

The Bare Minimum Engineer

Jul 20, 1969 • Christopher Kalitin

Longer title: How To Become The Bare Minimum Engineer

This is partly a play on Casey Handmer's [The Well-Rounded Engineer](#).

Slightly a continuation of How To Teach Engineering, though that was mostly just yapping.

Potential Structure:

1. Get on the Positive Slope Side of the J Curve
2. How To Become The Bear Minimum Engineer
3. How Not To Become The Bare Minimum Engineer

The Bare Minimum Engineer

1. Play on Handmer's The Well-Rounded Engineer
2. The MVP of an engineer, providing value
3. How to be a net positive in organizations

How to actually learn things and why university is inefficeint

1. Just take the conversation with Prashant and formalize it
2. Work from first principles
 - Maybe note that I'm just a student? Open to iteration? Or just go for it? Just go for it.
 - Get Mischa's feedback?
1. The Well-Rounded Engineer is about how to maximize your productive output as an engineer
2. The Bare Minimum Engineer is about how to get to the positive side of the J curve
3. As an engineering student, or young person in general, or new entrant into an engineering organization, you are a net negative
4. You need to acquire knowledge and skills to start becoming productive, and then pay back your debt to the organization (UBC Solar)

5. Be extremely curious, the side effects/fundamental reasons for this are extremely positive and increase your net lifetime productive output (increased salary too in engineering)

phenolphthalein

Prashant Conversation Notes:

- ○ ■ Hardest I've ever worked and not the most productive I've ever been
- ○ ■ You need to understand why you're studying something and what the goal is to know what information is important (Falcon 1 example toughness)
- ○ ■ Self motivation is extremely important. University's forcing function is extrinsic
- English class is low signal to noise ratio
- ○ ■ Projects that are useless in and of themselves are demotivating
- ○ ■ Complete focus = max productivity. Context switching is the mind killer (Joe Justice with Dr. Know It All Podcast)
- ○ ■ Feedback from people more intelligent than you is extremely important
- ○ ■ Classes mostly teach computations (maybe this changes in upper years) + tested on computations (MATH 120)
- ○ ■ Not incentivized to learn things from first principles
- ○ ■ Entry is based on grades
- ○ ■ First Principles give the side effect of being successful academically (mostly)
- ○ ■ No clue how to test. Do you even test? You have the real world, job interviews and portfolios are your test
- ○ ■ Back to the original point, you need to be able to integrate feedback
- ○ ■ First Principles and Projects show you why the things you're learning are important!
- ○ ■ High iteration rate is important and motivating, in contrast to fax projects
- ○ ■ Anecdotes? Nah fuck that I'm right. I am my father's son. Symptom of extreme self belief

Prashant conversation:

[7:24 PM]Christopher Kalitin: The paradigm of work in university is very inefficient [7:24

PM]Christopher Kalitin: This is simultaneously the hardest I've ever worked and not the most productive I've ever been [7:24 PM]Christopher Kalitin: In about the last year of online school I figured out the proper schedule to maximize productivity, sadly now this can't be utilized

- ○ ■ Hardest I've ever worked and not the most productive I've ever been

[7:27 PM]Christopher Kalitin: We're studying the fundamentals without knowing why, which

greatly decreases efficiency

You need to know why you're learning something to properly learn it

Like the transformer architecture, you need to weigh certain facts more highly than others - if you don't do this weighting you're far less efficient

The only way to figure out what knowledge to put more effort into learning than others (weights) is to actually do something, to build, this is why design teams are important

- ○ ■ You need to understand why you're studying something and what the goal is to know what information is important (Falcon 1 example toughness)

[7:30 PM]Christopher Kalitin: You have to be self motivated to do it, this is the problem with university, nearly everything you do is externally motivated "oh no I need to study for this test or I'll get a bad grade"

- ○ ■ Self motivation is extremely important. University's forcing function is extrinsic
- English class is low signal to noise ratio

[7:48 PM]Pieman: It's more about the practice and learning how to form a significant and debatable claim on a unresolved scientific issue [7:49 PM]Christopher Kalitin: Learning on projects that are useless in and of themselves - you need to write real research to know how to write research papers!

- ○ ■ Projects that are useless in and of themselves are demotivating

[8:55 PM]Christopher Kalitin: You'll get good at writing papers when its all you think about - this doesn't happen in school

- ○ ■ Complete focus = max productivity. Context switching is the mind killer (Joe Justice with Dr. Know It All Podcast)

[8:56 PM]Pieman: The willingness to wrote papers is all your own [8:57 PM]Christopher Kalitin: in university its extrinsic, not intrinsic

[8:57 PM]Pieman: Right you could have great ideas, but presenting them on paper is something that requires alot of practice [8:57 PM]Pieman: It requires the feedback of people that are more educated than yourself

- ○ ■ Feedback from people more intelligent than you is extremely important

[8:59 PM]Christopher Kalitin: In chem & physics this is especially clear because all you're tested on and learning is how to solve equations - and if you have a good teacher many some first principle understanding

On projects, all that matters is first principle understanding of the systems you're working on, do chemists actually solve stoichiometry problems by hand? This is extremely clear on ubc solar

- ○ ■ Classes mostly teach computations (maybe this changes in upper years) + tested on computations (MATH 120)

[9:04 PM]Christopher Kalitin: Exactly I completely agree

Now the more fundamental issue is that the incentive in university is not to learn things from first principles, it's to get a good grade on the test

I have a few friends like this and it's immensely sad and unproductive

Like you're saying, its a personal responsibility to try to understand things from first principles - personal responsibility, not academic responsibility

Your academic responsibility is to get a good grade on the test

- ○ ■ Not incentivized to learn things from first principles

[9:05 PM]Pieman: If you see uni through only the lense of the goal of grades I doubt your gonna go far [9:06 PM]Christopher Kalitin: Thats the way the university sees it

Your entry into the university is mainly by grades You getting into the program you want is mainly on grades Etc.

- ○ ■ Entry is based on grades

[9:06 PM]Christopher Kalitin: There is the side effect of understanding things from first principles that you'll probably do well on the tests - but this is not a direct effect, only a side effect

- ○ ■ First Principles give the side effect of being successful academically (mostly)

[9:07 PM]Christopher Kalitin: Sure, but they're optimised for testing you on quantitative computational knowledge, not testing you on your first principles understanding of various systems, which is what actually matters on technical projects

[9:07 PM]Pieman: How would you propose unis test us then [9:08 PM]Christopher Kalitin: No clue yet! for now I've only identified the massive problem

- ○ ■ No clue how to test. Do you even test? You have the real world, job interviews and portfolios are your test

[9:08 PM]Christopher Kalitin: I think the only way to really learn the way I want to is more personalised I've learned far more about engineering from my design team than the intro to

engineering class

[9:09 PM]Christopher Kalitin: This blog post taught me far more about why tedious engineering processes are important than class Because in class they're only tedious, and in reality you see why they're useful first hand

[9:11 PM]Christopher Kalitin: I should write a blog post to formalize all this, it'd be a lot clearer

The reason most feedback isn't useful is that you have no way to integrate it

Ideally, feedback would give you actionable insights - the defining feature of actionable insights is that you can take action now and see the direct benefits

Without a low iteration rate, you don't actually know if what you're doing works or what insights to look for

In school you are mostly aimless aside from just taking in random knowledge. And you don't know why thing random knowledge will be applicable

If you knew directly why you needed knowledge, you'd learn far more effectively

This is clear to me on design teams where I see why understanding the hall effect is actually useful in hardware, not just learning theoretically

- ○ ■ Back to the original point, you need to be able to integrate feedback
- ○ ■ First Principles and Projects show you why the things you're learning are important!

[9:11 PM]Pieman: Yeah I agree, I assume what your getting at is this is all ineficeint [9:13

PM]Christopher Kalitin: Over the summer I found the right way to do this One new project every 2 ish weeks First, spend a week learning the theoretical fundamentals, then apply it

The faster iteration speed is extremely important and I've proven it's more useful in that I'm a better at building real programming projects than most 4th years (I talk to them on ubc solar)

- ○ ■ High iteration rate is important and motivating, in contrast to fax projects

[9:13 PM]Pieman: I would keep in mind that experiences from what you learn in school can be anecdotal bur there are some common aspects [9:14 PM]Christopher Kalitin: Yes, there's a fundamental reason for this

The more efficient humanity is, the more we flourish - you can measure this flourishing directly through GDP or net global energy output

We will conquer Alpha Centauri faster if universities are more efficient

[9:15 PM]Christopher Kalitin: I've been leaning into anecdotes, they're far more actionable than vague data that you don't actually understand

Especially on the topic of if you stay in university to learn or not, personal anecdotes are extremely important because you live a personal life [9:15 PM]Christopher Kalitin: This has actually been an immensely useful I just laid out the structure of a blog post thanks

- ○ ■ Anecdotes? Nah fuck that I'm right. I am my father's son. Symptom of extreme self belief

 [Subscribe](#)

CKalitin

[x.com/CKalitin](https://twitter.com/CKalitin)

Geohot made a blog too. You should be working on hardware

