Hardwareimplementierung kryptographischer Algorithmen

Ein kleiner Einblick

Jan Kossick
Fakultät Informatik
TU Dresden

Geschichte der Kryptografie

Von den Anfängen bis zum Ende des 19. Jahrhunderts
Kerckhoffs-Prinzip und One-Time-Pad
Erste Hardwareimplementierungen - der 2. Weltkrieg
Computer vs. Verschlüsselungsstrategien

Kryptografische Hardware kurz vorgestellt

(Co-)Prozessoren, SmartCards, USB, RFID

Implementierung am Beispiel - RFID und ECC

Ein paar Worte zu ECC
Grundsätze für das Design von Implementierungen
Die Implementierung und ihre Effizienz

Sicherheit von Hardware

Die Zukunft der Hardwareverschlüsselung

Geschichte der Kryptografie

Von den Anfängen bis zum Ende des 19. Jahrhunderts

Kerckhoffs-Prinzip und One-Time-Pad

Erste Hardwareimplementierungen - der 2. Weltkrieg

Computer vs. Verschlüsselungsstrategien

Die Anfänge

1900 v. Chr.

Ägyptische Hieroglyphen

600 v. Chr.

1900 v. Chr. 600 v. Chr.

Hebräische Inschrift

600 v. Chr.

1900 v. Chr. 600 v. Chr.

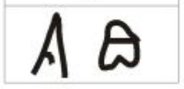
Skytala der Griechen

Mittelalter

Alphabetum Kaldeorum

(nach einer Handschrift von 1428, München, Univ.-Bibl. Cod. 4° 810, fol. 41v)

a	b	c	d	e	f	g	h
							

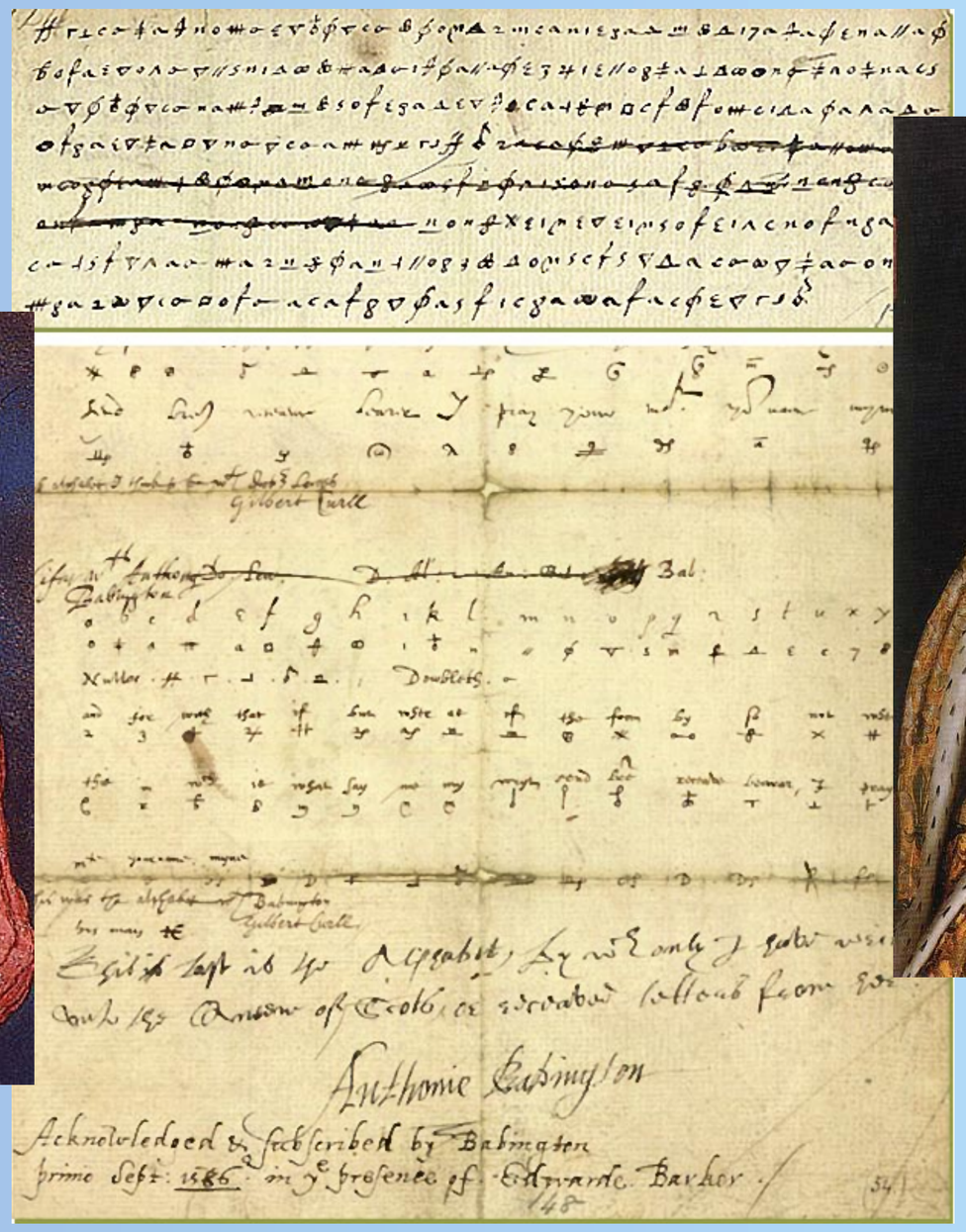
i	k	l	m	n	o	p	q
							

r	s	t	u,v	x	y	z
						

16. Jahrhundert

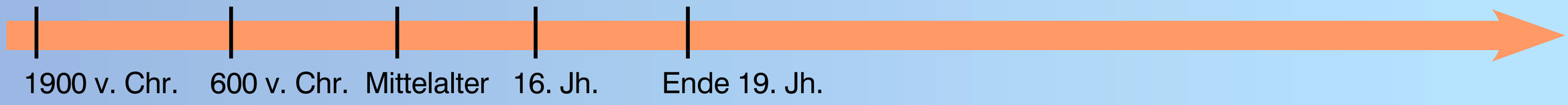
1900 v. Chr. | 600 v. Chr. | Mittelalter | 16. Jh.

Maria Stuart



Königin Elisabeth I.

Kerckhoffs-Prinzip und One-Time-Pad

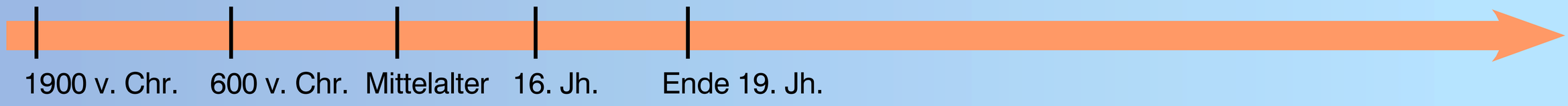


Kerckhoffs-Prinzip und One-Time-Pad

„Die Sicherheit eines Kryptosystems darf nicht von der Geheimhaltung des Algorithmus abhängen. Die Sicherheit gründet sich nur auf die Geheimhaltung des Schlüssels.“

Auguste Kerckhoffs von Nieuwendorf, 1883

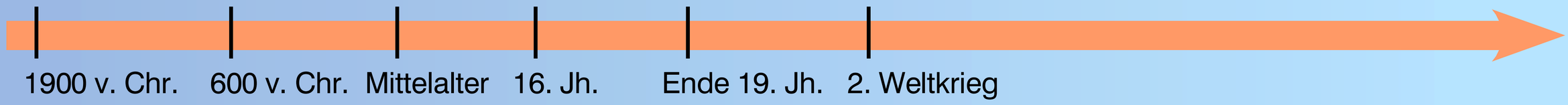
Kerckhoffs-Prinzip und One-Time-Pad



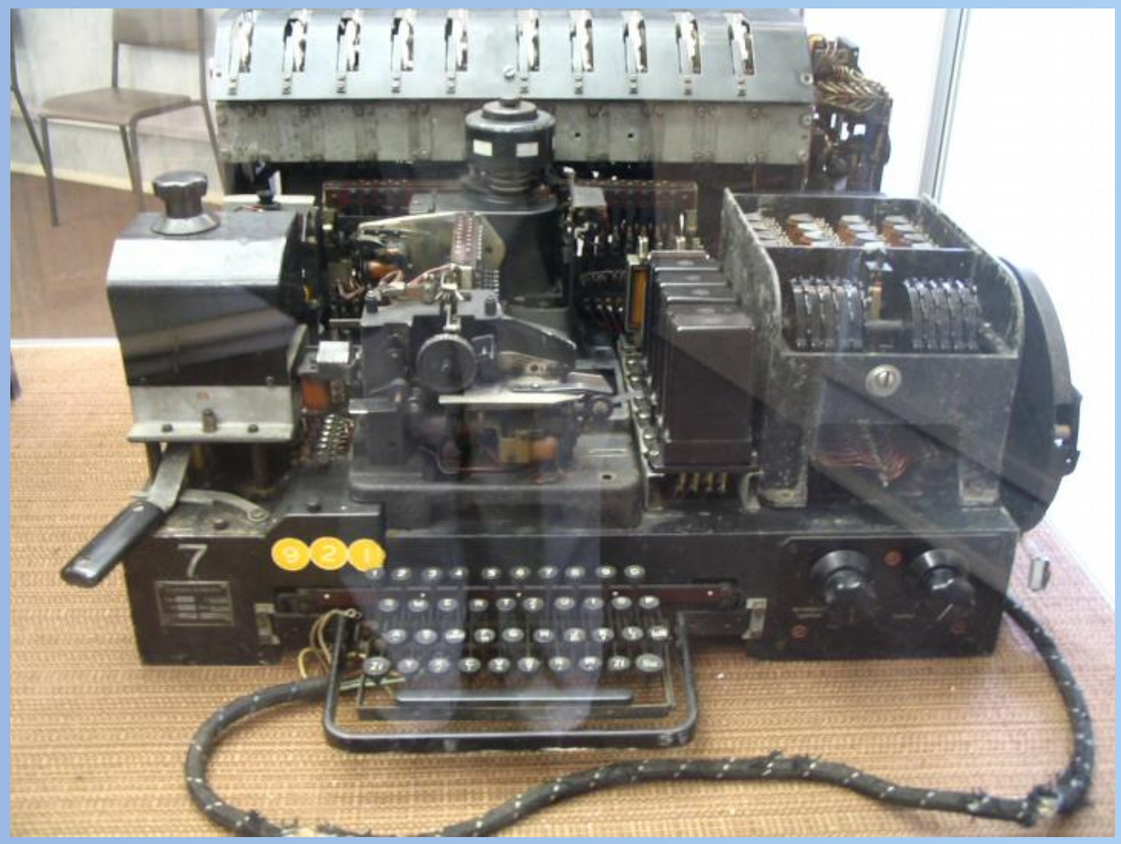
Gilbert Vernam, 1918

Verschlüsselung von Texten mit Schlüsseln, deren Länge gleich der Textlänge ist. Die Verschlüsselung ist beweisbar informationstheoretisch nicht entschlüsselbar ohne Kenntnis des Schlüssels.

Der 2. Weltkrieg



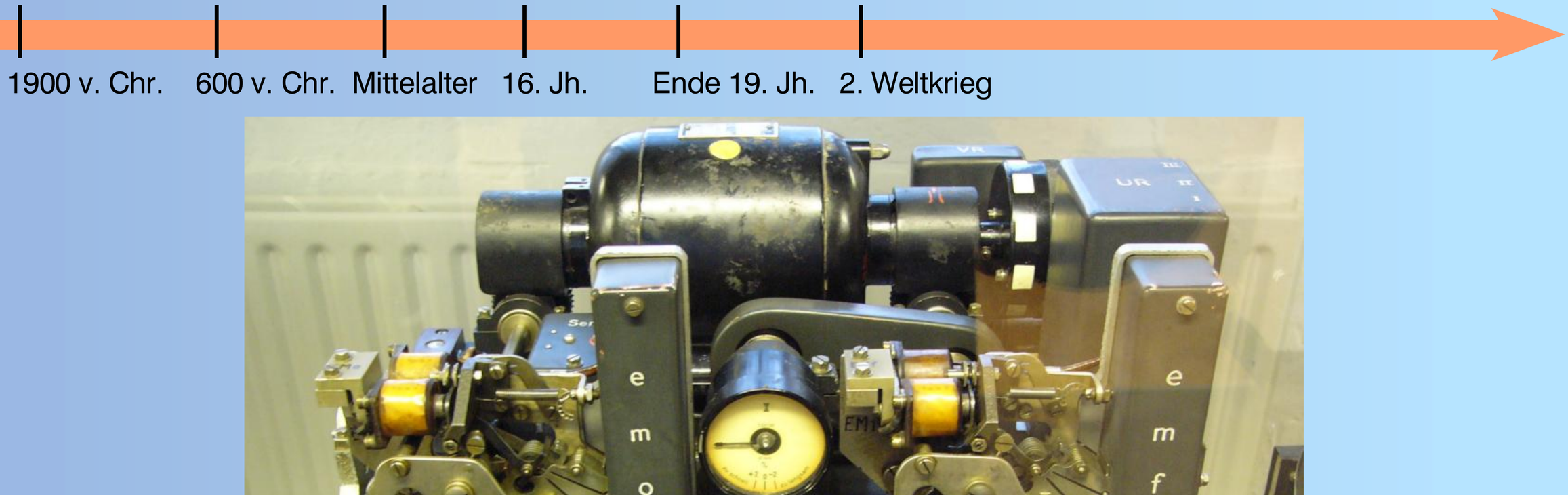
Der 2. Weltkrieg



Siemens T52

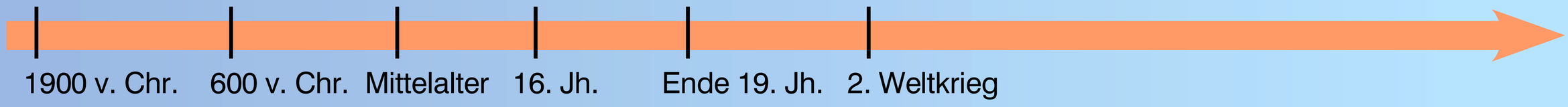
Enigma

Der 2. Weltkrieg



Lorenz SZ42

Der 2. Weltkrieg

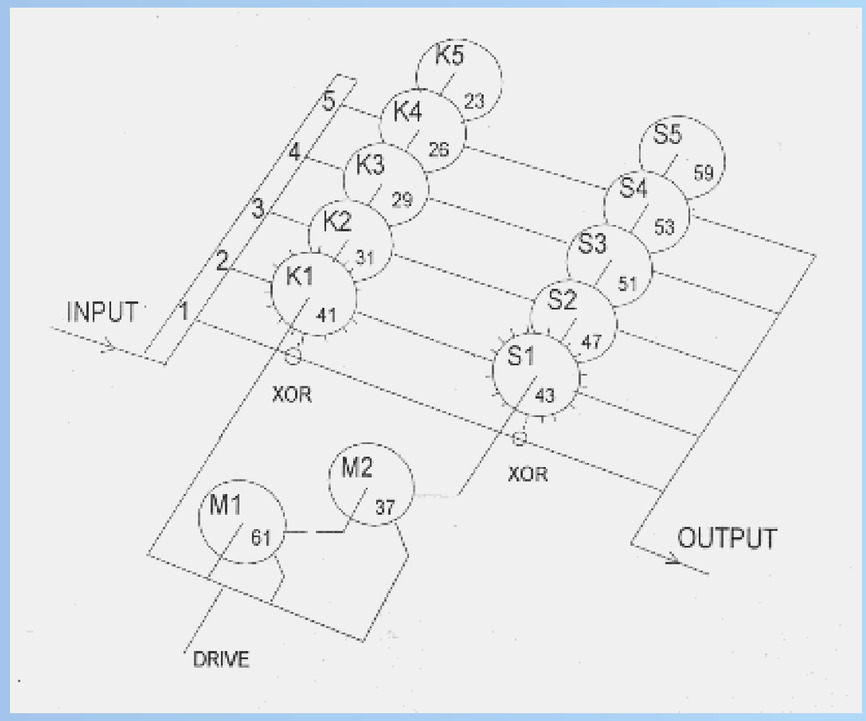


Baudot-Murray-Code

Code	Buchstaben	Ziffern/Zeichen	Code	Buchstaben	Ziffern/Zeichen
00011	A	-	10111	Q	1
11001	B	?	01010	R	4
01110	C	:	00101	S	'
01001	D	Wer Da?	10000	T	5
00001	E	3	00111	U	7
01101	F	unbenutzt	11110	V	=
11010	G	unbenutzt	10011	W	2
10100	H	unbenutzt	11101	X	/
00110	I	8	10101	Y	6
01011	J	Klingel	10001	Z	+
01111	K	(	01000	Wagenrücklauf	
10010	L	)	00010	Zeilenvorschub	
11100	M	.	00100	Leerzeichen	
01100	N	,	11111	Umschaltung Buchstaben	
11000	O	9	11011	Umschaltung Ziffern/Zeichen	
10110	P	0	00000	unbenutzt	

Der 2. Weltkrieg

1900 v. Chr. | 600 v. Chr. | Mittelalter | 16. Jh. | Ende 19. Jh. | 2. Weltkrieg

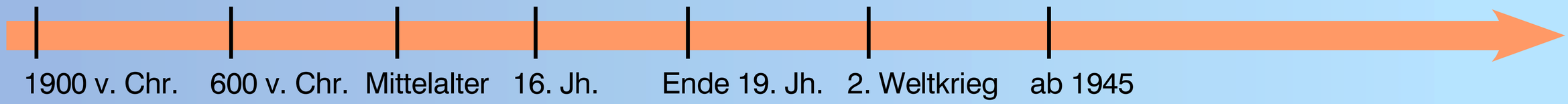


Computer vs. Verschlüsselung

Computer vs. Verschlüsselung

Entwicklung von Colossus, Zuse Z3, ENIAC

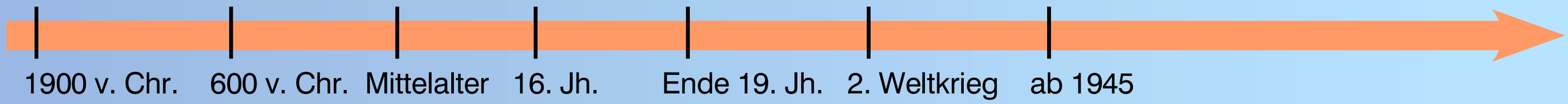
Computer vs. Verschlüsselung



Entwicklung von Colossus, Zuse Z3, ENIAC

→ Entschlüsselung chiffrierter Texte in annehmbarer Zeit

Computer vs. Verschlüsselung

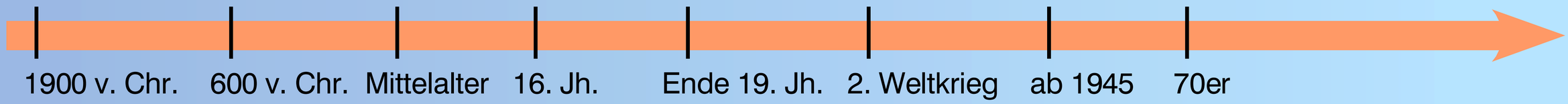


Entwicklung von Colossus, Zuse Z3, ENIAC

→ Entschlüsselung chiffrierter Texte in annehmbarer Zeit

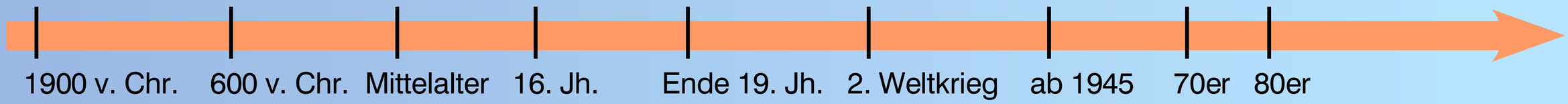
→ neue Verschlüsselungsalgorithmen mit dem Ziel auch auf Computern nicht in annehmbarer Zeit gelöst zu werden

Computer vs. Verschlüsselung



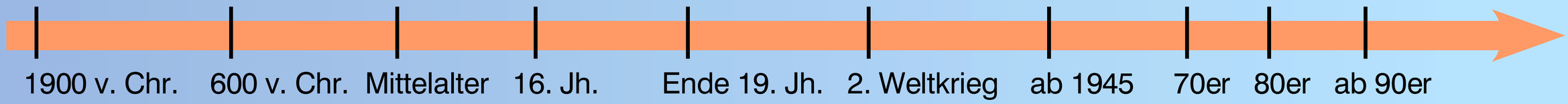
- Entwicklung „starker“ Algorithmen (1976: DES, 1977: RSA)
  - Prozessoren zu langsam für zusätzliche Aufgaben
    → Kryptografische Coprozessoren

Computer vs. Verschlüsselung



- Entwicklung „starker“ Algorithmen (1976: DES, 1977: RSA)
  - Prozessoren zu langsam für zusätzliche Aufgaben
    → Kryptografische Coprozessoren
- Prozessoren werden leistungsfähiger
  → Softwareimplementierungen werden möglich
- Ende 80er: Entwicklung ECC und HyperECC

Computer vs. Verschlüsselung



- Entwicklung „starker“ Algorithmen (1976: DES, 1977: RSA)
  - Prozessoren zu langsam für zusätzliche Aufgaben
    → Kryptografische Coprozessoren

- Prozessoren werden leistungsfähiger
  → Softwareimplementierungen werden möglich
- Ende 80er: Entwicklung ECC und HyperECC

- immer größere Verbreitung von Computern
  → Computer nicht für Sicherheit ausgelegt
  → Notwendigkeit guter, schneller Verschlüsselung
- 2001: Ablösung von DES durch AES

Kryptografische Hardware kurz vorgestellt

(Co-)Prozessoren

SmartCards

USB

RFID

(Co-)Prozessoren

Coprozessoren

- zur Entlastung von Hauptprozessoren
- speziell für kryptografische Algorithmen ausgelegt
- benötigen Software (Treiber)

SmartCards

Coprozessoren

- zur Entlastung von Hauptprozessoren
- speziell für kryptografische Algorithmen ausgelegt
- benötigen Software (Treiber)

SmartCards

- Sicherheit mit wenig Stromverbrauch und wenig Platzbedarf

Coprozessoren

- zur Entlastung von Hauptprozessoren
- speziell für kryptografische Algorithmen ausgelegt
- benötigen Software (Treiber)

SmartCards

- Sicherheit mit wenig Stromverbrauch und wenig Platzbedarf

USB Geräte

- Sicherheitsaspekte ähnlich wie SmartCards

Coprozessoren

- zur Entlastung von Hauptprozessoren
- speziell für kryptografische Algorithmen ausgelegt
- benötigen Software (Treiber)

SmartCards

- Sicherheit mit wenig Stromverbrauch und wenig Platzbedarf

USB Geräte

- Sicherheitsaspekte ähnlich wie SmartCards

RFID

- minimale Platz- und Energiereserven
- kontaktlose Datenübertragung → Abhörgefahr
- hohe Anforderung an Sicherheit und Effizienz der Umsetzung

Implementierung am Beispiel - RFID und ECC

Ein paar Worte zu ECC

Grundsätze für das Design von Implementierungen

Die Implementierung und ihre Effizienz

Anforderungen?

Wie schnell muss die Hardware arbeiten?

Anforderungen?

Wie schnell muss die Hardware arbeiten?

Wie viel Energie darf die Hardware verbrauchen?

Wie hoch darf die maximale Stromaufnahme sein?

Anforderungen?

Wie schnell muss die Hardware arbeiten?

Wie viel Energie darf die Hardware verbrauchen?

Wie hoch darf die maximale Stromaufnahme sein?

Wie viel Platz darf die Hardware einnehmen?

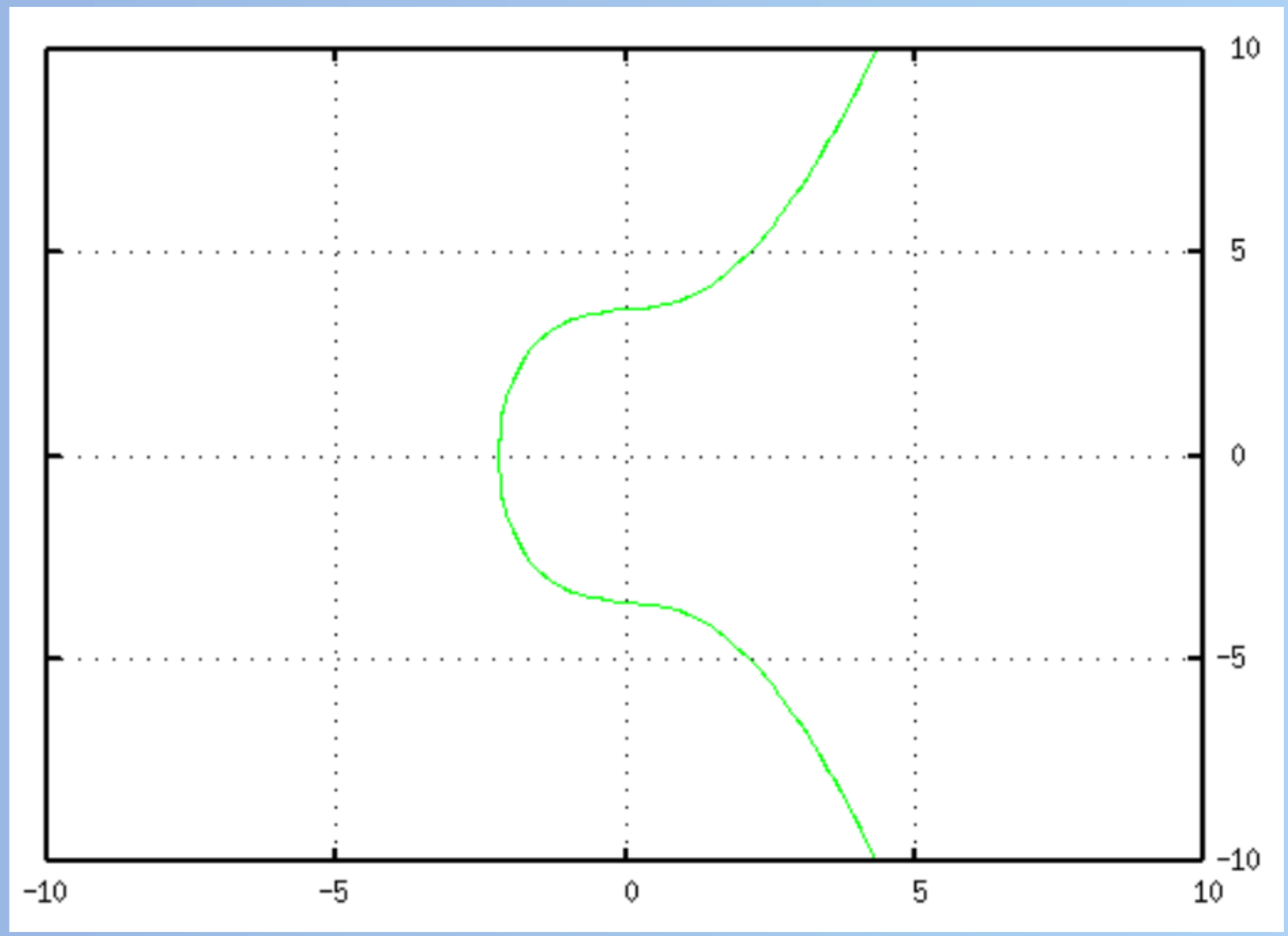
RFID und ECC

Ein paar Worte zu ECC

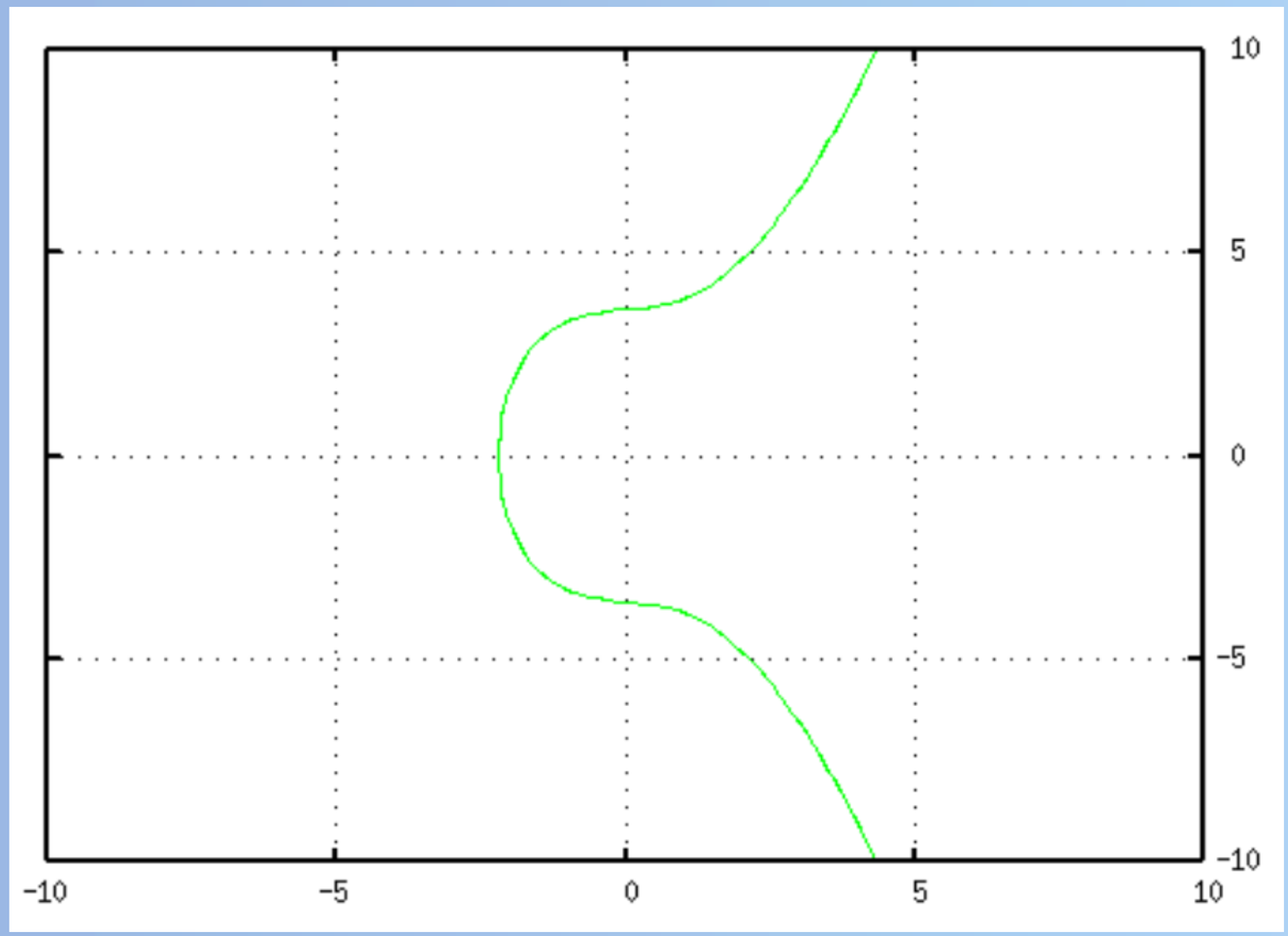
- asymmetrisch
- Verwendung des Problems des diskreten Logarithmus' über elliptischen Kurven (ECDLP)

ECDLP

$E = (G, \circ)$



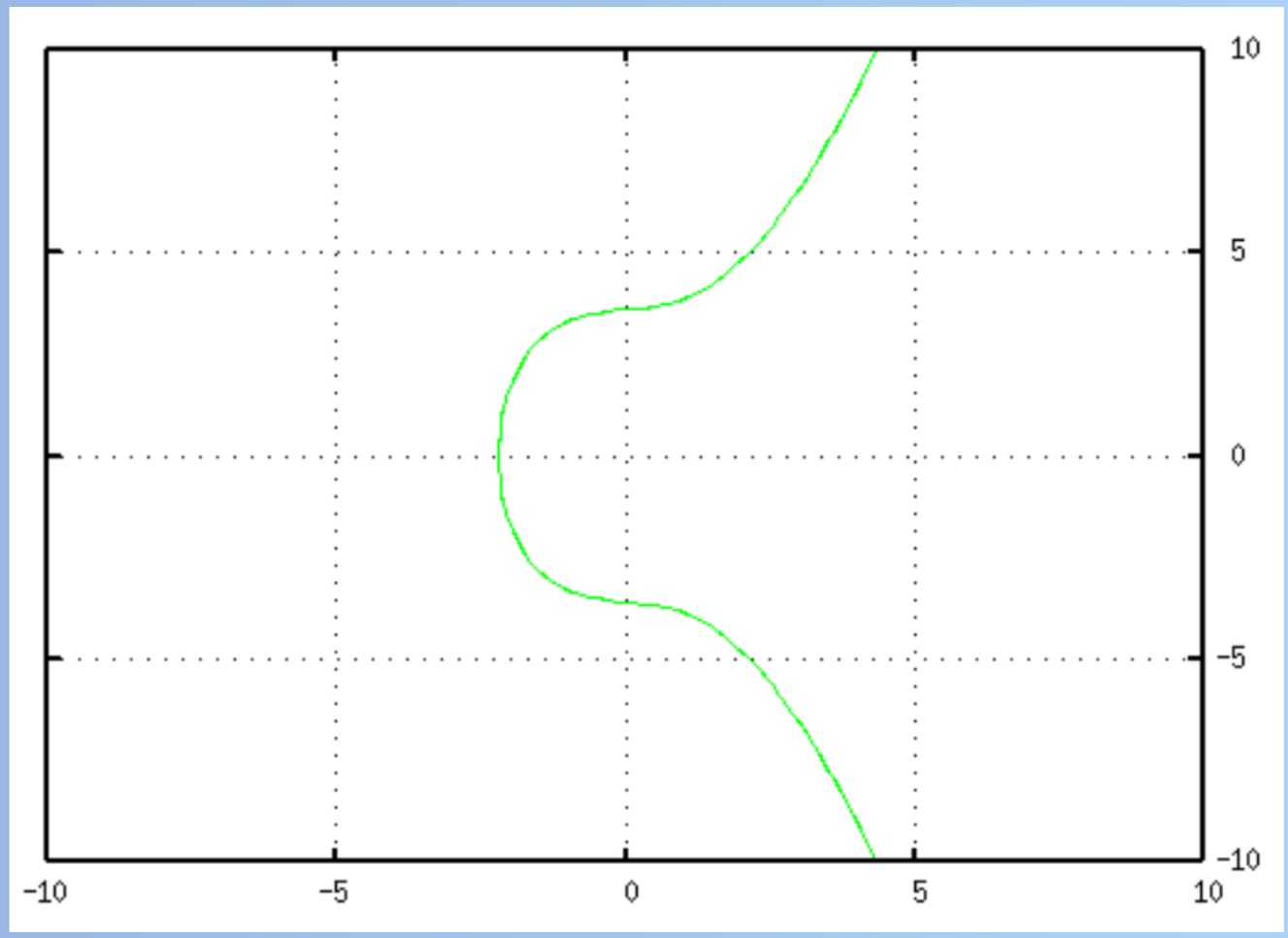
ECDLP



$E = (G, \circ)$

$g, h, x \in E$

ECDLP

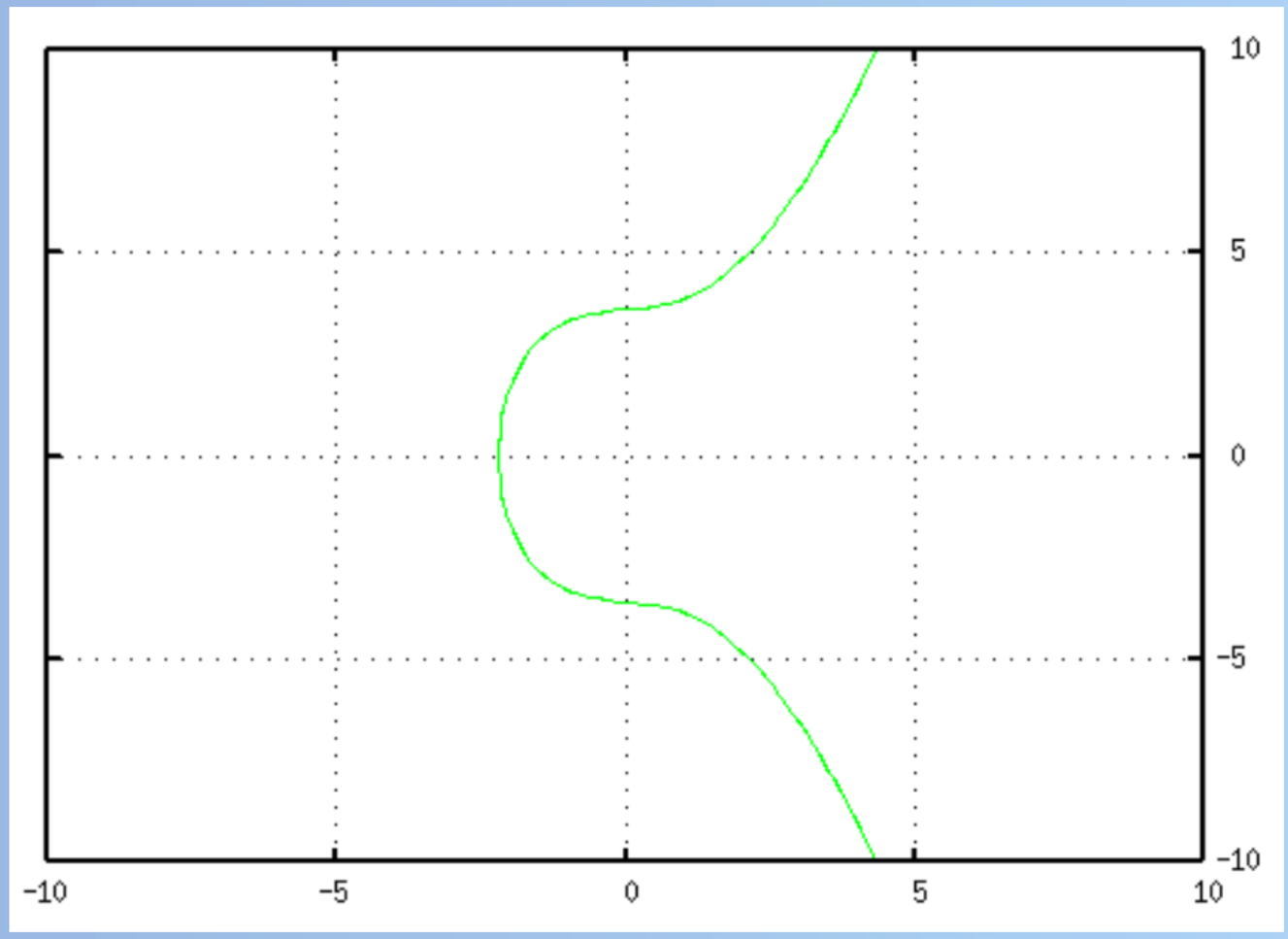


$E = (G, \circ)$

$g, h, x \in E$

$g^x = h \quad (x = \log_g(h))$

ECDLP



$E = (G, \circ)$

$g, h, x \in E$

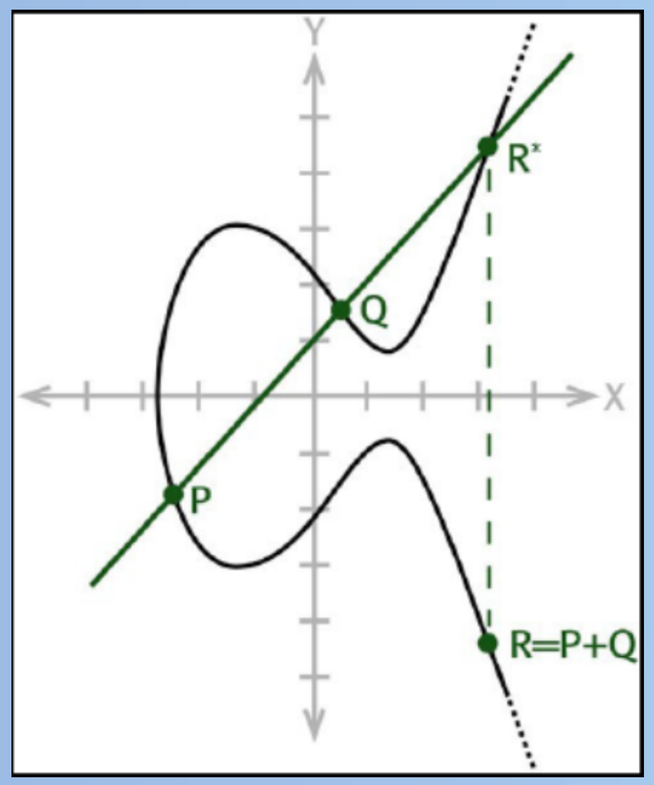
$g^x = h$

alt. Schreibweise:

$P, Q, x \in E$

$Q = xP$

Graphische Punktaddition



Beispielkurve:

$0 = y^2 + x^3 + ax + b$

Graphische Punktaddition:

Gerade durch Punkt P und Punkt Q → Punkt R*

Spiegelung an der x-Achse

→ Punkt R = Q + P

Datenaustausch

gegeben:

- elliptische Kurve
- genutzte Körper
- Basispunkt P
- zufälliger privater Schlüssel von Person A: x_A
- öffentlicher Schlüssel von Person A: $Q_A = x_A P$
- analog Person B

Ablauf:

- Person A berechnet $R_A = x_A Q_B$
- Person B berechnet $R_B = x_B Q_A$
- Schlüssel R: $R_A = x_A Q_B = x_A x_B P = x_B x_A P = x_B Q_A = R_B$
- Person C hat nur P, Q_A, Q_B und R und muss x_A oder x_B berechnen → genauso aufwendig wie das ECDLP

Grundsätze für das Design von Implementierungen am vorliegenden Beispiel

Anforderungen

Anforderungen:

- geringe Größe und wenig Stromaufnahme
- Datendurchsatz/Geschwindigkeit zweitrangig

Art der Hardware

Anforderungen:

- geringe Größe und wenig Stromaufnahme
- Datendurchsatz/Geschwindigkeit zweitrangig

Art der Hardware

- festverdrahtete Schaltkreise (ASIC)
- softwaregesteuerte Microprozessoren
- rekonfigurierbare Schaltkreise (FPGA)

Architektur

Anforderungen:
- geringe Größe und wenig Stromaufnahme
- Datendurchsatz/Geschwindigkeit zweitrangig

Art der Hardware
- festverdrahtete Schaltkreise (ASIC)
- softwaregesteuerte Microprozessoren
- rekonfigurierbare Schaltkreise (FPGA)

Architektur
- Größe des Schaltkreises minimieren
- Taktfrequenz möglichst klein halten
- Schaltvorgänge beschränken

Mathematische Grundlagen

Input: Integer $x > 0$ und Punkt P

Output: $Q = xP$

$x \leftarrow (x_{n-1}, ..., x_1, x_0)_2$

$Q \leftarrow P$

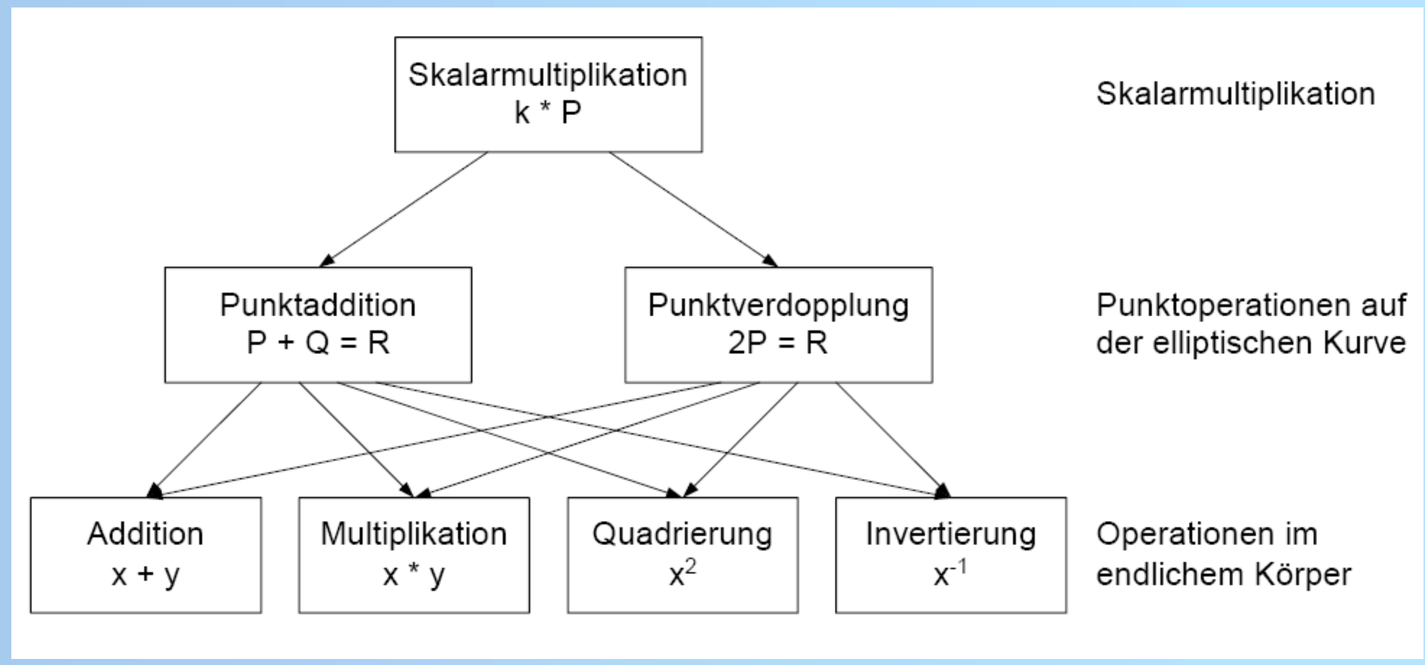
For i from (n – 2) downto 0 do

$Q \leftarrow 2 * Q$

If $x_i = 1$ then $Q \leftarrow Q + P$

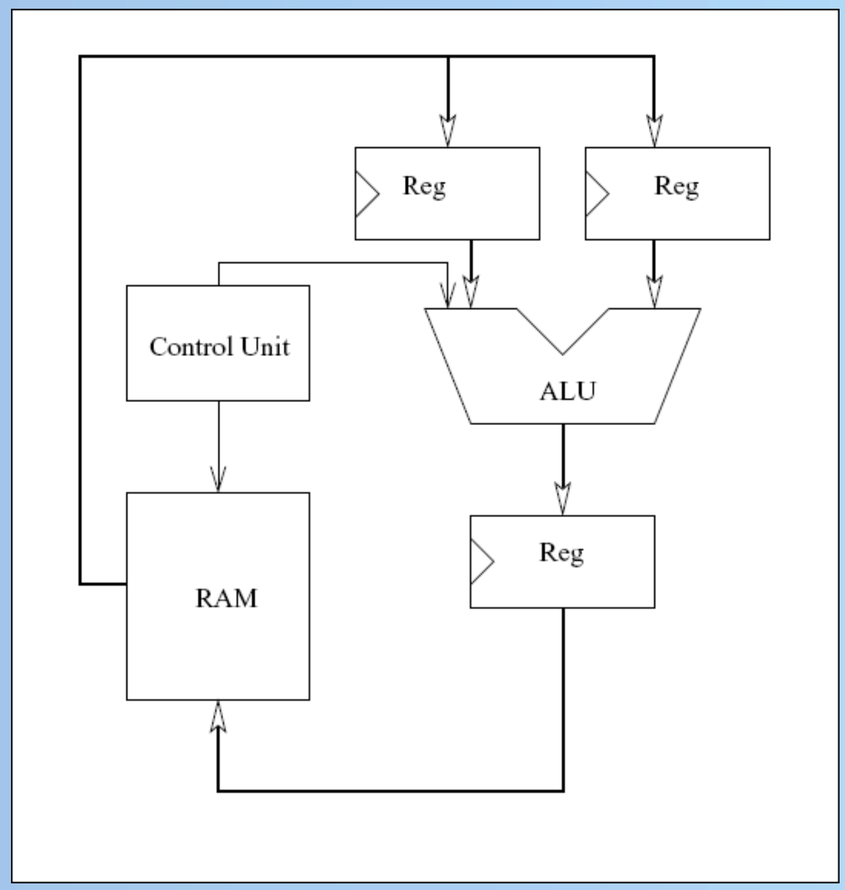
EndFor

Return Q

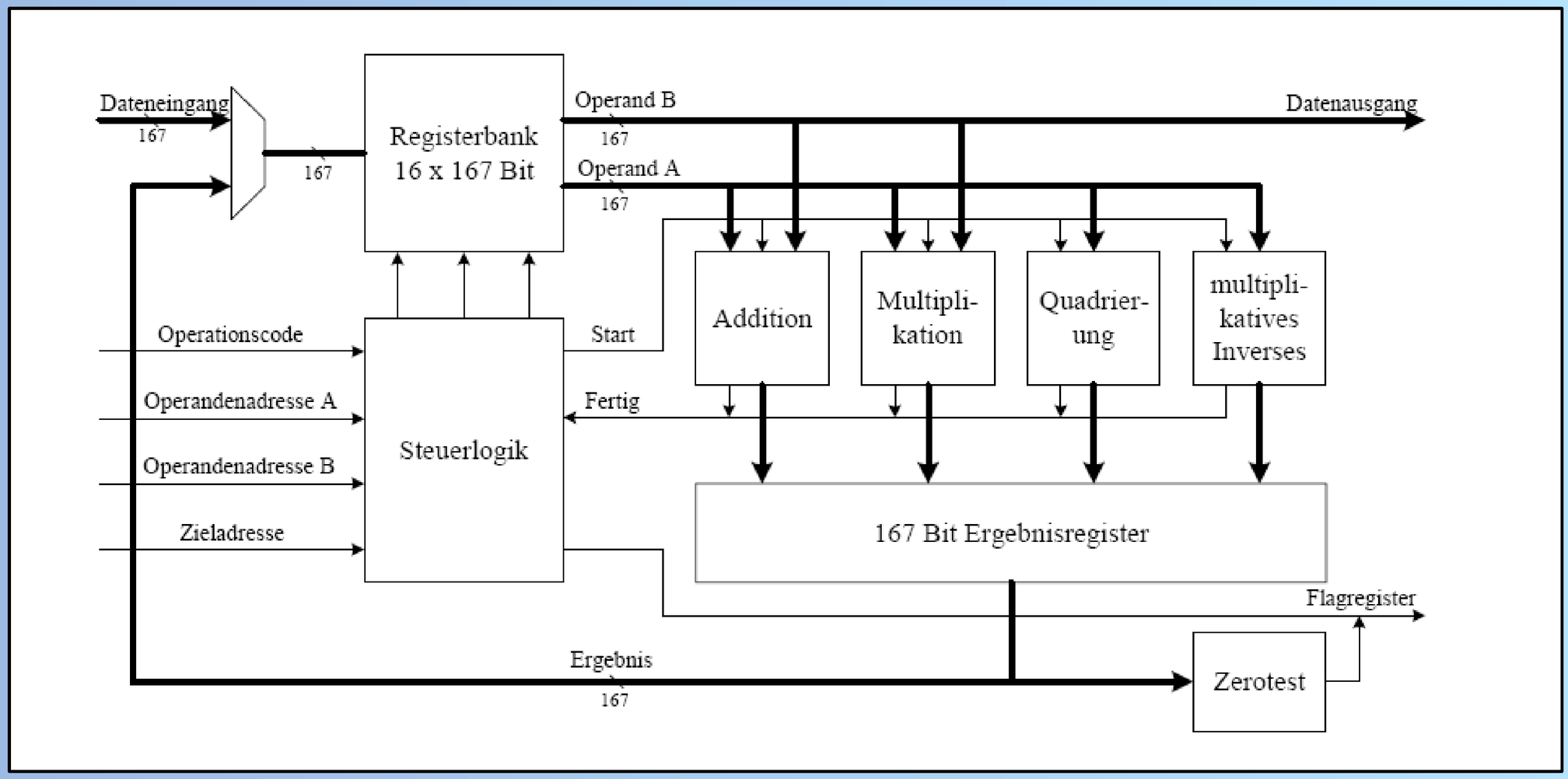


Elliptic Curve Processor

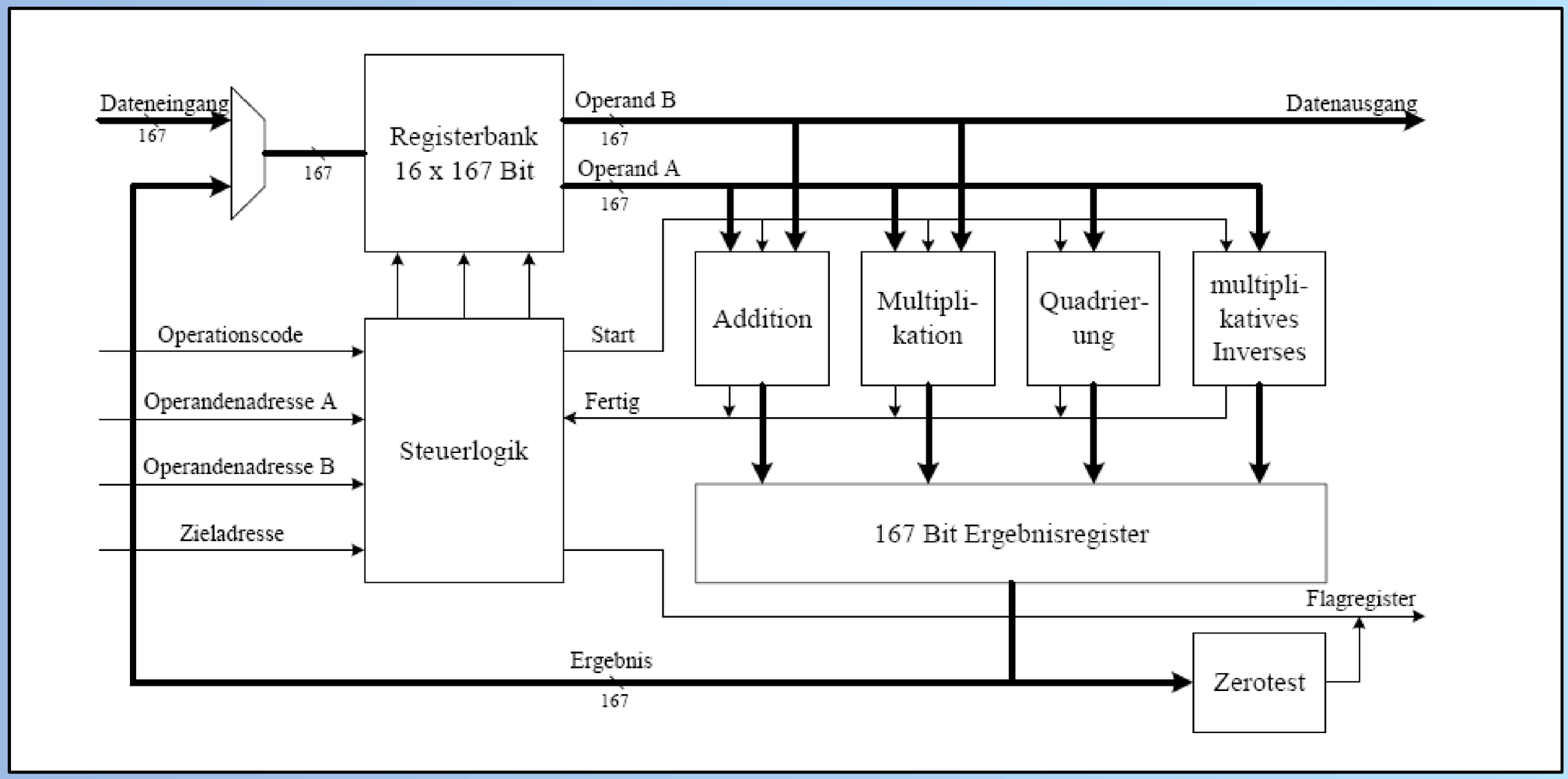
Elliptic Curve Processor



Arithmetische Einheit ALU

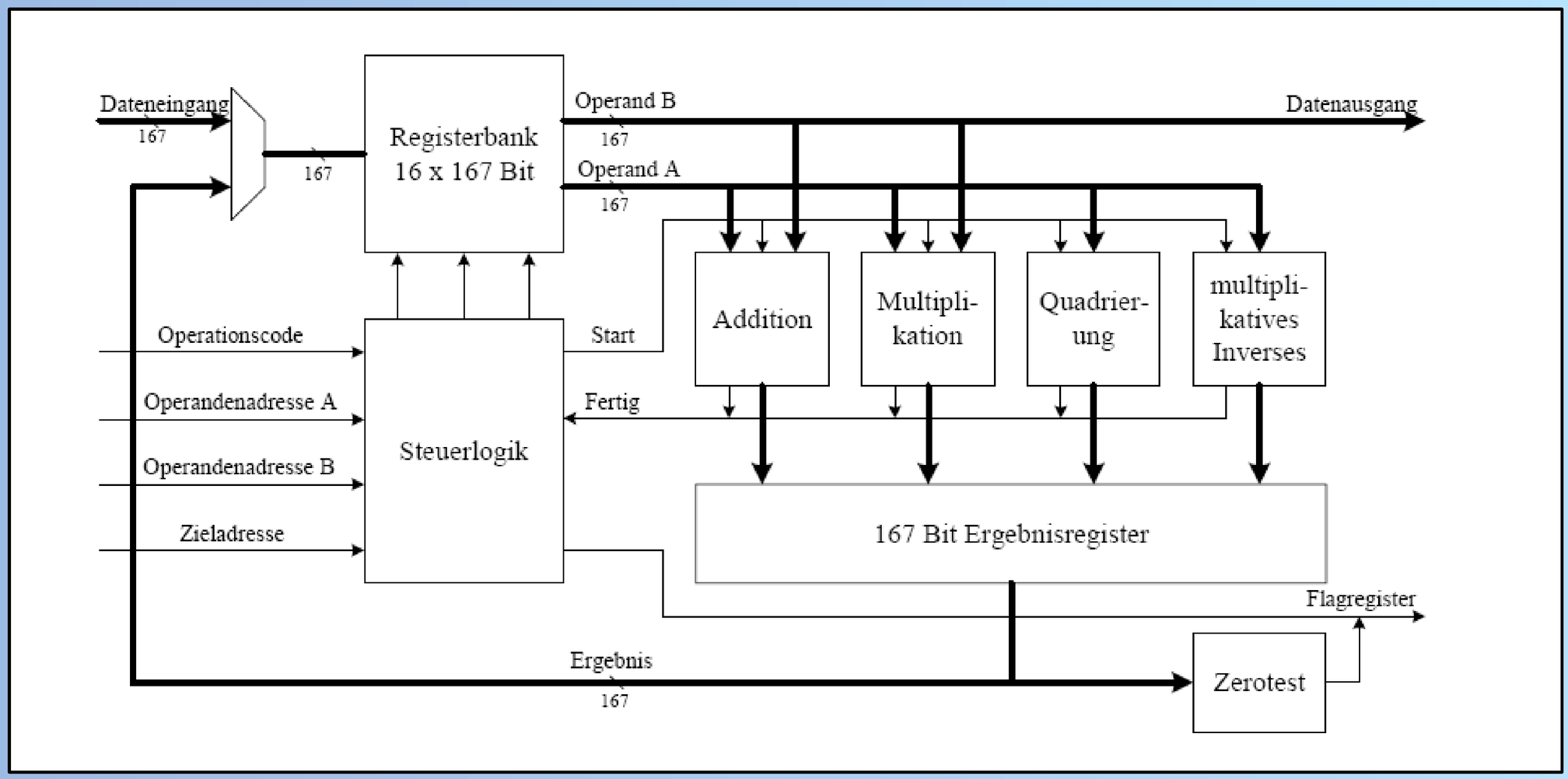


Arithmetische Einheit ALU



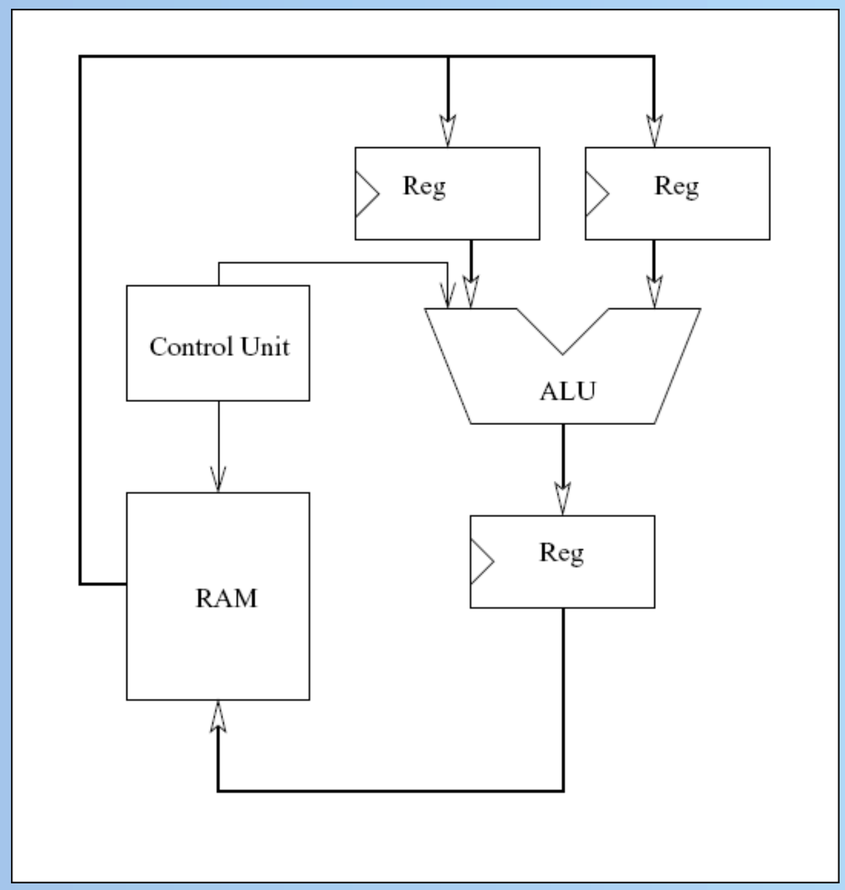
$$a + a = 2a \bmod 2 = 0$$

Arithmetische Einheit ALU



$$x = A * B = \sum A * b_i * 2^i \text{ mit } B = (b_{166}, ..., b_0)_2$$

Elliptic Curve Processor



Die Effizienz

ALU:

- Addition: 1Takt, 167 parallel arbeitende XOR Gatter
- Multiplikation: 167 Takte, 1 Zwischenregister, 1 Schieberegister von 168 Bit
- Quadrierung: 86 Takte, 1 Zwischenregister mit 333 Bit
- Invertierung: 167 Takte

ROM: Platzbedarf durch kombinatorische Schaltung < 0,1%

RAM: Platzbedarf durch FlipFlops > 50%

Größenbetrachtung bei 0,35 µm CMOS

ALU: 0,45 mm² **RAM:** 0,66 mm² **CU:** 0,2 mm² **Gatter:** ≈ 23000

max. Taktfrequenz bei 0,35 µm CMOS: 68,5 Mhz = 150 Kurvenoperationen/s

Reduzierung der Taktrate (bezogen auf ALU):
bei 80 µA pro 1 Mhz und 3,3 V: 17 mW Leistungsaufnahme
bei 80 µA pro 1 Mhz und 1,5 V: 4 mW Leistungsaufnahme

Sicherheit von Hardware

Zukunft der Hardwareverschlüsselung?

Quellen

Gerhard M. Glaser: "Verschlüsselung". 21.4.2003, http://www.tcp-ip-info.de/security/verschluesselung.htm

Wikipedia: "Kryptografie". 09.05.2007, http://de.wikipedia.org/wiki/Kryptografie

Unbekannt: "MEP: Codes and Ciphers. Unit 19 Lorenz Cipher Machine". http://www.cimt.plymouth.ac.uk/resources/codes/codes_u19_text.pdf

Frank Carter: "Colossus and the Breaking of the Lorenz Cipher". http://www.bletchleypark.org.uk/content/lorenzcipher.pdf

Unbekannt: "Die Geschiche der Kryptographie bis heute". http://www.iks-ruesselsheim.de/fileadmin/PDF/2._Geschichte.pdf

Martin Feldhofer, Kerstin Lemke, Elisabeth Oswald, Francois-Xavier Standaert, Thomas Wollinger, Johannes Wolkerstorfer: "State of the Art in Hardware Architectures". 05.09.2005, Revision 1.0, http://www.ecrypt.eu.org/documents/D.VAM.2-1.0.pdf

Arnaud Lagger: "Implementation of DES Algorithm Using FPGA Technology". 2002, http://www.alagger.com/des-vhdl/report.pdf

Lejla Batina, Guerric Meurice de Dormale, Elisabeth Oswald, Johannes Wolkerstorfer: "State of the Art in Hardware Implementations of Cryptographic Algorithms". 08.03.2006, Revision 1.0, http://www.ecrypt.eu.org/documents/D.VAM.10-1.0.pdf

cryptovision (Hrsg.): "ECC – Kryptographie auf Basis elliptischer Kurven (Eine kurze Einführung)". http://www.cryptovision.de/fileadmin/technologie/WP_ECC_a.pdf

Mathias Schmalisch, Marc-Sebastian Fiedler, Dirk Timmermann: "Eine ALU für die schnelle Berechnung der Kryptographie auf Basis elliptischer Kurven". 2003, http://www-md.e-technik.uni-rostock.de/veroeff/2003_iuk_paper.pdf

http://de.wikipedia.org/wiki/Bild:Egypt_Hieroglyphe2.jpg

http://www.kzu.ch/fach/as/aktuell/2000/03_schalttag/gezer.jpg

http://upload.wikimedia.org/wikipedia/de/2/2b/Alphabetum_Kaldeorum.jpg

http://upload.wikimedia.org/wikipedia/commons/e/e0/Babington_postscript.jpg

http://upload.wikimedia.org/wikipedia/commons/6/68/Kerkhoffs.jpg

http://upload.wikimedia.org/wikipedia/commons/5/59/BP-T52.jpg

http://upload.wikimedia.org/wikipedia/commons/5/50/Enigma_Verkehrshaus_Luzern.jpg

http://upload.wikimedia.org/wikipedia/commons/4/4d/Lorenz-SZ42-2.jpg

http://www.codesandciphers.org.uk/lorenz/pictures/lordiag.gif

http://upload.wikimedia.org/wikipedia/commons/b/b2/Skytala%26EmptyStrip-Shaded.png