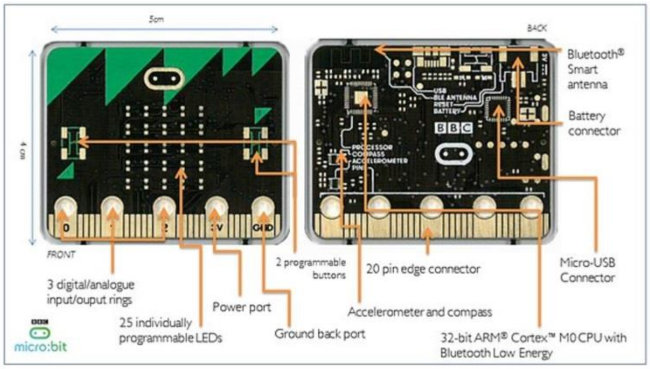
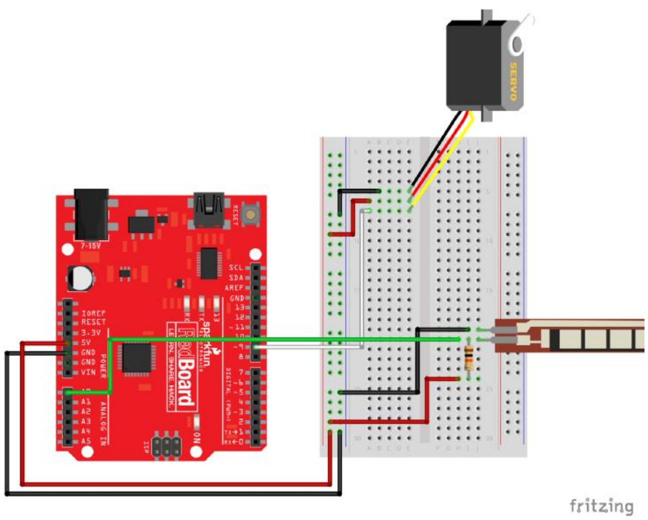


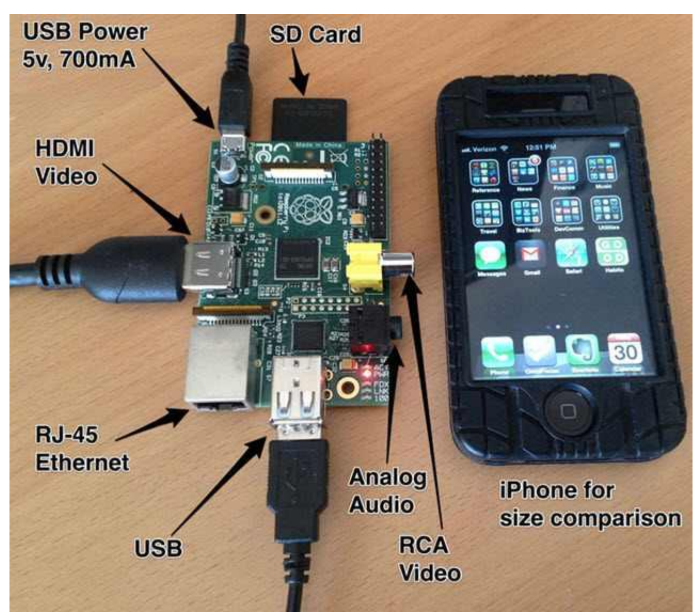
Notes on--

Engineering a STEM Curriculum



by
Steve Merten,
West Lutheran High School
2017 MISF STEM Conference
smerten@wlhs.net

Link to presentation:
goo.gl/ZShLWB



Note: the purpose of this presentation is not to give you a curriculum to adopt but an approach for your faculty to create a unique STEM curriculum

PROBLEMS WITH TRADITIONAL APPROACHES TO CURRICULUM

- The fourth "R" … Reading, wRiting , 'Rithmatic & tinkeRing.
- It's difficult to add some elements of "E" using traditional text/curriculum publishers
    - Tinkering, trouble shooting, hacking,
    - These are skills that require modeling
    - If you teach step by step tinkering you are missing essence of tinkering which trying first one solution, that another until one of these pops out as an actual solution
    - Tinkering is not a following a cookbook recipe of projects
    - Beware the craft-project masquerading as a STEM lesson [here's an example of wonderful lesson on prototyping gone wrong…step-by-step prosthetics lesson LINK]
- The nature of the "T" in STEM has some qualities that can elude publishers
    - Standardized systems-- lots of reasons not to show students just one system that works since adaptability/versatility is a major commodity
    - Moving target—Because the nature of engineering is to constantly be on the lookout for tech improvements, you have to embrace constant change (which is hard for a conservative, Midwestern, middle aged Lutheran)
- Curriculum development, the Curriculum Map & Block Plans
    - If you have the teacher personality type of needing day-by-day lesson plans for an entire year for your STEM course, this method might not be the easiest
    - No page #'s to write in your lesson plans
    - A great fit for standards based curriculum & assessment if your standards are skill driven *and* Project Based Learning (PBL)
- Packaged STEM teaching systems *(On the other hand…)*
    - A lot of teachers are very happy with *Project Lead the Way* and *VEX* curriculum and components (the author is still getting up to speed with these). More on these later

METRICS FOR A CURRICULUM

A good STEM curriculum approach should…

- ✓ Support inquiry
- ✓ Not be locked into one technology type or one technology system
- ✓ Is "hackable"
    - If you get an idea walking to your classroom, does your approach allow you add on the fly?
    - Lends itself to last minute changes or on the spot "What if we…" approaches
- ✓ Students can seek out more than one way to reach a goal
- ✓ Skills are taught with the idea that they may be carried out different, equally valued methods

The goal of your curriculum is not making engineers, not making repairmen, fabricators or programmers, but rather problem solvers not afraid to use creative applications of old or new technology.

- E.g.: you should be careful locking into a specific coding language with a junior high classroom

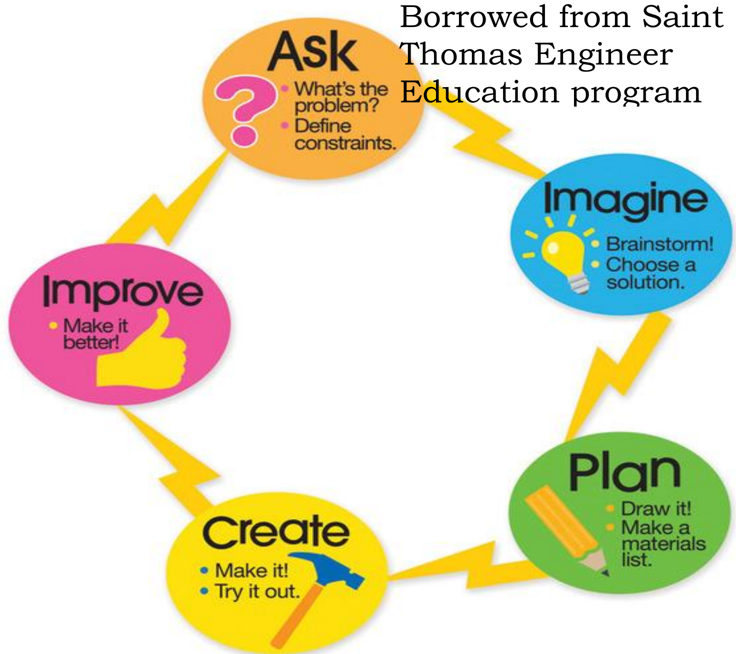
✓ Look out for the challenge/premises of a unit that is not authentic (not every lesson is looking for a solution to world hunger)

✓ Does the “A” of STEAM belong?

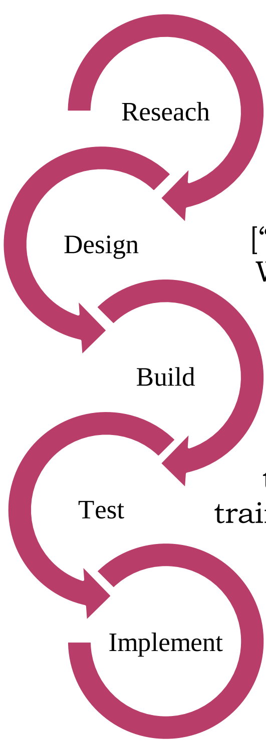
- If *art* = elegant or creative … yes
- Overlapping cognitive areas such as improvisation or Gestalt undertones
- Technology behind the instruments of music or tools / media of art
- However… what area of learning doesn’t overlap with STEM and deserve its own initial (writing, pub speaking and presentations, languages, culture studies, tech reading, etc.)

ENGINEER YOU CURRICULUM

If you can get permission from your administration, boards and colleagues, try to stretch your curriculum creation into a two year process. Use the same process that you’d apply to Engineering design for curriculum buildings



Borrowed from Saint Thomas Engineer Education program



OUR STORY

MISF GRANT PROPOSAL:

[“Systemic STEM- West Lutheran High School has the bits and pieces of a STEM curriculum but not a clear vision of what our final student product should look like in terms of STEM when walking up to the stage to get a diploma. We need a study of what pieces are already in place and what needs adding or overhauling in our Math and Science curriculums. To do this, we need a two year plan on how to get the needed teacher training and classroom supplies and place and a schedule for course modifications. The end result will be a system-infused approach that will bring more the “T” & “E” of STEM in our courses. In other words, a Systemic STEM Curriculum”]

Our goal was to develop a Maker-centric classroom built around project-based learning (PBL) that included elements of electronics, code writing, engineering design and with exposure to CAD
Below is a rough set of stages we would use to make our STEM more *systemic.*

1. **Teacher training** for some or all 5 of our Math/Science instructors. This would be aimed at making our teachers more adept in engineering technology
2. **Basic starter materials** such as an electronics starter assortment, simple robotics equipment, CAD ware, Arduino based student kits-- not full classroom sets but a sampling of possible classroom equipment additions for teacher learning.
3. **Staff support for developing a curriculum** that serves all of our students with a general STEM background and a specialized track for those with engineering gifts and interests
4. Staff support for writing a **two year implementation plan**.
5. **Purchasing** and obtaining the needed **classroom materials** and supplies.
6. Eventual **inclusion of association grade schools** by offering code building events/tech camps such as the code writing activities in offer through the "Hour of Code" organization

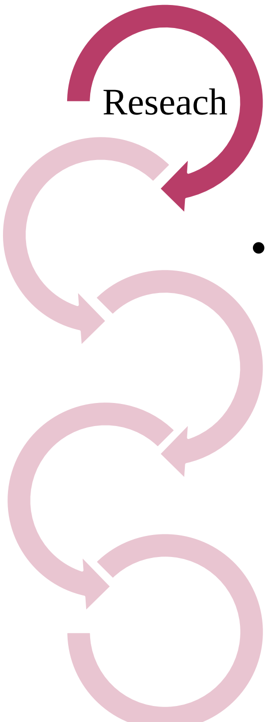
At this point I would see one or two of the staff starting in on steps 1 and 2 this spring and summer so that all of the staff would be able to begin step 3 over the course of a year. During that time lessons and units could be "taken out for a test-drive" with various classes and levels. Step 4 could be tackled during the summer of '17 with implementation for some of the classes during the fall of '17 and be full implementation by end of the '18-19 school year. This is subject to what our staff's final plan that would be developed during step #3.

We hope this request isn't dismissed because it isn't the typical request for funding to do something like building a planetarium or a Maker Space that may supplement a small part of a curriculum at best. We are really asking for help to build a robust curriculum. We want a school with clever, innovative teachers that are itching to spark the techno-creative side of our kids and develop an *engineer-like* thinking skill. That takes fresh training for staff and the ability for the school to surround teachers with the same type of "toys" and tools that might work well with students.

We also requested and received a grant from the American Chemical Society that supplied a number of Arduino Kits as well as training with Sparkfun in Boulder Colo.

STAGE TWO: FIND THE TOYS (AFTER PERMISSION & A GRANT)

(Bear in mind that this part of the plan was carried out by someone suffering from *Excessive Gadget Acquisition Disorder – EGAD)*



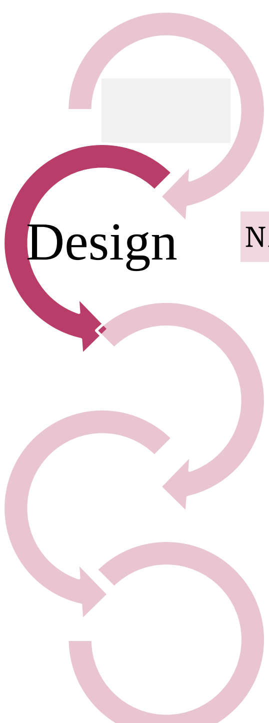
Survey of systems, devices & philosophies:

Here's some of what we sampled and tried out before introducing them to the students:

- Parallax Robot Shield w Arduino Kit
- Sparkfun digital Sandbox kit
- Sparkfun's Tinker Kit
- Sparkfun's Inventor's Kit
- Flash Forge Finder 3D printer
- Sparkfun's Redbot

And read some of the following literature in addition to myriad Maker space forums and STEM Ed websites

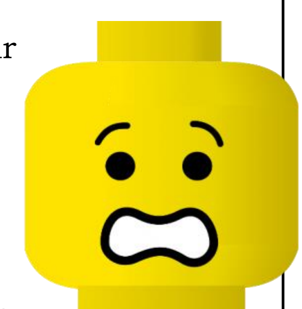
- Engineering the Future
- Principles of Engineering (Project Lead the Way)
- The Invent To Learn Guide to 3D Printing in the Classroom: Recipes for Success
- 3D Modeling and Printing with Tinkercad: Create and Print Your Own 3D Models
- Make: 3D Printing: The Essential Guide to 3D Printers
- JavaScript Robotics: Building NodeBots with Johnny-Five, Raspberry Pi, Arduino, and BeagleBone (Make)
- Sylvia's Super-Awesome Project Book: Super-Simple Arduino (Volume 2)
- Make: Volume 48 Desktop Fabrication: Fab Factory
- Make: Arduino Bots and Gadgets
- Invent To Learn: Making, Tinkering, and Engineering in the Class
- The Arduino Inventor's Guide



NARROWING THE CHOICES (MAYBE NOT LEGOS)

After a summer of reading and sampling, here are **the components we wanted in the Tech/Engineering portion of our 9th grade STEM semester**

- ➢ The elements of Structural Engineering and their overlap with Mechanical
- ➢ Electronics principles and circuitry
- ➢ Code writing
- ➢ Engineering Design
- ➢ Introduction to Computer Aided Drafting
- ➢ The Engineering process (what turns a procedure



into “engineering?”)

- Trouble Shooting /Tinkering practice
- Include a Project Based Learning (PBL) approach as often as practical.

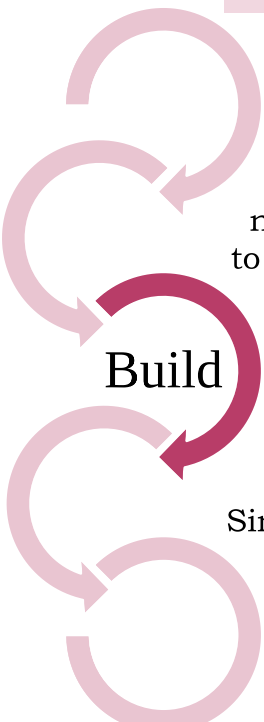
Things that were not included, but perhaps should have been:

- Bio/medical Engineering
- A weekly, collaborative engineering challenge
- A better intersection/coordination of physics and mechanical engineering

Methods, “prewritten” curriculum we wanted to avoid

- Systems that were too “toy-like” , that did not feel like they close to real-life applications
- Fool-proof kits that took away a creative design process

ELEMENTS OF OUR DESIGN



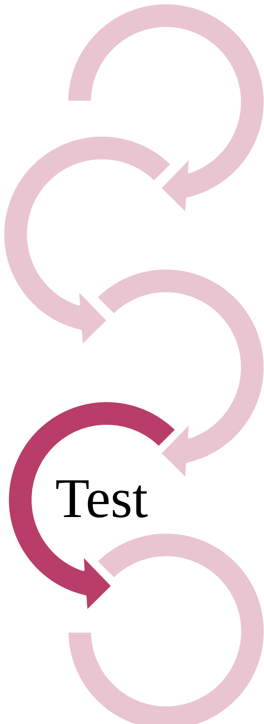
It was time to make order out of what was starting look like a random collection of lessons.

The ***Technology*** portion of our 9th grade STEM curriculum was narrowed down to an electronics survey that went from circuit basics to transistors. That led nicely to a study of microcontrollers and coding.

The ***Engineering*** portion could not only be imbedded in the electronics units, but was going to be studied as its own entity via an engineering design study.

Since we knew the scope was broad but the calendar narrow, the curriculum had to include multiple learning goals within each set of units. After the methods unit, the major units were not sequential and would be taught by two instructors that could each teach half of the curriculum. A section of students would spend a semester with each of the teachers. We arrived at the following basic outline:

- The Methods of Science (Teacher A & B)
- Physics (Teacher A)
- Chemistry (Teacher B)
- Mechanical /Structural Engineering (A & B)
- Electronics & Microcontrollers (Teacher B)
- Engineering Design, CAD & Prototyping (Teacher A)



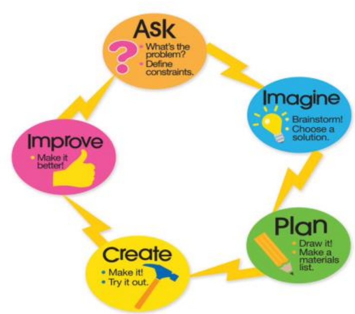
BLOCK PLANS (α VERSION EDITION)

Here are some of the units and lessons we used to try out meeting the design goals

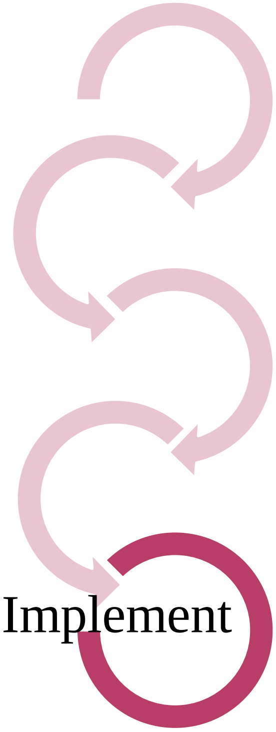
- 🛠 Mechanical /Structural Engineering (A & B)
  - For this study we used two traditional design challenges—the traditional popsicle stick bridge challenge taught around unit on structural designs and recognizing load paths and the egg drop challenge with an α and β version design process
- 🛠 Electronics & Microcontrollers (Teacher B)
  - The electronics unit is a home-made adaptation of a Bread board based circuitry curriculum called Tron-ix. [product link] This system has an excellent progression of skills and good components that are identical to what an engineer would use to prototype. The written materials give too many diagrams Allow students to complete projects without learning how to read schematic diagrams.

  - The second half of this unit combines electronics with microcontrollers. This circuits the students build can be programmed using code. I have settled on a kit from Sparkfun in Boulder CO that puts it together with some pretty sound pedagogy. It does have some of the same weaknesses as the Tron-ix materials with sheltering students from schematics, but still seemed pretty solid. [Link to my preferred kit]

- 🛠 Engineering Design, CAD & Prototyping (Teacher A)
  - Engineering Design Principles
  - CAD via tinkercad.com

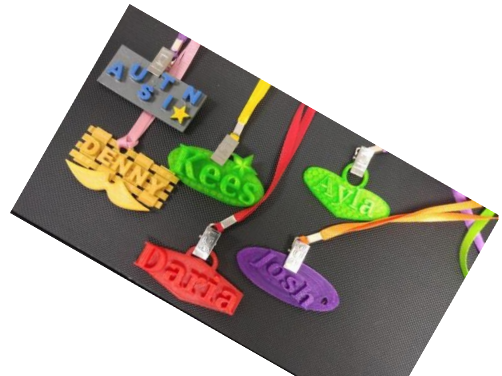


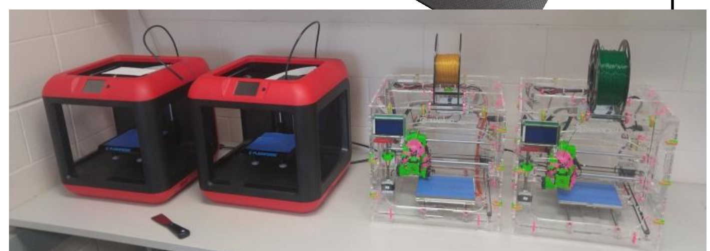
[CLICK]



THIS COMING YEAR & NEXT (WASH, RINSE, REPEAT)

[click for video]

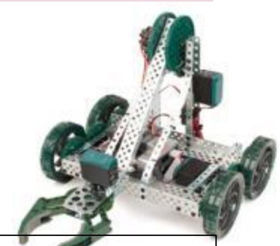




IF I WERE STARTING OVER…

Some areas that are worth further study and perhaps inclusion:

- A robotics club/competition
- The VEX EDR curriculum
- Project Lead the Way
- BBC Micro:bit
- Graphic Based code Language as an intro



Sample VEX

Click for video