WEEK 3 – SYNBIO STANDARDS

BioBrick

BioBrick standard biological parts are DNA sequences of defined structure and function; they share a common interface and are designed to be composed and incorporated into living cells such as E. coli to construct new biological systems. BioBrick parts represent an effort to introduce the engineering principles of abstraction and standardization into synthetic biology. The trademarked words BioBrick and BioBricks are correctly used as adjectives (not nouns) and refer to a specific "brand" of open source genetic parts as defined via an open technical standards setting process that is led by the BioBricks Foundation.

One of the goals of the BioBricks project is to provide a workable approach to nanotechnology employing biological organisms. Another, more long-term goal is to produce a synthetic living organism from standard parts that are completely understood.

Each BioBrick part is a DNA sequence held in a circular plasmid; the "payload" of the BioBrick part is flanked by universal and precisely defined upstream and downstream sequences which are technically not considered part of the BioBrick part. These sequences contain six restriction sites for specific restriction enzymes (at least two of which are isocaudomers), which allows for the simple creation of larger BioBrick parts by chaining together smaller ones in any desired order. In the process of chaining parts together, the restriction sites between the two parts are removed, allowing the use of those restriction enzymes without breaking the new, larger BioBrick apart. To facilitate this assembly process, the BioBrick part itself may not contain any of these restriction sites.

There are three levels of BioBrick parts: "parts", "devices" and "systems". "Parts" are the building blocks and encode basic biological functions (such as encoding a certain protein, or providing a promoter to let RNA polymerase bind and initiate transcription of downstream sequences); "devices" are collections of parts that implement some human-defined function (such as a riboregulator producing a fluorescent protein whenever the environment contains a certain chemical); "systems" perform high-level tasks (such as oscillating between two colors at a predefined frequency).

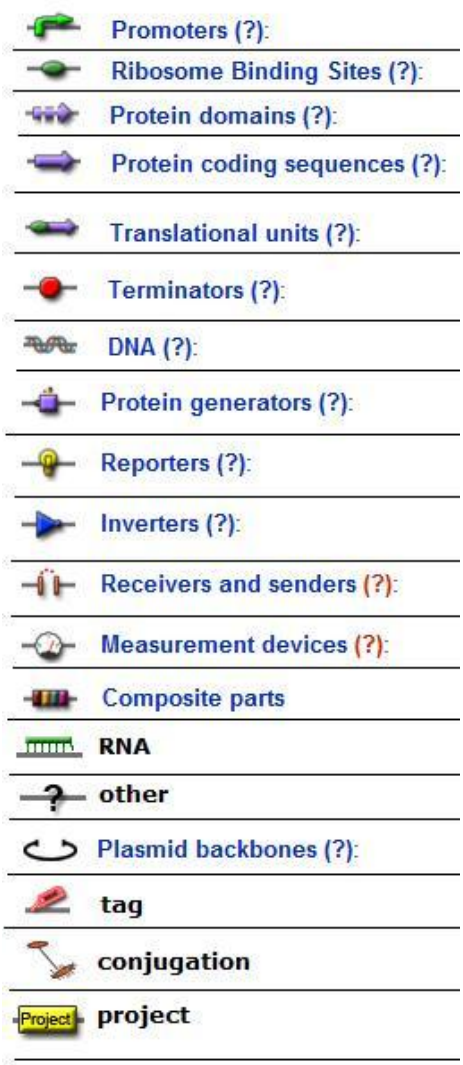
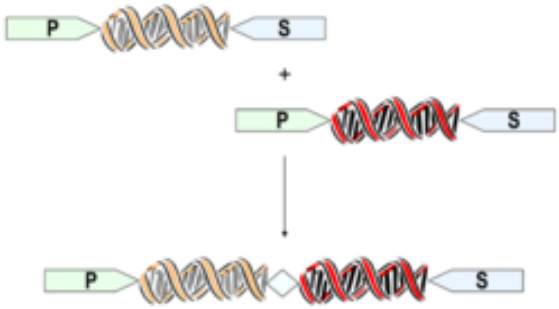


Figure 1. Parts

All BioBrick formats proposed so far follow the same basic scheme where restriction and ligation of two BioBricks forms a new BioBrick:



Abstraction Hierarchy

Abstraction hierarchies are a **human** invention designed to assist people in engineering very complex systems by ignoring unnecessary details. If the process to design a biological system was to write down the string of nucleotides, it would immediately become untenable even for experts to design anything but very simple systems. Most people just aren't capable of processing that kind of detail all at once. If instead, an abstraction hierarchy is specified, it allows the designer of a biological system to ignore some of the implementation details and focus only on the high-level design issues.

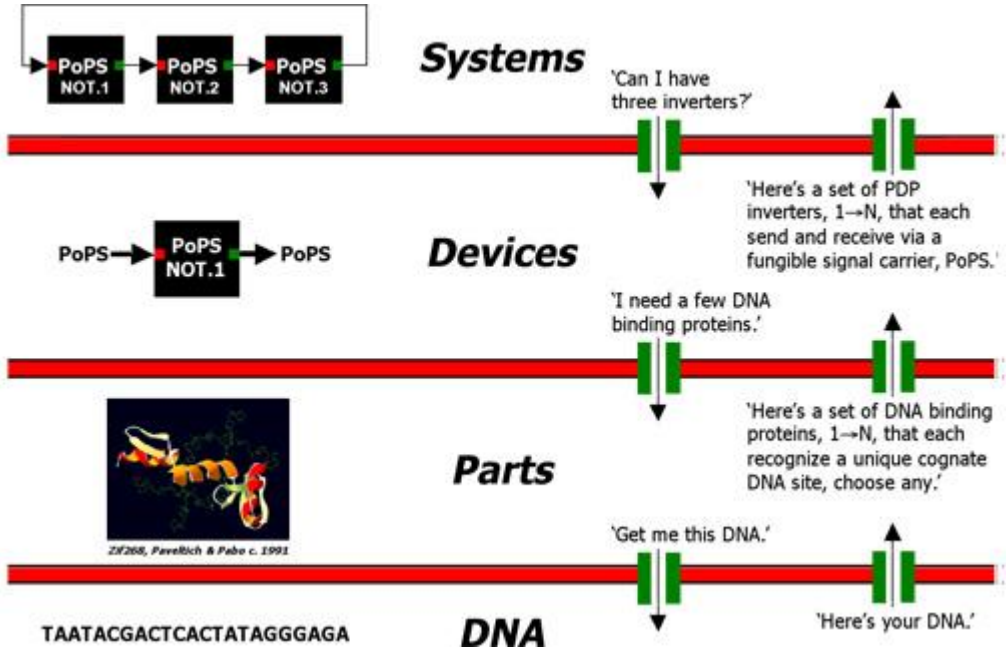
Engineers in all disciplines take advantage of abstraction hierarchies to design and build complicated systems. For instance, software engineers write in high level programming languages like C++ or Java which are designed to be easy for humans to read and write. These programs are then translated into lower level sets of instructions that are more easily translatable to bit strings that are machine interpretable and implementable. Thus, the people who write C++ programs do not need to know how to translate their programs to machine code and the people who work on instruction sets do not need to envision all possible programs that the software engineer might write.

To enable the engineering of very complex biological systems, it will be necessary to develop abstraction hierarchies for biological engineering. At this point, it is not necessarily clear which hierarchies are most useful and in fact it may be slightly premature to try and develop them. Nevertheless, thinking about what an abstraction hierarchy in synthetic biology should look like might help us think about the "right" way to engineer biological systems and to design biological parts.

Below several abstraction hierarchies are listed that might be appropriate for biological engineering. Anyone should feel free to revise them, add new candidate hierarchies or add comments as this is very

much a work in progress. An attempt has been made to give credit to the originators of each of the candidate abstraction hierarchies; however, this should in no way be a deterrent to those interested in offering revisions. The abstraction hierarchies have been listed in chronological order of inception.

DNA, parts, devices and systems model



Layer name	Definition	Example
DNA	sequence of nucleotides	ATGGATCATGATG
Part	a finite sequence of nucleotides with a specific function	RBS, CDS, promoter, terminator
Device	multiple parts with a higher level function	inverter
System	multiple devices hooked together	ring oscillator

The original abstraction hierarchy is posted on the Registry page and is originally from one of Drew's slides.

by Drew Endy.

Screenability model

Layer name	Definition	Example
DNA	sequence of nucleotides	ATGGATCATGATG
Part	a finite sequence of nucleotides with a specific function	RBS, CDS, promoter, terminator
Device	one or more parts which can be screened for functionality	promoter, terminator, inverter
System	multiple devices which cannot be screened for functionality	ring oscillator

by Jason Kelly.

Composition model

Layer name	Definition	Example
DNA	sequence of nucleotides	ATGGATCATGATG
Part	a sequence of DNA with a specific function that can be physically combined with other parts via an assembly standard	RBS, CDS, promoter, terminator
Device	a set of parts that can be functionally combined with other devices via a common, standard signal carrier (i.e. PoPS, RiPS, PhPS)	Inverter
System	a set of devices that cannot be functionally combined with other devices via a common, standard signal	ring oscillator

See Synthetic Biology:Abstraction hierarchy/Composition model for notes on the abstraction hierarchy developed based on composability.

by ReshmaShetty and Barry Canton.

Network layer model

This model derives inspiration from the Wikipedia:OSI model for computer network protocols.

Version 1

Layer Number	Layer Name	Example Standard	Role of User	Category
Layer 7	Application	chemical detector	Brainstorm need	System
Layer 6	Packaging	pSB plasmids	Physical handling of system	System
Layer 5	Environment	wavelengths of light	Provide input or observe output	System
Layer 4	Cell	cell-cell signaling	none	Cell
Layer 3	Protein	dimerization interface	none	Part
Layer 2	RNA	PoPS	none	Part
Layer 1	DNA	BioBricks assembly	none	Part
Layer 0	Chassis	nucleotides/amino acids	none	Chassis

by Austin Che.

Version 2

This version attempts to reconcile the network layer model with the composition model.

Layer Number	Layer Name	Example Standard	Role of User
6	User	Detector of Chemical X	
5	Environment	Batch/continuous, Temp., Media	Provide input or observe output
4	Population	cell-cell signaling	Design interactions between different cells
3	System	Signaling molecules, fluorescence	Design system to process external inputs into detectable outputs
2	Device	PoPS, RiPS	Use parts to design device with particular transfer curve
1	Part	BioBricks assembly	Plan and assemble
0	Materials	nucleotides/amino acids	Choose the materials

by Barry Canton.

See Synthetic Biology:Abstraction hierarchy/Network layer model for more detailed and extensive notes on the network layer model.