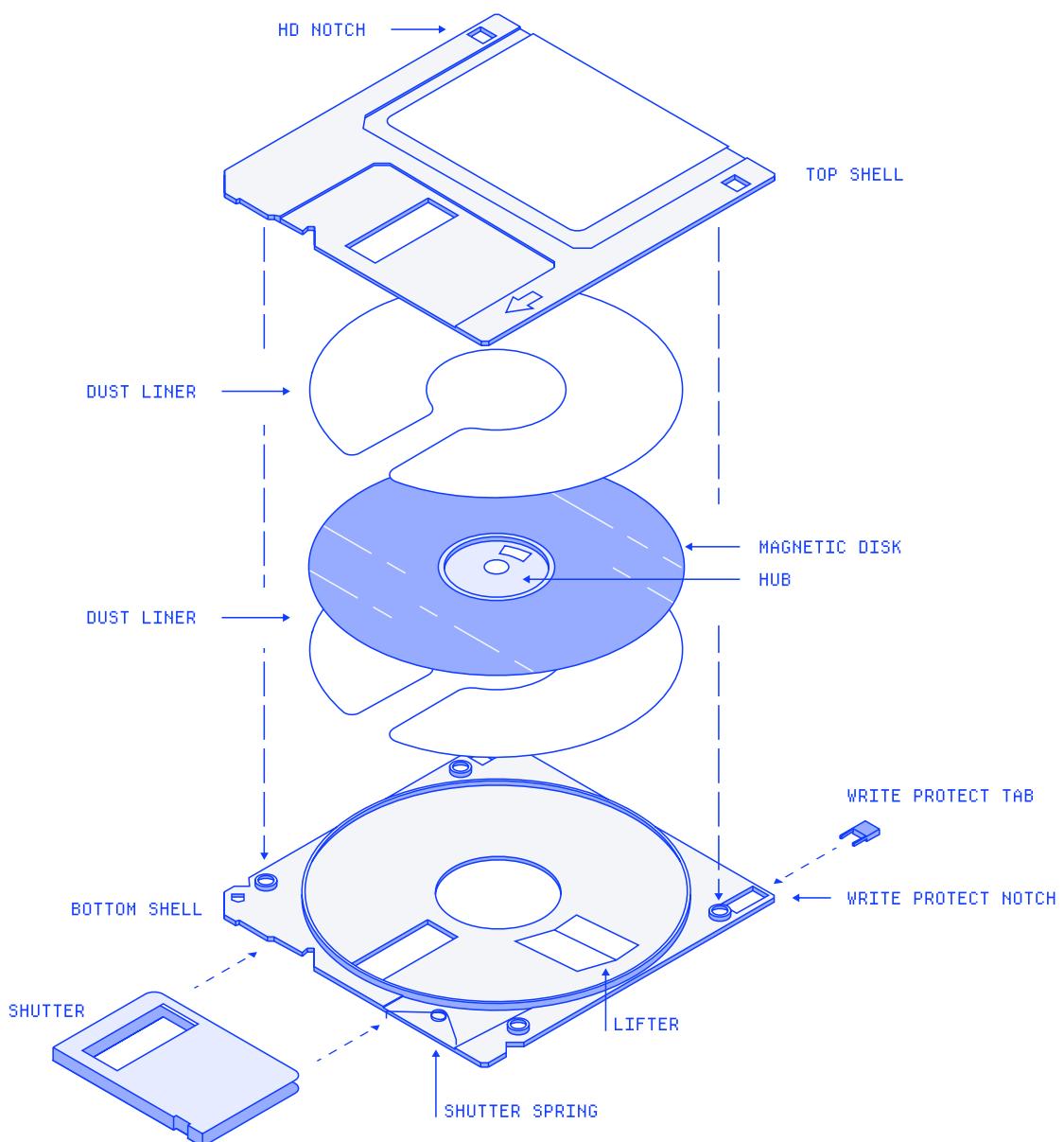


# MAKING SOFTWARE

Справочное руководство для разработчиков программного обеспечения.  
Автор текста и иллюстраций — Дэн Холлик.

FIG-001

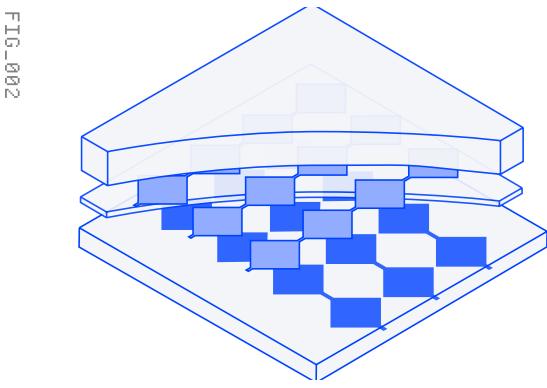
[ 3,5-дюймовая дискета ]



(c)  
1986

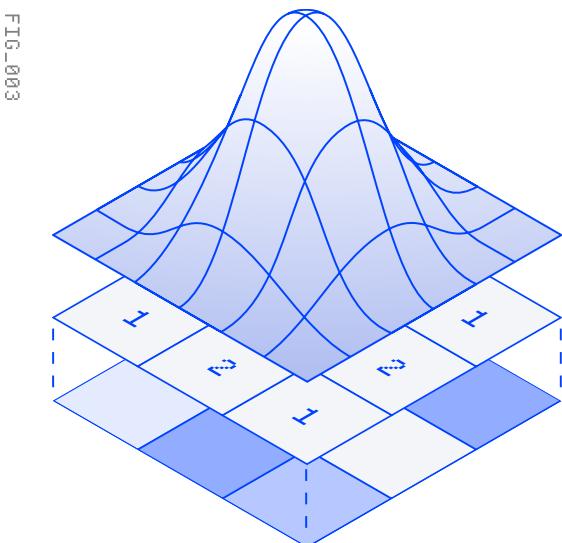
Вы когда-нибудь задумывались, как сенсорный экран понимает, что к нему прикасаются? Дело в том, что в дисплей встроены слои прозрачных металлических электродов. Когда палец приближается к экрану, он вызывает возмущение

магнитного поля, которое улавливают электроды.



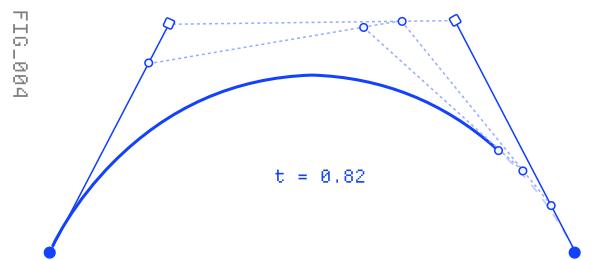
Поскольку электроды расположены в виде сетки, они могут передавать операционной системе координаты x и y источника помехи. Довольно удобно.

А может, вы задавались вопросом, почему мы называем это размытием по Гауссу? При размытии изображения мы просматриваем все соседние пиксели и умножаем их на матрицу весов, называемую ядром.



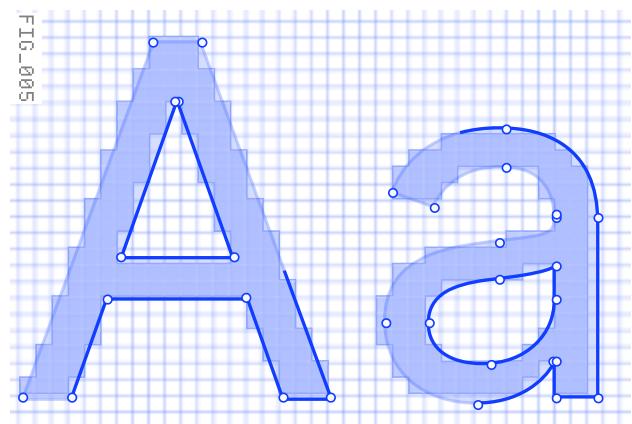
Наиболее распространённый тип ядра имеет гауссово распределение, то есть интенсивность ближе к центру выше, а по краям — ниже. Это позволяет добиться более реалистичного размытия без больших затрат на вычисления.

Возможно, вам всегда было интересно, как работает инструмент «Перо» в Figma и что на самом деле происходит с этими ручками, когда вы их перемещаете.



Они управляют точками на кривой Безье — это крутая математическая конструкция, которую мы используем для рисования кривых в векторной графике, например в шрифтах и SVG.

Но, конечно, наши экраны состоят из пикселей и с трудом отображают плавные линии. Поэтому нам нужно взять эти линии и придумать, как их отобразить, чтобы они максимально точно передавали форму.



Это называется растеризацией, но все не так просто, как кажется. Нам нужна целая куча хитроумных приемов, таких как сглаживание, чтобы наши глаза воспринимали изображение как прямые линии.

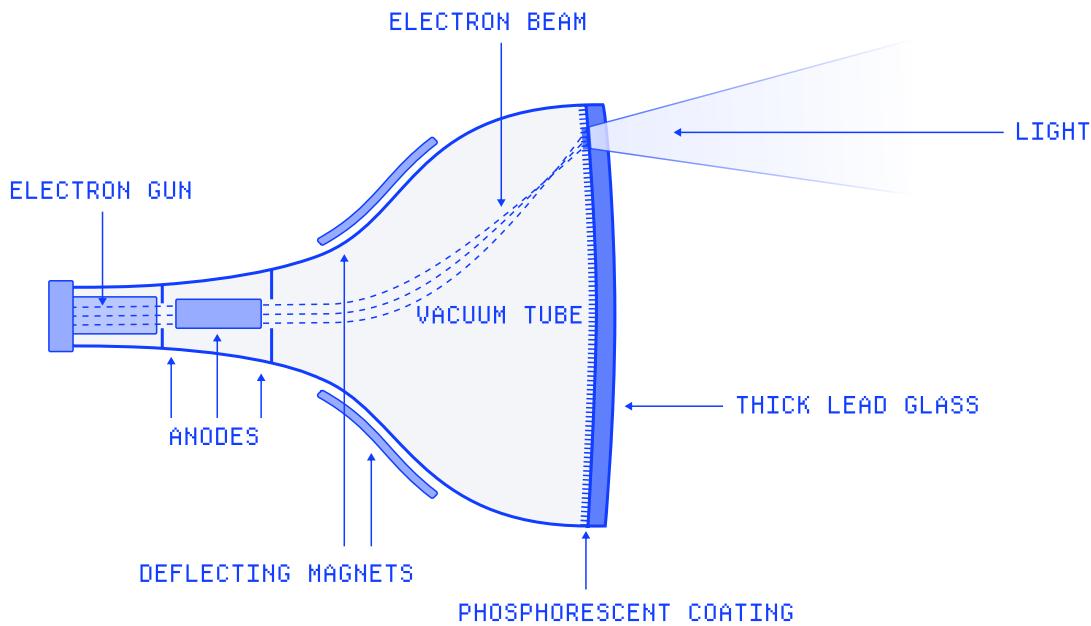
Если вы когда-либо задавались этими вопросами или они пробудили в вас любопытство, то эта статья для вас.

Эта книга не научит вас программировать — это не учебное пособие и не руководство, а нечто более интересное. Это руководство, в котором объясняется, как на самом деле работают вещи, которыми вы пользуетесь каждый день.

Завтра вы не станете лучшим дизайнером или программистом — здесь нет ничего полезного для практики. Но знание того, как все устроено, пригодится, когда вы окажетесь не в своей тарелке. Или, по крайней мере, вы сможете притвориться умным в глазах друзей.

Чтобы разобраться в этом материале, не нужно быть техническим специалистом — здесь много картинок и схем, которые сделают всю сложную работу за вас. Вам просто нужно быть любознательным.

FIG\_006



### SIGNUP FOR UPDATES:

*I promise to only email you cool shit. Draft chapters, progress updates, sneak peaks at illustrations I'm working on. Stuff like that.*

ENTER YOUR EMAIL

JOIN MAILING LIST

### TABLE OF CONTENTS [v1.0]

[ SECTIONS: 8 · CHAPTERS: 36 ]

#### 1. PIXELS AND COLOR

- How does a screen work? ..... 3.6K WORDS

- What is a color space? ..... 6.2K WORDS
- Color contrast.
- Blending modes.
- Digital images.
- Touch screens.

## 2. FONTS AND VECTORS

- Drawing curves.
- How to make a font.
- Rasterisation and anti-aliasing.
- Scalable Vector Graphics.
- Boolean operations.

## 3. 3D AND GRAPHICS

- How does a GPU work?
- Shaders. ..... 4.6K WORDS
- Rays and SDFs.
- Blurs, noise and other effects.
- 3D projection.

## 4. AI AND ML

- Neural nets and transformers.
- Gradient descent and backpropagation.
- Embeddings and attention.
- Generating images.

## 5. DATA AND COMPRESSION

- Bits, bytes and binary.
- Entropy and compression.
- Image compression.
- Cryptography.
- How is data stored?

## 6. NETWORKING AND THE WEB

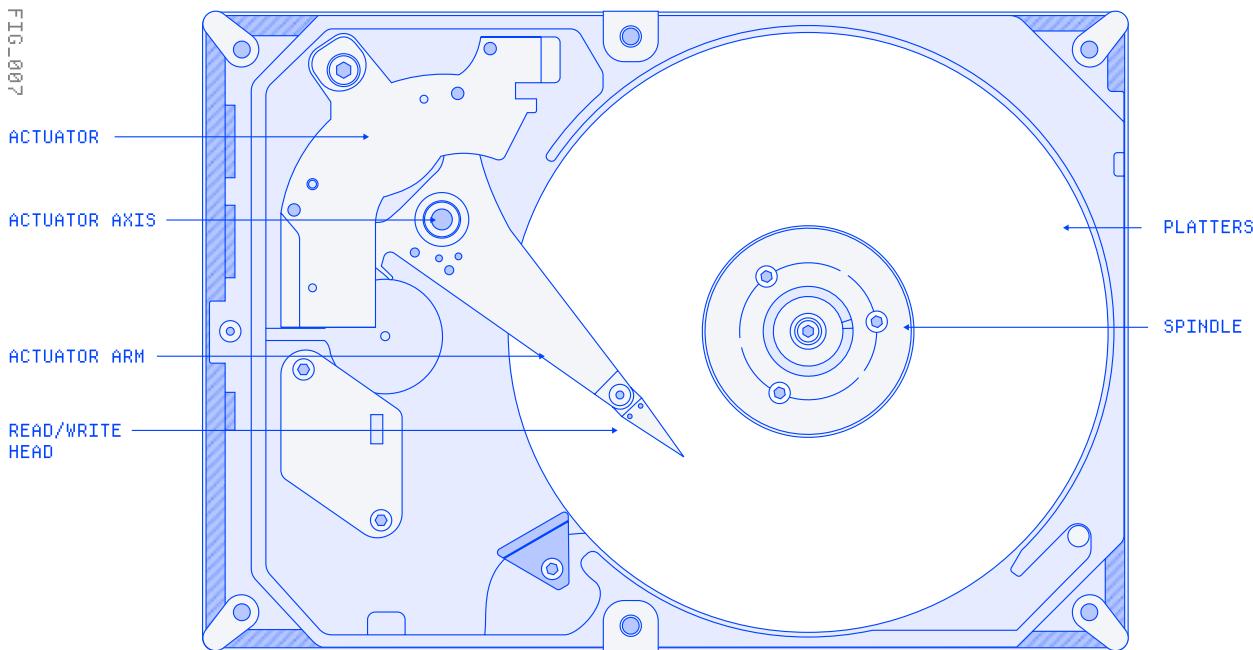
- Sending and receiving data.
- How the internet works.
- Servers and the cloud.
- What is a browser?

## 7. COMPILERS AND INTERPRETERS

- What is code?
- What is an interpreter?
- What is a compiler?
- What makes code fast?

## 8. MISC

- Regular Expressions.
- QR and barcodes.
- Quantum computing.



~~What does it do?~~  
~~How do you make the book?~~

## COMMON QUESTIONS

When will it launch?

Is this a physical book?

↑ ↓

How much will it cost?

~~What does it do?~~

~~How do you make the book?~~

IN: IS THIS A PHYSICAL BOOK?

OUT: No, it's a digital book *for now*.

Once it's fully ready, I'm going to do a print run but there's a ton of work involved in making a physical book and I want to do it properly.

I want it to be like a *really fucking nice* coffee table style book.

People who buy the advanced digital edition will get first dibs on pre-ordering any print editions, so if that's something

you're interested in, make sure  
you get that one.

\*\*\*

### SIGNUP FOR UPDATES:

*I promise to only email you cool shit. Draft chapters, progress updates,  
sneak peaks at illustrations I'm working on. Stuff like that.*

| MAKING SOFTWARE |

A REFERENCE MANUAL FOR PEOPLE WHO DESIGN AND BUILD SOFTWARE.  
WRITTEN AND ILLUSTRATED BY DAN HOLICK

| © 1990 |

CURRENT PROGRESS: