[9] M. Ferrara, "Smart Experience in Fashion Design: A speculative analysis of smart material systems applications," *Arts Basel 8*, 2019, 1-11.
[10] A. Adamatzky, A. Nikolaidou, A. Gandia, A. Chiolerio, and M. M. Dehshibi," Reactive fungal wearable," arXiv preprint arXiv:2009.05670 (2020).
[11] X. T Tao, *Wearable Electronics and Photonics*, Cambridge, Woodhead Publishing Limited and Textile Institute Abington, 2005.
[12] M. Ferrara, M. Bengisu, *Materials that Change Color*, *op. cit.*
[13] S. Needhidasan, M. Samuel, R. Chidambaram, "Electronic waste – an emerging threat to the environment of urban India," *Journal of Environmental Health Science and Engineering 12(1)*, 2014, 36.
[14] A. Adamatzky & T. Schubert, "Slime Extralligence: Developing a Wearable Sensorial and Computing Network with Physarum polycephalum," UWE Research Repository 927012, 2013.
[15] F. V. Appels, S. Camere, M. Montalti, E. Karana, K. M. Jansen, J. Dijksterhuis, P. Krijgsheld, H. A. Wosten, "Fabrication factors influencing mechanical, moisture-and water-related properties of mycelium-based composites," *Materials & Design 161*, 2019, 64-71; M. Jones, A. Gandia, S. John, A. Bismarck, "Leather-like material biofabrication using fungi," *Nature Sustainability*, 2020. https://doi.org/10.1038/s41893-020-00606-1
[16] Y.-S. Bahn, C. Xue, A. Idnurm, J. C. Rutherford, J. Heitman, M. E. Cardenas, "Sensing the environment: lessons from fungi," *Nature Reviews Microbiology 5*, 2007, 57.
[17] I. M. Van Aarle, P. A. Olsson, B. Soderstrom, "Arbuscular mycorrhizal fungi respond to the substrate ph of their extraradical mycelium by altered growth and root colonization," *New Phytologist 155*, 2002, 173-182.
[18] C. Kung, "A possible unifying principle for mechanosensation," *Nature 436(7051)*, 2005, 647.
[19] M. Fomina, K. Ritz, G. M. Gadd, "Negative fungal chemotropism to toxic metals," *FEMS Microbiology Letters 193*, 2000, 207-211.
[20] Y.-S. Bahn, F. A. Muhlschlegel, "Co2 sensing in fungi and beyond," *Current opinion in microbiology 9*, 2006, 572-578.
[21] K. T. Howitz, D. A. Sinclair, "Xenohormesis: sensing the chemical cues of other species," *Cell 133*, 2008, 387-391.
[22] S. Olsson, B. Hansson, "Action potential-like activity found in fungal mycelia is sensitive to stimulation," *Naturwissenschaften 82*, 1995, 30-31; A. Adamatzky, "On spiking behaviour of oyster fungi pleurotus djamor," *Scientific reports 8*, 2018, 1-7; A. Adamatzky, A. Gandia, A. Chiolerio, "Fungal sensing skin," arXiv preprint arXiv:2008.09814 (2020).
[23] A. E. Beasley, A. L. Powell, A. Adamatzky, "Fungal photosensors," arXiv preprint arXiv:2003.07825 (2020).
[24] A. Adamatzky, A. Nikolaidou, A. Gandia, A. Chiolerio, and M. M. Dehshibi, "Reactive fungal wearable," *op. cit.*
[25] *Ibidem*.

Brain Lego
Toy Computing with Lego Bricks

Stefan Höltgen[1]

"HIRNLEGOHIRNLEGOHIRNLEGOLEGOLEGO
HIRNLEGOLEGOLEHIRNLEGOLEGOLEGO
HIRNLEGOLEGOLEGOHIRNLEGOLEGOLEGO"
(*Einstürzende Neubauten*, Hirnlego, 1989)

"I have always had a predominantly visual approach to my environment. This is also probably why I never pursued music. This perhaps one-sided talent was also evident in the construction of my computer models; here, too, I preferred mechanical and electromechanical constructions and left the electronics to others who were better qualified."[2]
In this quote, computer pioneer Konrad Zuse describes his tinkering with construction kits he played with as a boy and a teenager from his viewpoint as an engineer. He used those kits in the 1920s to build all sorts of things with them: (award winning) models of cranes and excavators, spare parts for his bike, and mechanical household aids. Later on, when his computers were already working electronically, he used the thought pattern for a new system of self-reproducing machines.[3]

This mechanical thinking of computer functions has a long tradition reaching back into the Middle Ages: from Ramon Llullu's book *Ars Magna* published in 1305 where a theological 'converter' for Muslims to become Christians is drafted, to Leibniz' *Machina arithmeticae dyadicae* from 1679 (a mechanism to calculate with binary numbers) — both remained "paper machines" — through to the mechanical and electromechanical logical machines of the 19th and 20th centuries,[4] culminating in Claude Shannon's switching algebra from 1937. All of these drafts were based on the idea to make calculation and computation not only logically but also mechanically constructible.
From our present-day perspective, some of these drafts appear more like toys than serious calculators; toys that merely show the principles of computation but are not very suitable for actual usage. This view also has to do with the fact that those prototypes show their material and epistemological toy characteristics: they are often built from construction kits (for children and youth) or from everyday objects — according to the "Baukastenprinzip" ("kit principle"),[5] using heuristic design procedures, trial and error, and learning by doing.
The invitation to *think while tinkering* ("thinkering")[6] seems to be a basic principle of both logic and kit toying because both make it possible to comprehend/handle complex phenomena. This logic (the two-valued sentential logic, inaugurated in the 3rd century B.C. from Aristotle) is the timeless and non-spatial basis of all our thinking. It provides the transcendental basic structure of truth-apt propositions which are the foundation of our everyday thinking, actions, science, playing, … This reality is formalized in logics: propositions become tokens that can hold a truth value (true/false — no third option) and can be combined with junctors (and, or, if-then, not, …) to complex sentences.
A sentence like "*Tonight I'll go to the movies or I will read a book*" can be formalized as *p*

v q where *p* means *"I'll go to the cinema"*, *q* means *"I'll read a book"*, and the *v* stands for the logical OR (not *either-or* since I can read my book in the cinema as well). Since each of these sentences can be true (t) or false (f), their or-combination can also be true or false. Ludwig Wittgenstein, following the system of Chrysippos of Soloi (279-206 B.C.), inaugurated a notation table to show the possible iterations:[7]

p	v	q
t	t	t
t	t	f
f	t	t
f	f	f

The fact that topics of thoughts and deeds can be formalized and written down in this manner fueled the speculation that logic does not always have to be transcendental but can be transferred from the realm of the symbols into the real. The mention of the logical machines in the beginning proves this to be true: propositions can become variables and variables can be transformed into versatile rods within a logical-mechanical design.
With the help of these machines logic, and only logic, can be automatized. To transform logical machines into computers that can calculate anything calculable, a third transformation must happen: propositions must become variables again and truth values must become numbers. This transformation was accomplished by George Boole in the late 19th century when he invented an algebra on the basis of Leibniz' system of binary numbers. Here, the truth values of t became 1, f became 0, and the logical junctors became arithmetic operators: and as *, or as + and not as -. Only three operators were sufficient to build all of the 16 possible junctors as the logician Charles Sanders Peirce proved in 1880.
To implement those logical functions into machines, merely a process of mental reinterpretation had to happen where a real state is read as a mathematical cipher. The real basis could be different electrical currents, pressure

Fig. 1. Braun Lectron (top center), Dr. NIM (bottom left), "Denken mit Lego" (bottom right).

differences of air or liquid, the presence or absence of a sound (or two different frequencies) — or the locomotion of mechanical parts. To say *"computers do understand only ones and zeros"* means: all of their computation is based on binary switching logic on whatever substrate. This primitive foundation is capable of an enormous complexity which can be determined in modern computers. They are only using more, faster, and smaller logical gates but work on the same regime.
The incrementation of complexity that leads to all the emergent effects of modern computing is nearly incomprehensible let alone can

be reenacted with macroscopic materials. To get an understanding of the logic behind those computing machines, it would have to be made visible again — by magnification, spatialization, and deceleration. This is where a fundamental review of computer history starts that can be considered as didactics: the handiness of toys that led to the construction of the first computers nearly one century ago can be used to get an understanding of today's ancestor. That is mainly because their fundamental principles are the same. Even the didactic is not new.

As early as the 1950s, toys[8] were built that made logical computer functions comprehensible/tangible (Fig. 1). The advent of digital computers in children's rooms transformed the didactics into a hands-on comprehension. This was realized by the use of kit systems that allowed implementing logical values and junctors in many different ways: electronic kits[9] (with newly designed "digital expansions", fig. 1, middle) and building bricks that facilitate the construction of mechanism — like Fischertechnik[10] or Lego.
"Denken mit Lego" (Thinking with Lego) paved the way for these 'brick didactics': a lego kit with a book companion where two mathematicians suggested *"funny games with logics and set theory"*[11] to *"provide experiences of the cohesion between propositional logics and set theory to children."*[12] Lego bricks were particularly suitable for this and could also be used to explain *"different bases of number systems"*[13] — especially of the binary system. By playing so-called "goal games" (91) logical junctors should be 'passed through' mentally: *"Any passway walked through provides a meaningful experience to the pupils"*[14] — the authors note before they show a diagram of the paths for *p v q* resp. *p ^ q*.
Using the lego brick to actually convert such 'thinking paths' into operational mechanisms only came to the minds of Lego players in the course of the last decade. Initially, they had to try the 'misuse of toy devices' and convert all sorts of things[15] into computers — just to see if it was possible. Computers themselves had provoked such hacking — unfinished gadgets that were just waiting to be transformed into usable tools by *programming* them. The hackers had approached them down to their hardware substrates. So, why not shift away this last symbolic historic and spatial border and expose their basic functions to the hacker creativity? Just to explore what computers are made of …
This is why so many Lego projects can be found today that introduce computers as calculating machines,[16] as theoretical constructions (Turing Machines),[17] and as logical networks — for replication with bricks. Logical AND, OR, and NOT gates can be easily assembled

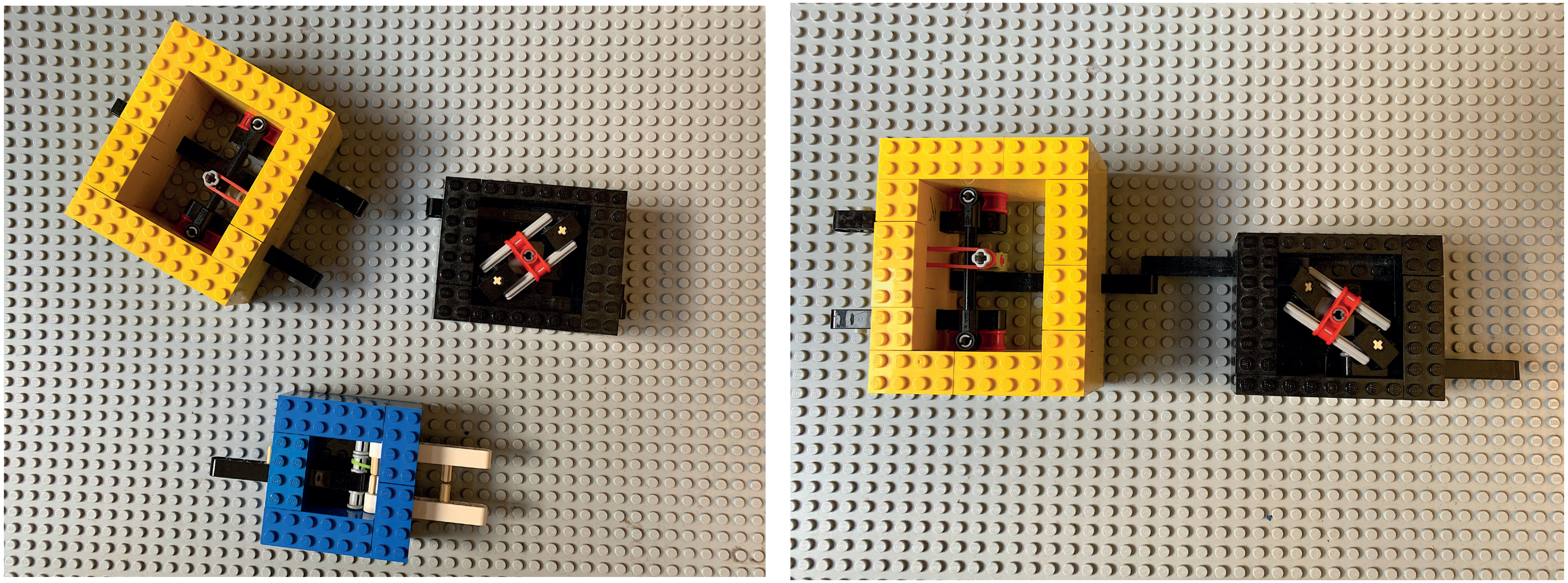

Fig. 2: Lego Lever Logic: AND, OR, NOT Gates (left), NAND gate (combined from AND and NOT) (right).

with only a few Lego bricks. (Only adding some rubber bands to help the gates return to their default state.) Instructions for "Lego Lever Logic"[18] gates can be recommended because they are easy to combine with each other to larger structures.

The gate's switching states can be recognized by the length of their levers: a long lever at the input stands for 0, a short lever stands for 1 — vice versa at the gate's output. Connecting the levers of a gate's output to another gate's input enables it to transmit a signal from the first to the second gate. Using some crafting skills, a stable construction of Lego gates that is anything but "*crude and* [...] *unworkable*"[19] can be built that provides complex working 'digital-mechanic' arithmetic circuits — like an adder that only utilizes AND, OR, and NOT gates. Such constructions are mostly build with the trial-and-error method: re-thinking and re-plugging bricks again and again — which can be seen as an re-enactment of Konrad Zuse's mechanical working Z1 computer architecture.[20]

Such projects can show how the abstract ideas of digital switching circuits become comprehensible hands-on. There are no boundaries for the creator's creativity and the circuit's complexity. The mechanical and material point of view of the formal principles of computing can help to not only comprehend the fundamentals of actual computer technology but also offers new ways of building computer technology by understanding the metaphor of "modular systems"[21] literally: "*Logic gates can be built in many ways. In mechanical computers they can be constructed by gear systems. In molecular computers they can be represented chemically.*"[22] These and even more 'alien' concepts are discussed within the realm of unconventional computing. Toy computing with Lego and other construction kits lines up into this experimental computer science — and enables an epistemological view on its historical and material sources.[23]

[1] Humboldt-Universität zu Berlin, Institut für Musikwissenschaft und Medienwissenschaft, Bereich Medienwissenschaft, Georgenstraße 47, D-10117 Berlin.

[2] K. Zuse, *The Computer – My Life*, Berlin/Heidelberg, Springer, 1993, 9.

[3] N. Eibisch, *Selbstreproduzierende Maschinen. Konrad Zuses Montagestraße SRS 72 und ihr Kontext*, Wiesbaden, Springer/Vieweg, 2016, 21.

[4] Cf. M. Gardner, *Logic machines and diagrams*, New York/Toronto/London, McGraw-Hill, 1958.

[5] N. Eibisch, *Selbstreproduzierende Maschinen*, *op. cit.*

[6] E. Huhtamo, "Thinkering with Media. On the Art of Paul deMarinis," in I. Beirer, S. Himmelsbach, C. Seiffarth (eds), *Paul DeMarinis: Buried in Noise*, Heidelberg, Kehrer, 2010, 33-46.

[7] L. Wittgenstein, *Tractatus logico-philosophicus*, trans. D. F. Pears & B. F. McGuinness, London/New York, Routledge, 2002, 38.

[8] http://txt3.de/brainlego1

[9] http://txt3.de/brainlego2

[10] http://txt3.de/brainlego3

[11] H. FREUND, P. SORGER, *Denken mit LEGO. Vergnügliche Denkspiele für Logik und Mengenlehre*, Freiburg/Basel/Wien, Herder. (Quotes translation by S. Höltgen.)

[12] *Ibidem*.

[13] *Ibid*.

[14] *Ibid*.

[15] "Dominoes even make for a fun gate implementation (albeit one that can only be run once)." J. Seiffertt, *Digital Logic for Computing*, Cham, Springer, 2017, 76.

[16] http://txt3.de/brainlego4

[17] http://txt3.de/brainlego5

[18] http://txt3.de/brainlego6

[19] J. Seiffertt, *Digital Logic for Computing*, *op. cit.*

[20] Cf. K. Zuse, *The Computer – My Life*, Berlin/Heidelberg, Springer, 2007, 35.

[21] H. G. Frank, B. S. Meder, *Einführung in die kybernetische Pädagogik*, München, dtv, 1971, 102-107; N. Eibisch, *Selbstreproduzierende Maschinen*, *op. cit.*, 36-45.

[22] J. Seiffertt, *Digital Logic for Computing*, *op. cit.*

[23] http://txt3.de/brainlego7