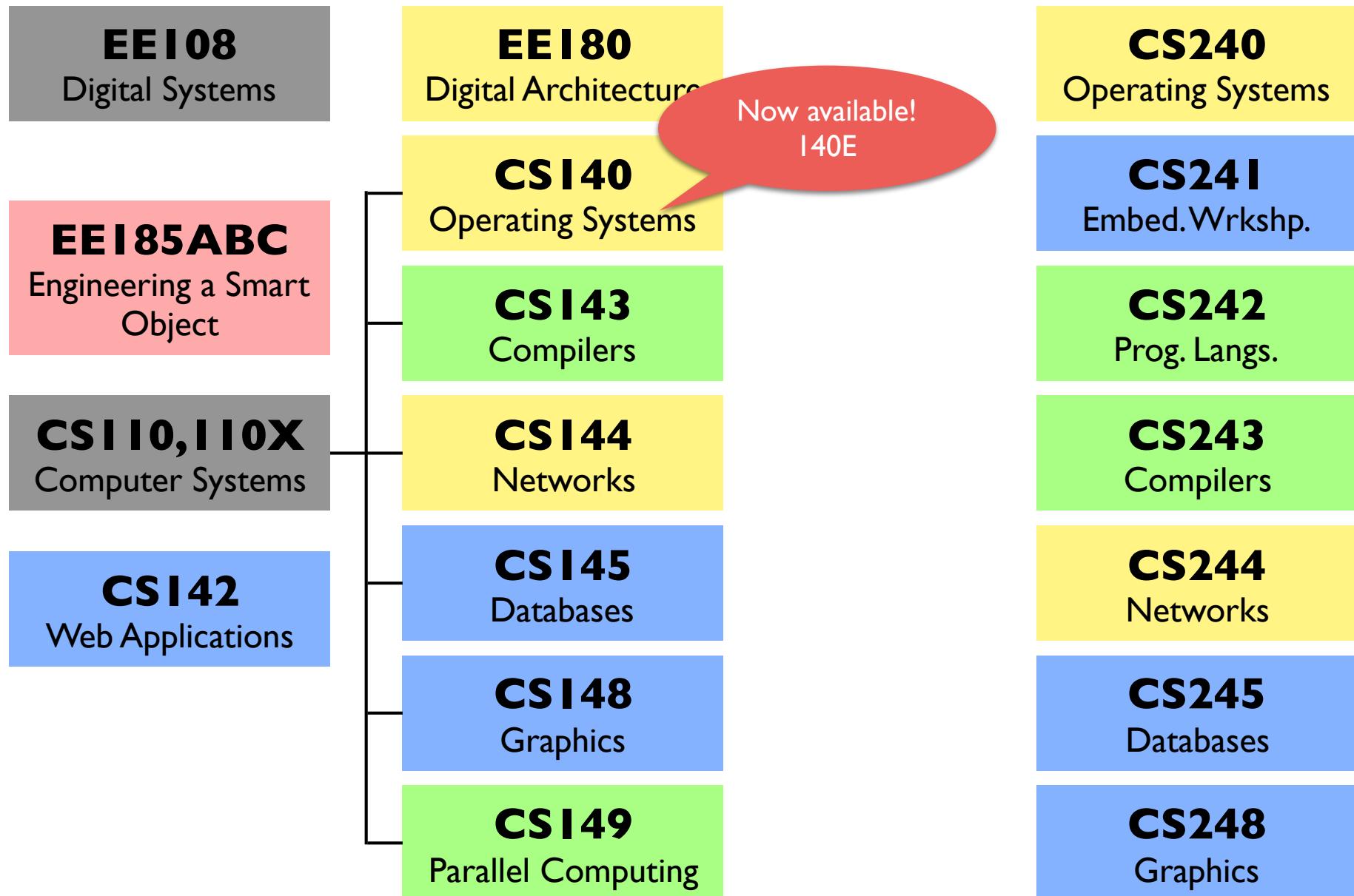
Essential for security

Operating System



Linux Boot

Next Steps



Demos on Friday!

- Milestones!**
- To submit receipts**
- <http://tinyurl.com/edaffairs>**
- Please do it this week! You may have to pay taxes if you wait**

Demonstrations

- **Friday Nov 20 from 2:30-3:50 pm in class**
- **We'll go around one by one to demo**
- **Email us beforehand if you're not comfortable demoing, but we encourage you to do so: everyone understands that there are a lot of important real-world things going on right now**

Submission

- **Due Friday night**
- **Final github commit of code**
- **Include README.md describing your project (pictures!!, attributions)**

Logistics

- **Submit reimbursements (see previous slide)**

CS107e — what have you accomplished?

<https://www.youtube.com/watch?v=th5A6ZQ28pE>