

P / REFERENCES OF DESIGN

CODING IN ART AND DESIGN: FROM DEMOCRATIZATION TO MAINSTREAM.

Kálmán Tarr*

Moholy-Nagy University of Art and Design, Budapest, Hungary

* tarr.kalman@teach.mome.hu

DOI: 10.63442/GPMR9128

KEYWORDS | VISUAL PROGRAMMING, ARDUINO, DISABILITY

ABSTRACT | In the „Stage of Impossibilities” project, I created a puppet moving mechanism for children with multiple disabilities. In this structure, the marionette is not manually controlled but can be moved by a robotic arm through tracking head movements. This technology has been used in a special needs school in Budapest as part of movement therapy and art classes.

A working prototype has been created in a few days using Arduino and visual programming languages. In my artistic research, I am exploring my role, primarily how a person with a background in arts education can take on a creative role in areas traditionally associated with engineering, particularly the creation of prototypes that require programming skills. The project led me to explore the origins of the tools I use, the ambition behind their creation, and their validity in today’s world.

A key element of my research is the study of development environments that have been created to democratize programming and make it accessible to artists and designers. John Maeda’s “Design by Numbers” project and its continuation with Processing and Arduino significantly contributed to making programming-based art an integral part of the cultural discourse, and many development environments - used in interactive and visual artworks - derive from them. These are certain types of development environments that I call the “build-your-own-tool” software. They approach the relationship between humans and machines from a different perspective. The „build-your-own-tool” software allows artists to create complex digital works without having deep programming skills.

It was the pioneers of computer art who gave directions for making this software, while they couldn’t rely on anything similar and had to write the algorithms themselves. The „build-your-own-tool” software gives the illusion that we create every part of the creative process ourselves, but at the same time, we are bound by the characteristics of the software itself. With these development environments, it is possible to make rapid progress, and they provide insight into the operation of the computer, but only a mediated and thus superficial knowledge.

The recognition of the boundaries recalls the Manifesto of Computer Art written by the Hungarian computer artist Tamás Waliczky in 1989, in which he stated, “Programmers develop software for us, the users, to use. Unless we know the capacities of the computer, we cannot sufficiently formulate our needs. Unless we sufficiently formulate our needs, we will receive nothing but traditional replies, programs that mimic the usual logic.” How relevant is the manifesto in today’s era when computers have become intricately more sophisticated? In my opinion, we have drifted from this statement just as much as computers have become more complex. In this context, even greater significance can be attributed to the Arduino and microcontrollers when creating programming-based art or design, as their complexity is comparable to that of early computers, and understanding them is much more approachable.

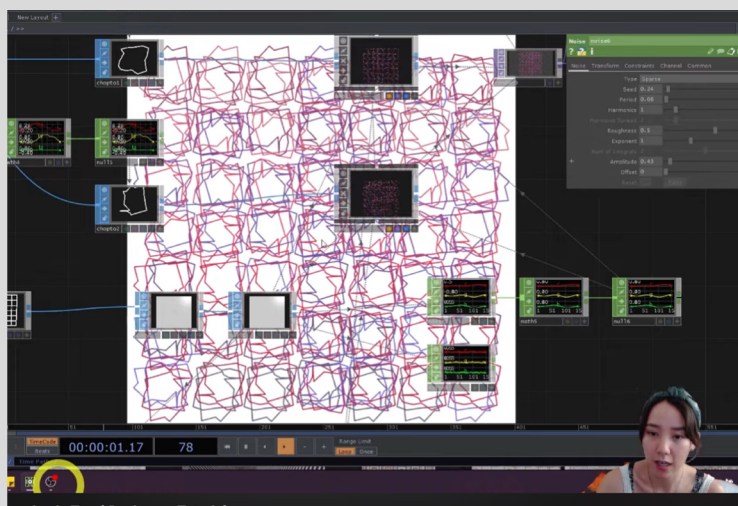
Focus of research

inspired by own project "Stage of Impossibilities"

DEVELOPMENT ENVIRONMENTS FOR DESIGNERS AND ARTISTS

- "Build your own tool"
- Origins and Historical Context
- Ambitions behind their creations
- Promises
- Limitations
- Creator's way of use
- Future, sustainability

Research methodology:
practice-based
interviews with artists and designers



The Stage of Impossibilities

ARTISTIC RESEARCH

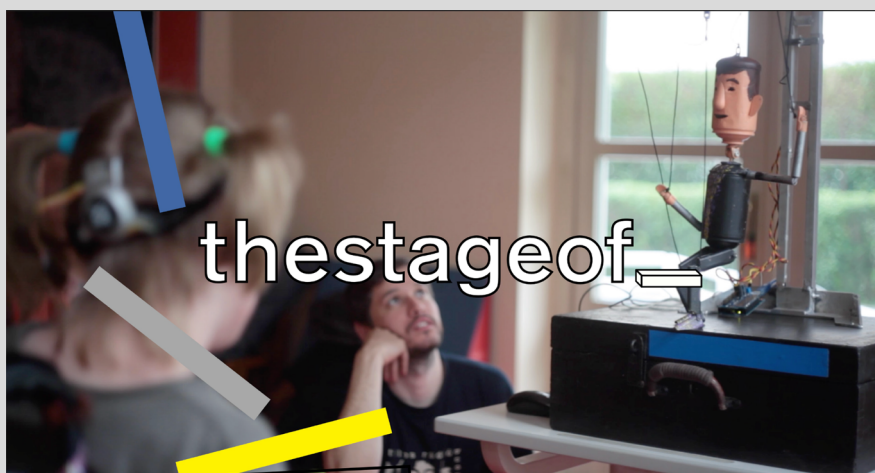
puppet moving mechanism
for children with disabilities

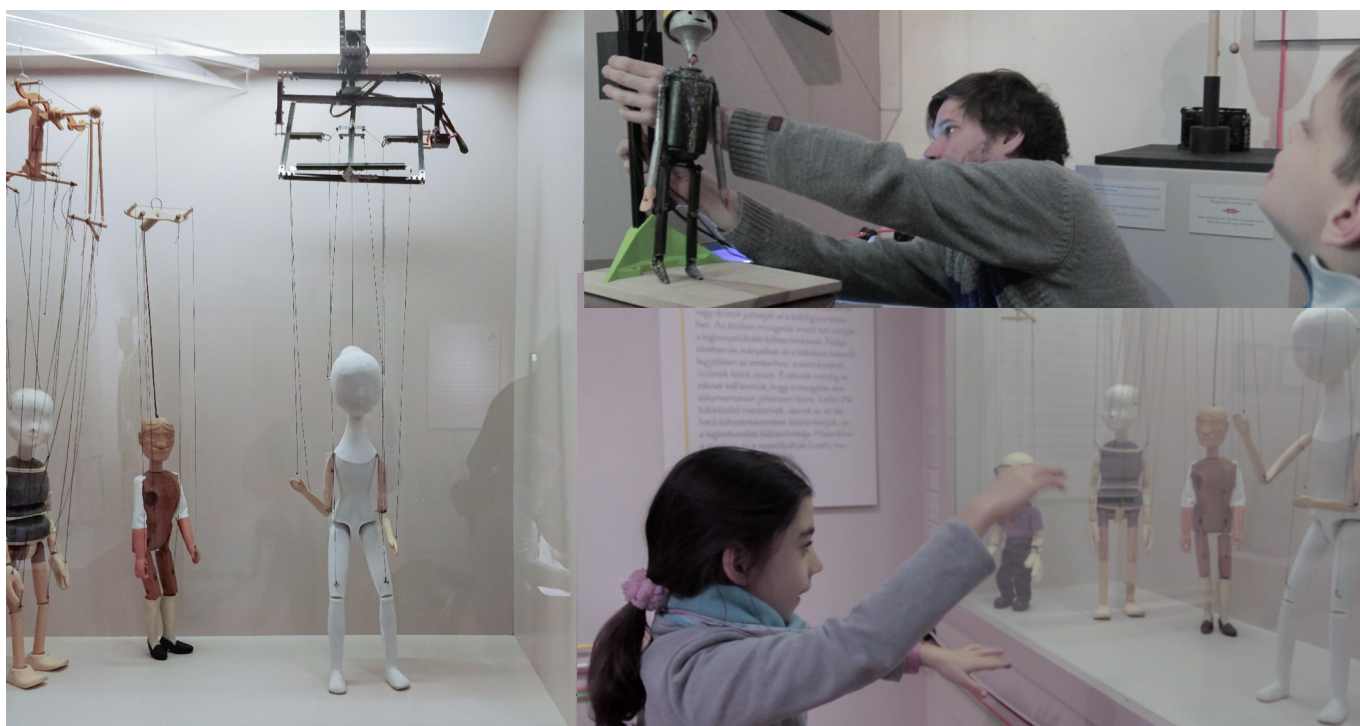
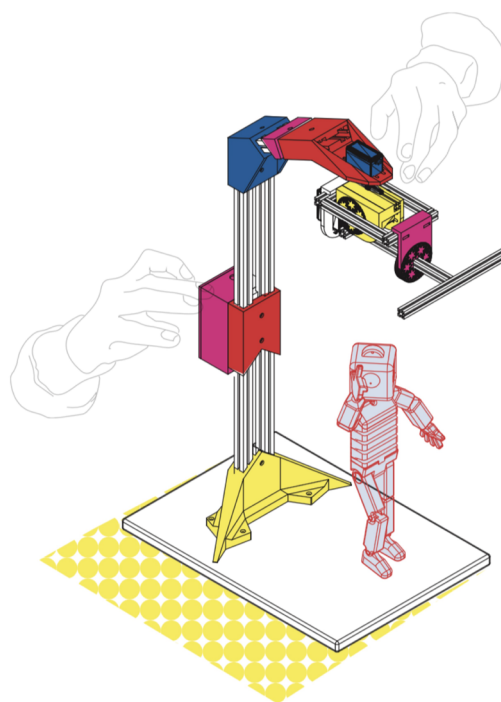
development in Csillagház
Elementary School in
Budapest

technical details

interdisciplinary network

examining own role in the
project





Democratisation of programming in art and design

Text-based IDE

DESIGN BY NUMBERS

- John Maeda, 1999

PROCESSING

- Ben Fry and Casey Reas, 2001

WIRING

- Hernando Barragán, 2003

ARDUINO

- M. Banzi, D. Cuartielles, T. Igoe, G. Martino, and D. Mellis, 2005

Visual programming language

AUDIO FOCUSED

MAX/MSP (1985), PURE DATA (1996)

- Miller Puckette

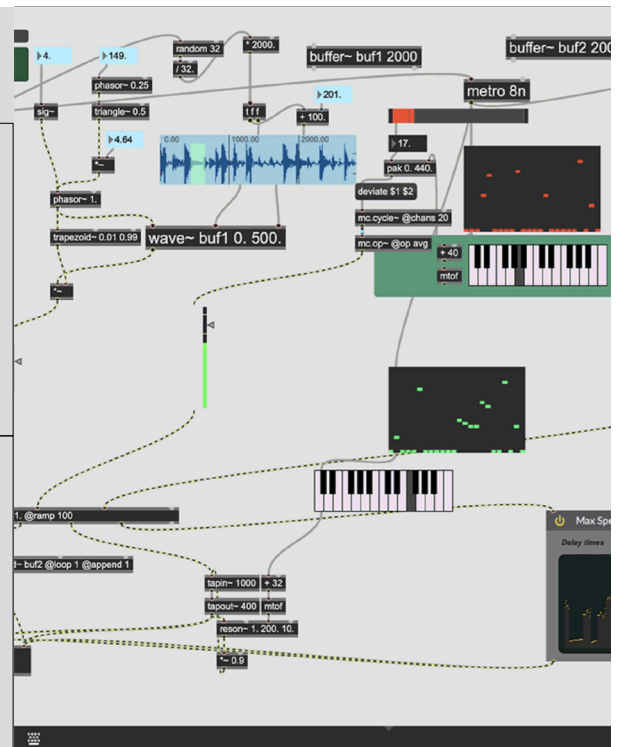
VISUAL FOCUSED

VVVV

- Mezo, 1998

Touch Designer

- Derivative, 2000



Historical Context - Aspirations in Art and Code

- MILLER PUCKETTE

- JOHN MAEDA

- TAMÁS WALICZKY

The Manifesto of Computer Art, 1989

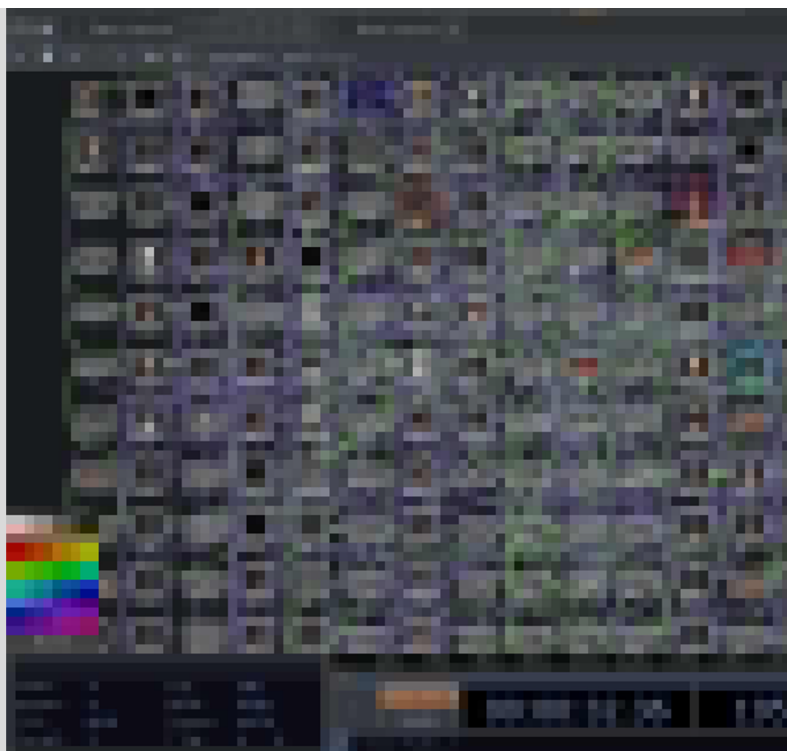
Let's establish a much better relationship with programmers. Because they become our co-authors even when we buy programs ready-made, and our work will bear their mark as well.

Programmers develop software for us, the users, to use. Unless we know the capacities of the computer, we cannot sufficiently formulate our needs. Unless we sufficiently formulate our needs, we will receive nothing but traditional replies, programs that mimic the usual logic.



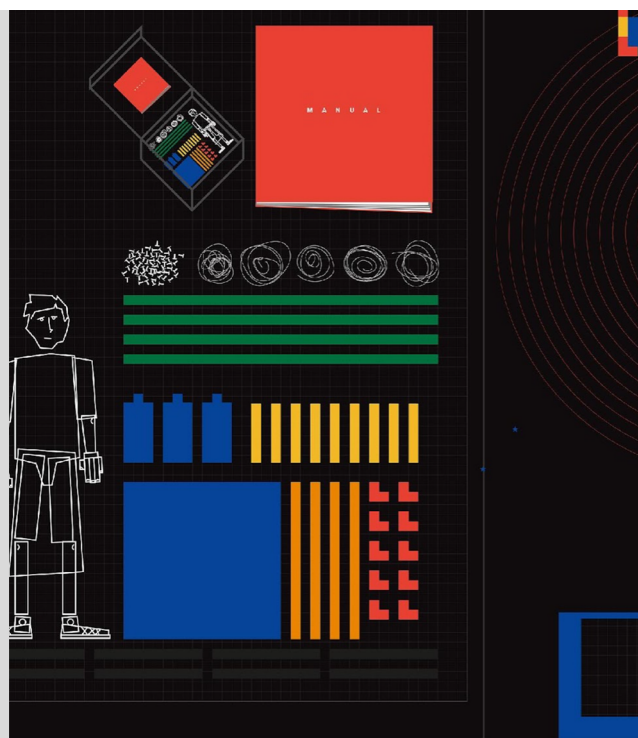
Limitations

- footprint
- prototyping versus product
- good for all, master of nothing
- cross-platform problems
- visual abstraction of written code
- licencing
- promise of easiness



Conclusion

- From democratisation to mainstream
- New directions: market segment of software
- Straightforward solutions - less 'build your own tool'
- Insight into the operation of the computer
- Effective tools in education and fast prototyping
- Software is leaving its mark on creations
- Preservation difficulties



Bibliography

Casey Reas, C., & Fry, B. (2010). *Getting started with Processing*. O'Reilly Media Inc.,

Maeda, J. (2001). *Design by Numbers*. The MIT Press.

Manovich, L. (2013). *Software Takes Command*. Bloomsbury Academic.

Manovich, L. (2002). *The Language of New Media*. The MIT Press.

Fuller, M.(Ed.). (2008). *Software Studies*. The MIT Press.

<https://puredata.info/>

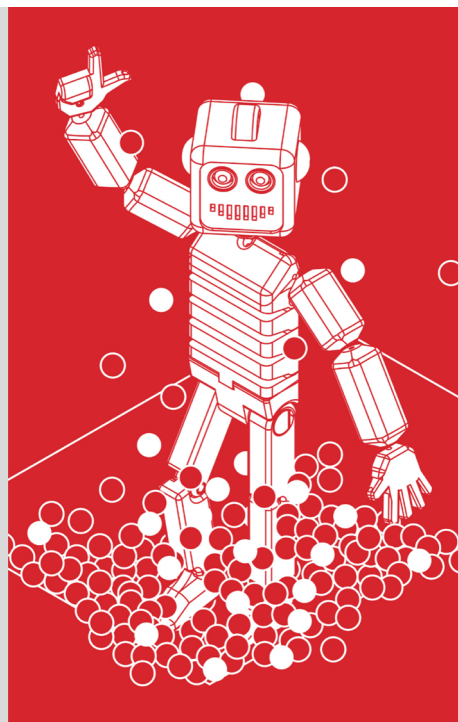
<https://cycling74.com/>

<https://derivative.ca/>

<https://visualprogramming.net/>

<https://wiring.org.co/>

<https://www.arduino.cc/>



P/REFERENCES OF DESIGN

This contribution was presented at Cumulus Budapest 2024: P/References of Design conference, hosted by the Moholy-Nagy University of Art and Design Budapest, Hungary between May 15-17, 2024.

Conference Website

cumulusbudapest2024.mome.hu

Conference Tracks

Centres and Peripheries
Converging Bodies of Knowledge
Redefining Data Boundaries
Bridging Design and Economics
Speculative Perspectives
The Power of Immersion
The Future of Well-being
Taming Entropy: Systems Design for Climate and Change
Ways of Living Together
Cumulus PhD Network

Full Conference Proceedings

<https://cumulusbudapest2024.mome.hu/proceedings>

ISBN Volume 1: 978-952-7549-02-5 (PDF)

ISBN Volume 2: 978-952-7549-03-2 (PDF)

DOI Volume 1: <https://doi.org/10.63442/IZUP8898>

DOI Volume 2: <https://doi.org/10.63442/TADX4016>

Conference Organisers

Moholy-Nagy University of Art and Design Budapest (MOME)

mome.hu

Cumulus Association

cumulusassociation.org