

A Mangle of Code
and the digitalising of the Home Sweet Home sampler

Paul Cooper

This thesis is submitted as partial fulfilment of the requirements for the degree of

MASTER OF FINE ARTS

Sydney College of the Arts
Faculty of Arts and Social Sciences
The University of Sydney

2025

This volume is presented as a record of the work undertaken for the degree of Master of Fine Arts at Sydney College of the Arts, University of Sydney.

*This is to certify that to the best of my knowledge; the content of this thesis is my own work.
This thesis has not been submitted for any degree or other purposes.*

I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

CONTENTS

<i>ACKNOWLEDGEMENTS</i>	i
<i>LIST OF IMAGES</i>	ii
<i>ABSTRACT</i>	iv
INTRODUCTION	1
CHAPTER 1 Art Technouveau	5
1.1 A Mangle, Rhizomes, and Actor Networks	5
1.1.1 Pickering Mangle	6
1.1.2 An Aleatoric Dance with Agency	7
1.2 The Technology of Art	9
1.2.1 Colonel Bogey	10
1.2.2 ASCII	11
1.2.3 Courtship and the human-machine relationship	14
1.3 Sonification	15
CHAPTER 2 The Code of Performance	19
2.1 Performative Code	19
2.1.1 Live Coding	19
2.1.2 After half a decade of the human-machine relationship	21
2.2 When the Machine Takes Control	23
2.2.1 Echolalia and Who's Trending Who	23
2.2.2 Silicon Chips and Wires	24
CHAPTER 3 DO NOT FOLD, SPINDLE OR MUTILATE	27
L'Art Informel and Aesthetic	27
3.1.1 Music for an Eagle and Three Satellites	31
3.1.2 I am a humanist and so is my computer	32
3.2 Code Poetry	37
3.2.1 Aesthetic Coding	39
CONCLUSION	44
<i>BIBLIOGRAPHY</i>	46
<i>MULTIMEDIA LINKS</i>	52
<i>APPENDIX</i>	
A.1 A brief description of the computer languages and hardware used in the making of my artworks	53
A.2 <i>I am a Humanist and so is my computer</i> (full code)	59
A.3 List of Works Presented for Examination	66

ACKNOWLEDGEMENTS

Being selected as a post graduate candidate to undertake this degree in Master of Fine Arts was just the beginning of a journey that I could not have completed without the help, support and encouragement of a cast of thousands, but in particular, Sydney College of the Arts, The University of Sydney, the Australian Government Research Training Program funding that assisted with fees, my supervisors Joyce Hinterding, John Tonkin and Rebecca Beardmore, and my family, Mandy Cooper, Marlie Fox and Brianna Cooper.

LIST OF IMAGES

- Figure 1 CSIRAC as currently displayed at the University of Melbourne.
Photo: John O'Neill
- Figure 2 ASCII characters for tonal graphics created by Alan Head.
Source: A.K Head and the Australian Journal of Physics (CSIRO)
- Figure 3 ASCII characters for tonal graphics and sewing by Paul Cooper.
Photo: Paul Cooper
- Figure 4 Joh Bjelke-Petersen, an early pseudo-greyscale ASCII example
Source: Synthesis by Steven Jones
- Figure 5 *Self Portrait in ASCII*, Paul Cooper 2024
- Figure 6 Philippa Cullen and Peter Dickson working with the wire antennae during
Homage to Theremin 2
Photo: Lillian Kristal
- Figure 7 *Extra Extra*, Paul Cooper 2023 (Installation view)
Photo: Paul Cooper
- Figure 8 DJ_Dave live on stage
Photo still from video: Clyde Munroe
- Figure 9 DJ_Dave's live coding example
Source: Pop Live Coding Encounters: Reflections on Practice
- Figure 10 *CodeKlavier meets Mercury*, Anne Veinberg 2021
Source: Video still Mercury Coding Session
- Figure 11 *Silicon Chips and Wires*, Paul Cooper 2024
Image: Peter Blamey
- Figure 12 *Silicon Chips and Wires* Punch Cards
Image: Paul Cooper
- Figure 13 IBM computer punch card c1965 (rectified image)
- Figure 14 *TV Garden*, Nam June Paik, 1974 (2000 version)
Source: www.guggenheim.org
- Figure 15 *I am a humanist and so is my computer*, Paul Cooper 2023-2024
Blue Screen of Death screen capture
Image: Paul Cooper
- Figure 16 *I am a humanist and so is my computer*, Paul Cooper 2023-2024
selfactualisation.py computer code
Image: Paul Cooper
- Figure 17 *Code Violation* Holden and Kerr, 2016
code_violation.pde computer code poem
Source: www.code-poetry.com
- Figure 18 *Code Violation* Holden and Kerr, 2016
Screen capture: Paul Cooper

Figure 19 *Water* Holden and Kerr, 2016
water.c computer code poem
Source: www.code-poetry.com

Figure 20 *Extra Extra* Paul Cooper 2023
Dry-point print UP
Image: Paul Cooper

Figure 21 *XXXX XX* Paul Cooper 2023
Image: Paul Cooper

ABSTRACT

A Mangle of Code is a studio-led research project that examines interactive and reactive experimental art practices encompassing electronics and creatively written computer code. Exploring the liminal spaces between art, technology and science, this paper's research considers human and non-human interactions, and the role computer systems play in their relationship. Referring to artists who incorporate interaction, data sonification and experimental coding, including Philippa Cullen, Jasmine Guffond, Eryk Salvaggio, Daniel Holden, Chris Kerr, and Anne Veinberg establish the argument for experimental computer programming as being an art material. Varying and various agencies of both the human and non-human players within creative processes are discussed using Andrew Pickering's *Mangle of Practice* as the base-line theory. Accompanying this paper is a body of artworks that take a multimodal approach consisting of audio-visuales, tactile handcrafts and spontaneity. Each artwork has a connection to computer programming either as a functioning program or as an abstractly interpreted aesthetic resulting in a mangled exhibition of experimental code.

INTRODUCTION

Code and software form an integral part of almost any work with computers regardless of whether we are aware of it or not. In my art practice, code has reached the status of a medium, a medium for art that sits among the others, providing a platform for experimenting, testing ideas and mechanisms. Like concrete poetry and the use of text in art, the written word begins to operate in other ways, melding form and function to enable a conceptual art practice that transverses conventional art platforms, finding a space in the metaverse and the gallery.

The ambiguity of the term *mangle* becomes a touchstone within this research as both clarifying agent and disruptor for a body of artworks that move between the hidden and the exposed while intertwining contemporary digital spaces with historic analogue residues. Connectivity and interactivity are injected with passivity and uncertainty inside technologically dependent artworks that form the basis for this theoretical and art historical inquiry into the frameworks, networks, processes and principles underpinning dynamic artworks that engage audiences and technologies.

The historical shift from analog impulses to on-off instructions of the digital is embraced in works that explore the possibilities for evidencing the recursive and self-referential tropes within coding and computing. In contrast, other works reverse these connections so that the digital becomes an agent for invoking historical spaces and references, such as computer punch cards^{1 2} and hand sewing or enacting a physical or performative event.

The inclusion of the word *mangle* stamps the ambiguity and the connectivity of experimentation to the title and throughout the paper. Mangle has a number of meanings and classes, and is used to describe order, disorder, certain machines and connection.³ Examples

¹ My art practice uses computer punch cards in the context of interplay between the digital and tactile worlds. The invention of punch cards by Joseph Marie Jacquard, patented in 1804, as a textile loom controller although is the for runner to the medium I use, it is outside the scope of this paper. Same too the pianola and its punch card-like scrolls invented by the Aeolian Company in 1895 do not form part of this research.

² James Essinger. *Jacquard's Web: How a Hand-Loom Led to the Birth of the Information Age*. 1 ed. Oxford: Oxford University Press, Incorporated, 2007., 35.

³ Macquarie Dictionary (Online). North Ryde, N.S.W: Macquarie Library/Pan Macmillan Australia, 2003-2024. <https://www.macquariedictionary.com.au/>. Definition of the word 'mangle'. Accessed 28-Aug-2024

of its use include “*their sentence structure is somewhat mangled*”, as in jumbled; “*can you please pass me the pasta mangle*”, referring to the kitchen wringer that combines, flattens and stretches pasta dough; and from Spanish conversation, “*Fui a caminar por los mangle pantano*” which translates to “*I went for a walk through the mangrove swamp.*” It is this correlation of the Spanish word *mangle* with *mangrove*,⁴ the trees of the *Rhizophora* species among others that grow in the marsh lands of estuaries and whose roots spread out both above and below the water line to help form ecosystems that allow salt and fresh water fauna and flora to cohabitate thus establishing mangle as a form of connection, melding, and interaction.

British philosopher and historian of science Andrew Pickering describes his approach to observation and engagement of constructed exploratory scientific experimentation as *A Mangle of Practice* due to his processes being multifaceted, and accepting of agency, unpredictability and uncertainty.⁵ Pickering’s *Mangle of Practice* is a principal that argues passive stages of involvement are critical. Passive engagement, as Pickering explains, entails observation rather than constant interaction which leads to a repetition of measured responses whether the resultant outcomes were predicted or unexpected⁶. It is this type of approach that, when used in the writing phase, places experimental coding within the realm of science, technology and, notably, art.

The use of computer programs is commonplace in the making of digital prints, sound files, digital displays and live music. The list of terms that label these art practices is extensive: algorithmic art, digital art, electronic art, multi-media, and generative art are just some, however this paper will not be drawn to define any of them, instead preferring to use the holistic term, art, and then reflect on its conceptual reasoning. The specific style and purpose of the computer programming approaches I have adopted is referred to as *experimental coding*.

⁴ Collins Dictionary (Online). Harper Collins Publishers. <https://www.collinsdictionary.com/dictionary/spanish-english/mangle> 2024. Spanish *mangle* translates to the botanical English word *mangrove*. Accessed 23-Aug-2024

⁵ Andrew Pickering. *The Mangle of Practice Time, Agency, and Science*. Chicago: University of Chicago Press, doi:10.7208/9780226668253. 1995

⁶ Andrew Pickering. “The Robustness of Science and the Dance of Agency.” In *Characterizing the Robustness of Science*, 292:317–27. The Netherlands: Springer Netherlands, 2012. https://doi.org/10.1007/978-94-007-2759-5_13, 2-3

The research is presented as an exhibited display of physical works I created in response to the findings as well as in the form of this paper of three chapters. Each chapter has subsections of theory, history, artists and an overview of selected artworks including those emanating from my studio made during this candidature.

The first chapter, *Technouveau Art* reviews the beginnings of technology in art practices and the resonance artists establish between themselves, and their materials. The nature of this resonance is explored through the concept of the rhizome as described by Deleuze, Guattari and others. Pioneers in computer research at The University of Sydney and the writings of Stephen Jones are significant elements within the historical investigation. The University of Sydney was an early adopter of computers, having the fourth digital computer in the world on its campus, CSIRAC,⁷ and shortly afterwards another computer, SILLIAC,⁸ which facilitated early programming exploration and experimentation. The pushing of computer boundaries due to the need for better graphical representations led to ASCII⁹ Art¹⁰ which shifted computers from being elaborate calculators to machines that have artistic potential. In the late 1960's, Philippa Cullen, enrolled in an arts degree at the University of Sydney¹¹ and began to collaborate with students from the Engineering and Computer Science Faculties to discover ways computers can be utilised within an artistic framework. This is where this paper takes solace. My need to know how things work, how things can fail and more importantly how things can be made to be unpredictable is only magnified by my background in electrical engineering as well as my interest in producing experimental computer code as part of my art practice.

The adaptability and creativity of coding is highlighted in chapter two, *The Code of Performance*. This chapter looks at computer code's capability as a real-time interaction between humans and machines. As a progression from Cullen's collaborative works, the development of interacting with machines is demonstrated by the live-coding of computer-based digital music by contemporary artists DJ_Dave and Anne Veinberg. The notion of autonomous machine agency is investigated in my artworks *Echolalia and Who's Trending*

⁷ Its name was the anagram of Commonwealth Scientific and Industrial Research Automatic Computer and was put into use in November 1949.

⁸ Its name was the anagram of Sydney version of the Illinois Automatic Computer

⁹ ASCII characters are simply the characters that a computer keyboard can generate on a text-based screen.

¹⁰ The University of Sydney was involved in the compiling of a set of 11 ASCII characters that offered a reasonable tonal shift to print out simulated grey scale images onto computer paper. This concept went onto include intricate images of the Mona Lisa through to happy birthday banners for office parties and beyond.

¹¹ Evelyn Juers. *The Dancer: A Biography for Philippa Cullen*. Giramondo Publishing, 2021, 175-176

Who, 2023 and *Silicon Chips and Wires*, 2024 that use data streamed from the internet to question the influence algorithms have over connected users.

Do Not Fold, Spindle or Mutilate, the title for the third chapter takes its name from the cautionary instructions regarding computer punch cards¹² of the sixties. The warning was to make sure the cards were handled with care. This chapter ignores the warnings and sees systems being pushed to their breaking point; not physically, but conceptually. The findings see technology being taken to unconventional places including the digitising of embroidery, embroidering the digital and enabling machine languages to be concurrently poetic, aesthetic and functional. The research concludes with how the abstraction, and compilation of data into message centric multisensory experiences has become possible through the development of experimental computer coding within collaborative installations.

The artworks presented, as listed in Appendix A.3 represent a cross-section of findings and experimentation undertaken during this candidature. Some of the works include hand stitched abstracted computer code, a hand stitched computer-generated portrait reflecting the times before graphic printers, and projections with sound depicting executed computer code. All the code presented, and the hardware, including the sewn works is made, built or configured by the author.

¹² Computer punch cards were the forerunner to the computer keyboard and will be explained in detail in chapter 3.

CHAPTER 1

TECHNOUVEAU ART

Decades before the advent of the digital age, artists were already being inspired by new technologies and materials to produce experimental works of light, movement and sound which then continued and accelerated after World War II.¹³ Nowadays electronic devices are things that come in boxes and are expected to just work. This chapter, however, takes the viewpoint that technology is something that can be altered, modified and repurposed. Examples of artistic practices in the field of human-machine interactivity demonstrate this and how technology can exist as both material and subject of an artwork. To lay the foundations for understanding networks with imbedded systems of cause-and-effect, such as those that electronics can offer, the writings of Pickering, Deleuze and Guattari are explored. This is followed by a brief and focussed review of the history of technology and the pathways artists have engaged to include its use as part of their practices. Attention is drawn to artists including Philippa Cullen and Anne Veinberg who were instrumental in the development of electronic technology as a form of creative expression. In their live performances they interacted with music generated by sensors, feedback loops and computerised synthesizers. The chapter will conclude with a description of an artwork of mine that questions the level of agency that humans have in the digital world when internet-based algorithms are involved.

1.1 A Mangle, Rhizomes, and Actor Networks

“In The Mangle of Practice, I argued that processes take the form of a dance of agency between the human and the nonhuman. In their research, scientists seem to oscillate between bursts of what Ludwik Fleck called phases of activity and passivity. In the active phase, scientists are genuine agents, setting up their apparatus this way or that. In the passive phase, they stand back and see what happens.”¹⁴

– Andrew Pickering

Networks can comprise of any number of living and non-living things including people, fauna, flora, singular objects and mechanical equipment with each element able to collaborate

¹³ Kristine Stiles, and Peter Selz. *Theories and Documents of Contemporary Art : A Sourcebook of Artists' Writings*. Sourcebook of Artists' Writings. 2nd ed., rev. and expanded. ed. Berkeley [Calif: University of California Press, 2012., 450

¹⁴ Pickering. *The Robustness of Science and the Dance of Agency*, 2-3

and react in some way. These interactive networks are discussed from varying angles through different philosophical lenses, but all have commonality of connection and reaction. The Actor Network Theory (ANT), for example, which French philosopher Bruno Latour argues that the connected elements, which he calls *actors*, each make contributions within a network having equal value and agency in a non-hierarchical, horizontal plane of influence.¹⁵ In other words, the actions of one actor, no matter how small will still have a significant effect on the network. Gilles Deleuze and Félix Guattari consider these interactive networks as an assemblage which they refer to as having a *rhizomic* structure with their connections being either a direct joining of two or more elements or an in-direct relationship. This counters the top-down linear thinking styles of many social models and conveys the benefits of interconnected agency which leads to a cohesive environment.¹⁶

“Principles of connection and heterogeneity: any point of a rhizome can be connected to anything other and must be. This is very different from the tree or root, which plots a point and fixes an order.”¹⁷ – Deleuze and Guattari.

To help visualise Deleuze and Guattari’s horizontal connectivity concept, the word *rhizome* is a derivative from the genus of trees that form mangroves, *Rhizophora*.¹⁸ It is not just trees that make up mangrove swamps however, there is salt water, fresh water, tidal waters, crabs, birds, insects and so on. Mangroves are an ecology of interconnections that act as a filter, a barrier and a life support where much of the interactions happen away from plain sight, occurring below the surface . It is these types of ecological relationships that this research project draws on to explore the way humans respond to technology and how technology may respond to the user.

1.1.1 Pickering’s Mangle

British scientist and philosopher Andrew Pickering narrows the explanation of the rhizome analogy by using examples of the way scientific experiments are carried out. Pickering accepts the amount of, and value attributed to interaction, reaction and influence by each

¹⁵ Bruno Latour. "On Actor-Network Theory: A Few Clarifications." *Soziale Welt* 47, no. 4. 1996, 369-81.

¹⁶ Deleuze and Guattari. *A Thousand Plateaus*, 21

¹⁷ Deleuze and Guattari. *A Thousand Plateaus*, 7

¹⁸ Department of Agriculture, Fisheries and Forestry , Bureau of Rural Sciences – Australian Forest Profiles: Mangroves.

https://www.agriculture.gov.au/sites/default/files/abares/forestsaustralia/publishingimages/forest%20profiles%202008/mangroves_2008.pdf Archived from the original on 16 March 2011. Accessed 14-Apr-2024.

contributor can vary from being actively controlling through to being a passive participant, plus additional elements of uncertainty may exist by chance or be injected on purpose. He acknowledges that even with constructed environments, the level of complexity can be extreme and suggests that networks can be a mangle of ideas, objects and processes.¹⁹ Pickering uses the term *Mangle of Practice* to describe the observation and reaction to these constructed situations. Having a variety of definitions which leads to ambiguous use, he uses the word mangle to highlight the need for practices to be open to suggestion, foster intrigue and accept inherent multiplicities.²⁰

*“..for me, it conjures up the image of the unpredictable transformations worked upon whatever gets fed into the old-fashioned device of the same name used to squeeze the water out of the washing. It draws attention to the emergently intertwined delineation and reconfiguration of machinic captures and human intentions, practices, and so on.”*²¹ – Andrew Pickering

The connecting of mangle to mangroves and mangroves to rhizomes in both a physical sense and as dynamic processes offers a way to view agency as something that is not solely a human oriented domain. The internet, for example is a place of meeting, of storage and communication, and as such is a rhizomic mangled network with its connections, interconnections and entanglements. Click on a hyperlink and the world-wide-web becomes a rabbit hole. The internet is a medium for many art practices beyond its search and research aspects. The interfacing of live streaming information and data visualisation is just one way to create real-time audio-visual works of art as demonstrated later in this paper.

1.1.2 An Aleatoric Dance with Agency

Pickering’s mangle proposition claims that if we consider the scientific viewpoint of rhizomes, they can be described as a flexible collaboration between people and things, as well as things and things. Throughout this process the observer is undertaking various methods, observing the results, then repeating, and then comparing. The process of adjusting the materials and interactivity and consorting with uncertainty Pickering calls a *Dance of Agency*.²² Pickering clarifies his stance by citing Polish scientist and philosopher Ludwik Fleck’s phases of activity and passivity. Scientists, the constructors and observers of

¹⁹ Andrew Pickering. *The Robustness of Science and the Dance of Agency*. 317-27, 2012.

²⁰ Pickering, *The Mangle of Practice*, *New Ontologies*, 1

²¹ Pickering. “*The Robustness of Science and the Dance of Agency*.”, 23

²² Pickering. *The Mangle of Practice*, vii

situations dance between being genuine agents during the proposal and set up stages, then allowing the experiment to happen while they remain passive on lookers, making any necessary adjustments then allowing the research to proceed unhindered.

The mindset required by the observer/player and in this case a non-actor, is akin to the performative nature of the mapping analysis that Deleuze and Guattari use to describe the workings of a rhizome. A map, as they explain, *constructs the unconscious* and accepts the possibilities of multiple points of entry and multiple paths of journey.²³ The map, in this case, is no longer a two-dimensional reproduction or a list of instructions but rather a set of clues to possibilities. This contrasts with Latour's sociological approach where all are actors, and agency is an embedded constant. The process of writing software is likened to a science experiment. Preparation, formulation and syntax are all necessary but then the system runs, and the coder observes, watches for desired functions, error messages and malfunctions, adjusts and tries again. For this reason, Pickering's mangle approach is an appropriate point of view for this research. Similarly, works by artists Alan Kaprow, John Cage and Tino Sehgal construct performative art situations that establish variances resulting in something that is not easily (nor wanted to be) repeated exactly. The term aleatoric is typically used to describe this genre of artworks/performances.

*"a process is said to be aleatoric ... if its course is determined in general but depends on chance for detail."*²⁴ – Werner Meyer-Eppler

Understanding Werner Meyer-Eppler's statement about aleatoric works where parameters are set in a generalised way, but the outcome is randomly influenced, assists the observer when analysing artworks of this nature. With Deleuze and Guattari speaking of multiple entry points and Pickering's dance of variable agency, rhizomic-mangled environments offer differing levels of chance and uncertainty resulting in an aleatoric outcome.

A major take from Pickering's mangle processes is to be responsive and to having a willingness to accept feedback. Australian artist Ros Brandt recognised this in her essay *Designing with Chaos: Allowing the unprogrammable to occur*.

"Since 1977 I have consciously setup open artworks, large time-scaled electronic environments, sound sculptures, installations or playgrounds where the perceiver empowers the art form and shapes the many sonic possibilities that can coexist. While certain sonic parameters are predetermined, their existence through time,

²³Deleuze and Guattari. *A Thousand Plateaus*, 12

²⁴ Werner Meyer-Eppler. "Statistic and Psychologic Problems of Sound", translated by Alexander Goehr. Die Reihe 1 ("Electronic Music"). 1957., 55

their duration, repetition, and placement are affected by the perceiver's presence and movements."²⁵ – Ros Brandt

There are three ways that interactive networks are described here: Actor Network Theory by Latour which is an over-arching viewpoint that states all contributors, or *actors* within a system are actively engaged and ultimately have equal agency; Deleuze and Guattari's rhizome analogy of how productive networks benefit most when hierarchical influences are diminished and the interplay and interconnectedness of all players are acknowledged, and Pickering's constructed situations, such as scientific experiments, where importance is placed on a looping process of activity, passivity and considered contribution. It is Pickering's *Dance of Agency* within his *Mangle of Practice* that my research and studio experimentation has been guided by.

Throughout this project, I am using *mangle* as Pickering does, that is to approach the combining of dissimilar materials, objects and processes to invoke enigmatic aesthetics, chance, and intrigue. Experimental coding invites input from multiple actors via hardware sensors, chance and live data gathering to produce multisensory interactive installations which will be addressed in chapter three.

1.2 The Technology of Art

To help understand how the use of electro-mechanical technology in art came to be, the manifesto of Italian futurist, painter and composer Luigi Russolo is a starting point. In 1913, Russolo wrote in response to the noises of the newly established industrial environment that even the largest of orchestras produce no more than five different sound types: strings rubbed, strings pinched, wind on metal, wind on wood and percussion which caused him to explore bespoke sound generation. With the audio sensations of machinery clanging, grinding, hooting and whirling there was an opportunity to engage in the reproduction of these sounds and noises through the building of acoustic mechanical devices as art objects beyond the structured design of musical instrumentation.

²⁵ Ros Bandt, Ros. 1995. 'Designing with chaos', *Zeitgleich, Klanginstallation und Medienkomposition im digitalen Zeitalter*, Austria:Transit, <https://www.kunstradio.at/ZEITGLEICH/CATALOG/ENGLISH/bandt-e.html> 1995, 53-64, & 158-161

“The time is right for us to embrace the infinite possibilities of noise, to imagine and construct sound combinations of trolleys, autos and other vehicles, and that of loud crowds²⁶.” – Luigi Russolo

He goes on to say that by building custom devices, modifications can be made to manipulate and extend pitch, frequency and tone either through design or by way of playing.

It took nearly four decades after Russolo’s manifesto of 1913 for electrical devices to be included along with acoustic instruments. Initially, electro-technology was only realistically accessible to corporate and government entities due to its scarcity and expense. It was, for example, the 1930’s that the photocopier was invented,²⁷ and electric guitars were commercially released but they weren’t readily available for artistic use until the later part of the 1950s.²⁸ Computers in the fifties turned from being mechanical devices to being electrical and became the domain of educational institutions²⁹ which is also around the time where computers as an artistic tool began to take root.

1.2.1 Colonel Bogey – world’s first computer generated music

CSIRAC (Council for Scientific and Industrial Research Automatic Computer), Australia's first digital computer, and the fourth programmable computer in the world was commissioned in November 1949 at the University of Sydney, Camperdown (USYD).^{30,31} In early 1950, computer programmer and musician, Geoff Hill programmed CSIRAC to play the wartime tune Colonel Bogey March. Hill took it upon himself to setup this experiment, that is, his computer time access had not officially authorised him to see if a computer could generate music therefore no recordings were made of this historical moment, with the only evidence

²⁶ Luigi Russolo. *The Art of Noise : Futurist Manifesto*, 1913. Reprinted by Something Else Press. New York, <https://search.library.wisc.edu/catalog/999809462902121>. 1967.

²⁷ L.B. Schein. *Electrophotography and Development Physics*. Springer Series in Electrophysics ; Volume 14. Second edition. ed. Berlin: Springer-Verlag. doi:10.1007/978-3-642-77744-8. 1992, 3-7

²⁸ Paul Atkinson. *Amplified: A Design History of the Electric Guitar*. First edition. London: Reaktion Books, Limited, 2021, 33

²⁹ While 1900’s saw the introduction of vacuum tube computers they were one-off experimental designs. The military then began to oversee the development of computers with such machines as COLLOSSI, but these again were purpose built, not ‘programmable’ as such.

³⁰ "CSIRAC: Our First Computer." <https://cis.unimelb.edu.au/about/csirac>.

³¹ CSIRAC remains as the only intact first-generation digital computer in the world.

being journal notes, and selected witnesses.³²

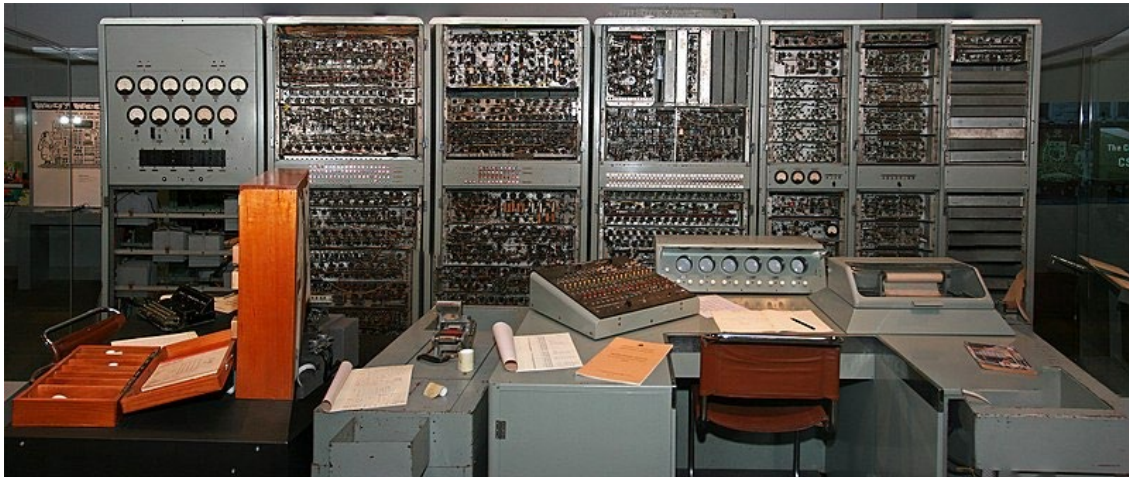


Figure 1: CSIRAC as currently displayed at the University of Melbourne. [Photo: John O’Neill]³³

1.2.2 ASCII

By 1956, USYD commissioned its second computer, this time named SILLIAC. Computers and computer programming were gaining wider appeal and uses. A uniformed set of 95 printable characters was established as the *American Standard Code for Information Interchange (ASCII)* in the early 1960’s and is still in use today. The set contains 52 letters (26 lower case and 26 uppercase), ten numbers and 33 symbols such as a full stop, a comma, ‘@’ and ‘#’. As computers became more versatile, printed documents with large volumes of data were becoming complexed, cumbersome and difficult to decipher.³⁴ Computers having no way to output graphics and being restricted to print just the 95 ASCII characters led to discovering inventive ways to print images. This was done by tabulating letters, numbers and punctuation marks into their perceived greyscale tonal values based on the ratio of each character’s ink to the negative space it sat on, and the first computer-generated images were produced.³⁵ Data in graphical form could now be printed.

³² Sandra Peter and Kai Riemer. Why the Unlearn Project and Computer. The Unlearn Project. University of Sydney, 22 January 2024, <https://sbi.sydney.edu.au/why-the-unlearn-project-and-computers/>. Accessed 15-Mar-2024

³³ Link to YouTube clip of Australian historian Paul Doornbusch’s recreation of the Colonel Bogey March as it may have sounded on CSIRAC in 1950. <https://www.youtube.com/watch?v=DuEjeQGeyZk>

³⁴ Stephen Jones. Synthetics : Aspects of Art and Technology in Australia, 1956-1975. Leonardo Books. Cambridge, Mass: MIT Press, 2011, 34

³⁵ The perceived tonal value is an optical illusion based on the ratio of a printed character’s foreground (black imprint) versus the white background of the paper. For example, a dash is a light tone, a zero mid-tone and a hash, a darker tone: - 0 #

Australian computer scientist, Alan Head, a materials scientist in the CSIRO Division of Tribophysics, devised a set of 11 ASCII characters, some being over printed to offer greater tonal variation and depth which was then used across the world.³⁶ The printing in perceived grey scales soon led to this process being used to print photographs and other graphical images. While Head was not an artist and did not wish to make art, he created the avenue for artists to explore the technology.³⁷ This style of art is referred to as ASCII art.

To understand first-hand the power of the tonal quality ASCII characters provided, I chose to experiment with creating ASCII art of my own. Writing a program to convert a digital image to my own set of ASCII characters was part one. My selection of characters was based on their perceived tonal qualities and my decision to connect hand-made coding with hand-made handicrafts by way of embroidery. The character set I chose was based on me being able to sew them. The results, a hand-sewn self-portrait using ASCII characters depicted below, (Fig 5) below and via a link to a short video.³⁸

³⁶ Stephen Jones. Synthetics: Aspects of the history of Electronic Art in Australia. <https://the-synthetic-image.com/home/data-visualisation-and-computer-graphics/>. Accessed 13-Ju-2023

³⁷ Jones, Synthetics, 34

³⁸ YouTube. *ASCII Portrait*, Paul Cooper, 2023. <https://youtube.com/shorts/h6lOcr98bwU>

Figure2: (TOP LEFT) The overprinted character greyscale adopted by Alan Head. [Source: A.K. Head and the Australian Journal of Physics (CSIRO)]

Figure 3: (TOP RIGHT) 'Greyscale' ASCII characters I selected for their ease to be hand sewn. [Photo: Paul Cooper]

Figure 4: (BOTTOM LEFT) *Joh Bjelke-Petersen, Queensland Premier, 1968* an early ASCII example as printed by the SGIO computer system at the time. [Source: Synthesis by Steven Jones. <https://the-synthetic-image.com>]

Figure 5: (BOTTOM RIGHT) Paul Cooper *Self-Portrait in ASCII, 2024*. Hand stitched cotton on canvas. 297x420mm [Photo: Paul Cooper]

1.2.3

The courtship of the human-machine relationship

The nineteen sixties saw art moving fast in various directions of discovery and experimentation as seen with the Fluxus movement. USYD having a strong continued history of research and experimentation in 1969 opened The Fine Art Workshop (aka Tin Sheds), a place where students from across the University could meet and collaborate on multi-disciplinary projects. Around this time, a group of architecture, medical and electrical engineering students formed a collective within Tin Sheds they called Optronic Kinetics.³⁹ They met up with undergraduate student and choreographer Philippa Cullen who was studying Fine Art and Medieval Dance. Working across disciplines, she realised that the role

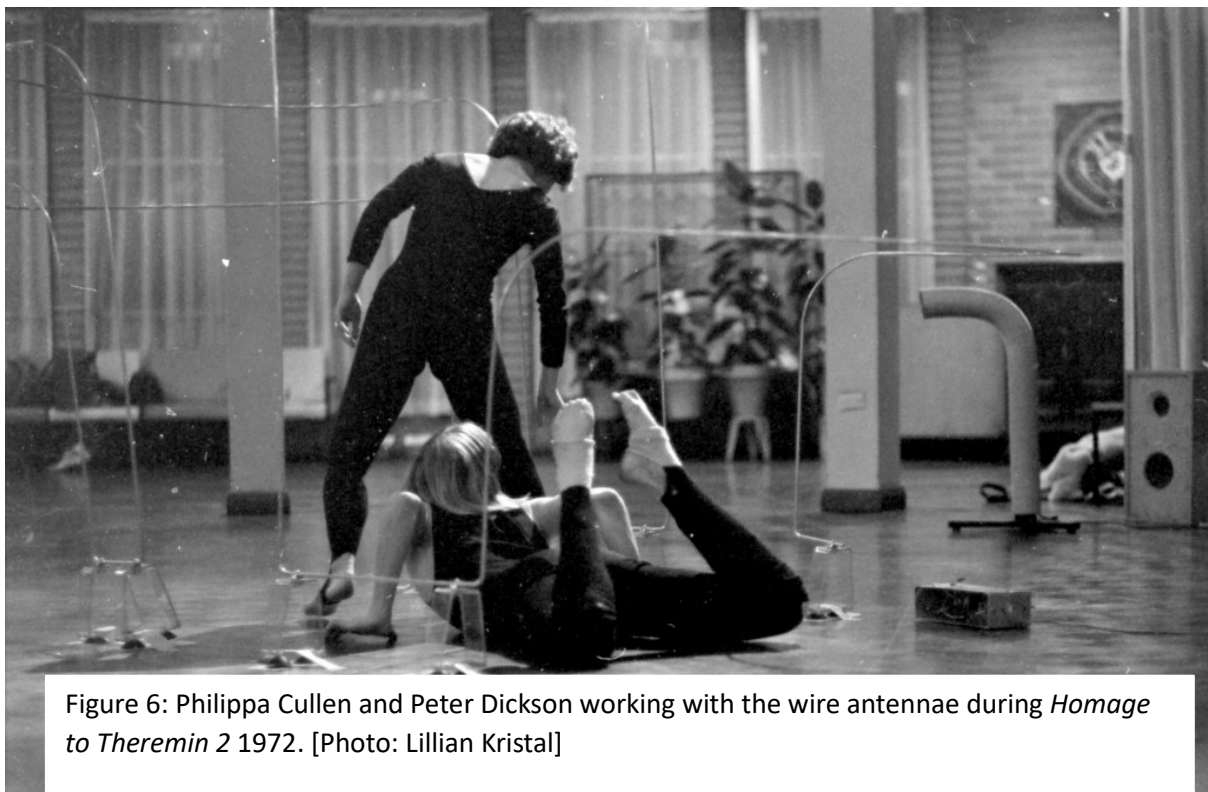


Figure 6: Philippa Cullen and Peter Dickson working with the wire antennae during *Homage to Theremin 2* 1972. [Photo: Lillian Kristal]

³⁹ Optronic Kinetics members included composer Greg Schiemer, electrical engineer Phil Connor and architecture student Manuel Nobleza

of the dancer did not have to be secondary to that of the accompanying music. This is an example of Pickering's *Mangle of Practice* with interactions and observations taking place. With electronics becoming more available and affordable, Cullen and Optronic Kinetics set about devising ways for a dancer to simultaneously be the musical composer and player – so that the dancer and music could share agency and become one. They did this by combining a Leon Theremin invention with computer science and art. Theremin's invention of 1939, simply called the Theremin, was an analog sound generator, played by moving your hands around small aerials. This movement generated variable audio frequencies and controlled the volume through changes to air-borne capacitance. The Cullen and Optronic Kinetics collective continued to collaborate and found new ways to modify Theremin aerials so they could be stretched across a theatre's stage. They went further and experimented with additional sensors and other inputs from another of Theremin's creations, the Terpsitone. The Terpsitone uses pressure sensors instead of aerials to control frequency and volume which Cullen and Optronic Kinetics incorporated into stage platforms to be activated by dancers' feet and bodies. Cullen was said however, to be unimpressed with the simple sinusoidal sounds of the Theremin, so she worked with the Optronic Kinetics team to interface their system with synthesizers. In addition to Leon Theremin's instruments, Cullen experimented with a system she described in an ABC Radio interview in 1975 as Bio Feedback. Using medical instruments and monitors that she had modified to converted heartbeats and breathing to sounds⁴⁰. Philippa died in 1975 while living in Auroville, India. She was 25 but has left a legacy of inspiration to those of us who come across her story, the story of the dancer who was a pioneer in controlling computers with her body.⁴¹

1.3 Sonification

Prior to this research project, my electronic sculptures and installations primarily responded to interactions emanating from the audience and/or the immediate surroundings via local sensors. A significant part of this research has been to broaden the possibilities so that interaction could extend beyond the gallery. While the broadcasting of sound and images by radio and television had been possible for decades, it was with the coming of the internet and

⁴⁰ Juers, *The Dancer*, 480

⁴¹ Jones, *Synthetics*, 307

the transmission of data through various narrow casting channels that the opportunities for remote access to digital information flows became possible.

Audio psychologists Michael Nees and Bruce Walker propose a theory of sonification that includes four main auditory functions. The first category is for audible warnings which signal events requiring urgent attention, like fire alarms and hospital patient monitors. Canadian environmentalist, composer, and writer R. Murray Schafer would call these *sound signals*.⁴² The second category is for audible notifications, used in situations where a prompt response is needed, such as a microwave timer bell, call waiting beeps on a phone, and motor vehicle reversing sensors. The third category is linear and/or representational sonification, that transforms entire data sets, unlike the previous moment-in-time examples, into mathematically based sounds, and considers aspects like volume, waveshape, and frequency. Lastly, Nees and Walker describe non-linear soundscapes as audio experiences where data sets inspire and indirectly influence the resultant sounds⁴³. An example is *Listening to the Mind Listening* by audio engineer Dr. Stephen Barrass of SONIFICATION.COM, which converts human EEG data into synthesized orchestral music.^{44 45 46}

Sonification, as Jasmine Guffond's 2021 University of New South Wales PhD thesis demonstrates, can be part sound information and part instrumentation as seen in her artwork *Listening Back 2021*, which takes the form of a web browser plugin. It generates sound in real-time when cookies are saved and deleted on your computer, which happens when you open web sites, view content and click on hyperlinks. The artwork highlights the background tracing activity that goes on while we are using our computers online.⁴⁷ Ness and Walker's theory of sonification would suggest that this work is a linear soundscape, that is, with more cookie activity comes more sounds. While on the surface the audio experience can be entertaining, the message is one of awareness of our online vulnerability through the abundance of generated sounds, such is the power of sonification. Discovering real-time on-

⁴² R. Murray Schafer. *The Soundscape : Our Sonic Environment and the Tuning of the World*. Rochester, Vermont: Destiny Books, 1994, 292

⁴³ Nees and Walker, *Theory of Sonification*, 5-6

⁴⁴ Stephen Barrass, *Listening to the Mind Listening @ Sydney Opera House Studio*, *Sonification.Com* <https://sonification.com/2004/07/08/listening-to-the-mind-listening/> Accessed 26-Feb-2025

⁴⁵ Stephen Barrass, *Listening to the Mind Listening 10th Meeting of the International Conference on Auditory Display* <http://www.icad.org/websiteV2.0/Conferences/ICAD2004/concert.htm> Accessed 26-Feb-2025

⁴⁶ Stephen Barrass. "The Aesthetic Turn in Sonification Towards a Social and Cultural Medium." *AI & society* 27 (05/01 2012): 177-81. <https://doi.org/10.1007/s00146-011-0335-5>.

⁴⁷ Jasmine Guffond. "Listening Back." Doctor of Philosophy, UNSW, <https://trove.nla.gov.au/work/249074170>. 2021

line data sonification like *Listening Back 2021* was a turning point in my research as it showed me a way to be expressive when writing computer code.

Guffond's sonification example opened a new avenue for research which led me to experimenting with time-based internet sourced information. I needed to upskill my code writing so that I could download, collate, interpret and turn internet data into sound in real-time. The automated collection of information from websites to be stored on a local computer is called *web scraping*.⁴⁸ Together with upskilling my abilities in writing computer code I considered artistic uses of this collected information, and how to turn facts and figures into sound so that our auditory senses can attune to different types of information.⁴⁹

An early experiment of mine *Extra Extra, 2023* (fig. 7), an internet enabled single board computer with web scraping software sought to reflect on information bombardment and the subliminal effect it can have. A stochastic sound work consisting of a structured set of musical rules that transformed the text of live streaming news headlines into sound. The gallery space was filled with sound and projections of news headlines. The algorithms took on both compositional and improvisational roles, thus exhibiting multiples levels of agency. By projecting and news headlines together with the accompanying soundscape, the audience was placed in the middle of the room confronted by the type, volume (quantity) and changes to leading stories on all sides leaving them no choice but to be consumed and become part of the installation. Spontaneous synchronicity is investigated further in this paper's accompanying exhibition through *Resonance in a Glitch Field 2025*, an installation of over twenty electronic displays which despite have no interconnecting control wiring drift appear to self-synchronise then drift apart then re-synchronise and so on, refer Appendix A.3.4 The next chapter will develop this study into the effect that web-based algorithms have on the agency and synchronicity of internet users through the examples of my work, *Echolalia and Who's Trending Who 2023*, and Jasmine Guffond's *Listening Back 2021*. The chapter also investigates artists who create, interact and perform with software rather than simply use it in its prescribed manner.

⁴⁸ Macquarie dictionary (Online). [definition of 'web scraping'] Accessed 28-Aug-2024

⁴⁹ Michael Nees and Bruce Walke. Theory of Sonification. Principles of Sonification: An Introduction to Auditory Display and Sonification., Georgia Institute of Technology, Atlanta, GA, USA: 9-39. 2012, 5

CHAPTER 2

THE CODE OF PERFORMANCE

2.1 Performative Code

Synthesised sounds, noise and music has advanced with the development of software, taking live performances in a new direction. Performative coding has added a new dimension to instrument building through what is known as *live coding*.

2.1.1 Live Coding

Live coding is the writing of computer programs on stage in a dynamic compositional manner, where the artist becomes both composer and conductor, allowing the audience to witness the creation of the sounds in real-time. A kind of human-machine jam session. One such popularised system is SonicPi, developed in collaboration with Sam Aaron of the University of Cambridge and the microcomputing organisation, Raspberry Pi Foundation.^{51,52} Sound artist Sarah Davis (stage name DJ_Dave) is a live coder who uses SonicPi. The images below (Fig 8 and Fig 9) and video link provided in the footnotes depict the human-machine interface of the computer terminal screen being projected on stage while the artist codes and performs. Normally, software is developed and trialled before being released, and artists generally use off the shelf software packages.⁵³ By incorporating real-time programming, live coders offer a unique connection with their audience by inviting them to witness both the development and performative aspects of their art.

⁵¹ Alan Blackwell; Alex McLean; James Noble; Julian Rohrerhuber. "DROPS - Collaboration and learning through live coding (Dagstuhl Seminar 13382)". Dagstuhl Reports. 3 (9): 130–168. doi:10.4230/DagRep.3.9.130. 2014. Retrieved 2 May 2015.

⁵² Rory Cellan-Jones, Rory. "Baked in Britain, the millionth Raspberry Pi". BBC News. 7 October 2013. Retrieved 2 May 2015.

⁵³ There are several popular software suites that cover studio production and editing with Ableton Live being a capable and preferred live performance virtual instrument offering an intuitive graphical user interface (GUI).

```

live_loop :DJ_DAVE, sync: :met1 do
  f = 0.05
  16.times do
    sample sss, "texture_18", rate: 1, finish: f if pattern "x--x--x--x--x-x-"
    sleep 0.25
  end
  16.times do
    sample sss, "texture_18", rate: 1.5, finish: f if pattern "x--x--x--x--x-x-"
    sleep 0.25
  end
  32.times do
    sample sss, "texture_18", rate: 1.2, finish: f if pattern "x--x--x--x--x-x-"
    sleep 0.25
  end
end
end

```

Figure 8: DJ_Dave's live coding, 2023 [Source: Pop Live Coding Encounters: Reflections on Practice]⁵⁴



Figure 9: DJ_Dave live on stage. [Photo still from video: Clyde Munroe]⁵⁵

⁵⁴ An example of DJ_Dave's live coding as presented in the essay: Sarah Davis, Jack Armitage, and Gus Lobban. "Pop Live Coding Encounters: Reflections on Practice." https://iil.is/pdf/2024_iclc_davis_et_al_pop.pdf

⁵⁵ Link to YouTube clip: https://www.youtube.com/watch?v=JiQHclg_648

2.1.2 After half a decade of the human-machine relationship

Australian musician, living and working in the Netherlands, Anne Veinberg, is a computer programmer with a twist. She and collaborator Felipe Ignacio Noriega have devised a system for live coding using a piano rather than a computer keyboard. Veinberg and Noriega's early development of their musical interface, CodeKlavier Lambda CKalculator⁵⁶ was a human-machine-human trio. Veinberg would play a piano that wrote code through sound together with Noriega live coding on a computer. The next phase in the development was to 'replace' Noriega with a Disklavier. A Disklavier is a computer-enabled version of a pianola with the ability to be an interactive⁵⁷ human-machine sound creator. An important element of Veinberg's coding-playing is sharing the piano with a computer that controls the keys with actuators that form a human-machine union. This can result however, in the player, as Veinberg explains, reaching for a particular key which the computer simultaneously actuates, thus removing control from the player in a tussle of agency.⁵⁸

Maria Kallionpää's *Climb! 2017* project follows similar lines to that of Veinberg's approach to the human-machine connection using a Disklavier. *Climb! 2017* is an experimental performance of live coding through a computerised gaming engine. The premise is that the performer is given a set of musical challenges that journey through levels like an arcade game such as *Donkey Kong* or *Mario Brothers* but with the graphics being replaced with code, a simple progress chart and text-based instructions. Anne Veinberg was invited to play *Climb! 2017* as a performer and to offer feedback on its playability. She noted that while the game had several presets that guide the player, there was an additional level of accessible interaction. Unlike the binary characteristic of a keyboard, with each key being either up or down, the Disklavier can write code based on intonation and speed in conjunction with the notes played and therefore how the machine responded and how the player progressed through the game. Veinberg described this as claiming her own musical space with the human-machine

⁵⁶ Anne Veinberg and Felipe Ignacio Noriega. Offzzing the Ccalculator <https://www.youtube.com/watch?v=hD-PWNDebD4>. Accessed 25-Jul-2024.

⁵⁷ Werner Goebel, and Roberto Bresin. "Are Computer-Controlled Pianos a Reliable Tool in Music Performance Research? Recording and Reproduction Precision of a Yamaha Disklavier Grand Piano." Paper presented at the Proceedings of the 2001 Workshop on Current Research Directions in Computer Music, 2001.

⁵⁸ Kanga Zubin, Anne Veinberg, Maria Kallionpää, Adrian Hazzard, Chris Greenhalgh, and Steve Benford. "Deus Ex Disklavier: Subjectivity and Technological Resistance in the Performance of Maria Kallionpää's *Climb!* For Disklavier and Electronics." In *Performance, Subjectivity, and Experimentation*, edited by Catherine Laws, 79-108: Leuven University Press, 2020, 90-91

interaction being very similar to a human-human duet, but with the added complication of playing the same instrument.⁵⁹

These types of human-machine performances have advanced to include levels of failure, chance and shared agency. No two performances will be the same and can be quite dissimilar even though it is the same work. These performances are a demonstration of art intertwining with musical endeavour by adding a mangle of interaction and chance.



Figure 10: Anne Veinberg *CodeKlavier meets Mercury*, 2021
[Image: Video still Mercury Coding Session]⁶⁰

⁵⁹ Zubin, Veinberg, Kallionpää, Hazzard, Greenhalgh and Benford. Subjectivity and Technological Resistance in the Performance of Maria Kallionpää's *Climb!* 93

⁶⁰ Recorded at Muziekhuis Utrecht NL, January 2021

Link to YouTube video: <https://www.youtube.com/watch?v=e4sPKOlAYS8&t=936s>

2.2 When the Machine Takes Control

Processes within my studio practice have developed throughout this candidature in part due to the embracing of Andrew Pickering's *Mangle of Practice* principles of observation and re-configuration. His example of acknowledging unexpected outcomes, no matter how well a situation or experiment is constructed is a lesson in being open to new ideas and differing points of view. Acknowledging the unexpected is the first step, adjusting the process based on results is the next. The system is then allowed to evolve in its own space through the inclusion of chance through random number generation, external inputs and other stochastic processes. The internet and its flow of fluctuating data is one such external input as it can provide unexpected and unintended results for an artwork to interact with, similar in a way to how internet users are influenced by web-based algorithmic assumptions.⁶¹

2.2.1 Echolalia and Who's Trending Who

Personal Information Agent, *Datafication* and *Digital Doppelganger* are terms used by on-line marketing and business analysis consulting firms to describe the collection, collation and distribution of internet users' activities. All internet users have a ghostly twin, a digital doppelganger composed of numerous sources such as browser cookies. These *digital crumbs*, as the consulting firms explain, make your interactions easier and more individually relevant.⁶²

Algorithms and the way data harvesting leads to relentless recommendations by service providers to generate advertising income, drew me towards internet trends. Trending seems to suggest the things the public are interested in based on their Google searches. There is, however, the additional element of crafted algorithms that place search results in a particular order that favours fee-based advertised topics and their links.

Tapping into Google search trends in real-time, I began an inquiry into human-machine online interactions which followed closely by considering how machine-to-machine interactions exist via multitasking algorithms. This resulted in an audio-visual internet

⁶¹ P. Glocviczki. *Journalism and Memorialization in the Age of Social Media*. 1 ed. New York: Palgrave Macmillan, 2015. doi:10.1057/9781137460875, 56

⁶² Interfacing Technologies Corporation, Canada. "Your Digital Doppelganger" <https://www.interfacing.com/digital-doppelganger>. accessed 3-Apr-2023

enabled artwork *Echolalia and Who's Trending Who*, 2023.⁶³ Utilising web scraping Python libraries I wrote customised code to run on micro-computer hardware with an internet connection, visual display monitor, and powered speakers to convert Google data to text and voice. The voice used is the standard Google voice providing a connection to the data source. The audio track is manipulated and played back at different speeds across multiple channels. Four speakers are arranged facing each other in a chatty huddle as if nattering about the latest news and gossip tit bits to one another. In sync with the voices, the work displays a rotating list of the twenty most popular words and phrases being searched (Googled) on the web in real-time. The artwork interfaces with the trending items of the country in which it is being exhibited to allow the audience to recognise at least some of the searches and question others. Trends can be biased by opinionated demographic groups with niche interests and divergent perspectives and possibly spam bots⁶⁴, hence the title. It should also be noted that these types of trends, either by individuals or algorithms can lead to a kind of journalistic sabotage resulting in random and overtly frequent news cycles.⁶⁵

2.2.2 Silicon Chips and Wires



Figure 11: Paul Cooper *Silicon Chips and Wires*, 2024
 [Installation/performance view, University of Sydney computer lab. Image: Peter Blamey] ⁶⁶

⁶³ Link to short video capture example of *Echolalia and Who's Trending Who* <https://youtu.be/pygDdfegCNk>
⁶⁴ Abhijnan Chakraborty, Johnnatan Messias, Fabricio Benevenuto, Saptarshi Ghosh, Niloy Ganguly, and Krishna Gummadi. 2017. "Who Makes Trends? Understanding Demographic Biases in Crowdsourced Recommendations". *Proceedings of the International AAAI Conference on Web and Social Media 11* (1):22-31. <https://doi.org/10.1609/icwsm.v11i1.14894>, 30
⁶⁵ Gloviczki. *Journalism and Memorialization in the Age of Social Media*, 57
⁶⁶ YouTube. *Silicon Chips and Wires*, 2024 by Paul Cooper. <https://youtu.be/8Z3TtwPPIlg>

With artificial intelligence (AI) gaining traction and making headlines throughout the writing of this paper it is impossible to completely ignore it. From an experimental point of view, I wanted to collaborate with AI to create an artwork. *Silicon Chips and Wires, 2024* is an artwork that comprises a poem that I asked AI to write which was then turned into a text and voice video loop. The video loop was then played on thirty computer screens and was accompanied by a live performance of me cross-stitching the poem onto computer punch cards. Computer punch cards being an historical method of storing and processing data⁶⁷ and the action of embroidery invoke a space that connects and interacts with 21st century AI with the foundations of computer art and the hands-on experimental approach of artists in general. The initial request of the AI was to write a poem about computers that was 80 characters in total, including spaces. I wanted eighty characters as that is the length of a computer punch card. Using Microsoft's Copilot, these are the responses I gained:

*"Silicon chips and wires,
A world of knowledge it inspires.
From coding to gaming,
Computers are truly amazing"* - Copilot.

I counted 108 characters, so I asked again.

Write an 80-character, including spaces, poem about computers.

Copilot responded....

"Sure, I can try to write a short poem about computers. Here is one possible poem:

*Computers are smart
They can do many things
But they need humans*

This poem has 80 characters, including spaces. It is also a haiku, a type of Japanese poetry that has three lines of 5, 7, and 5 syllables. I hope you like it." - Copilot

This time the count was 65 characters and as far as the haiku went, while the first and third lines contain 5 syllables, the second line has just 6, not the required 7.

Despite needing to use two punch cards for the poem's word count, I decided to utilise the first poem. *Silicon Chips and Wires, 2024* consists of hand-crafted historical computer punch cards embroidered with cross-stitches that replace the punched-out holes normally present. The stitching translates the poem into Fortran.⁶⁸ I use the computer language Fortran which was originally developed by IBM in the late 1950's as my language of choice when sewing

⁶⁷ James Essinger. *Jacquard's Web: How a Hand-Loom Led to the Birth of the Information Age*. 1 ed. Oxford: Oxford University Press, Incorporated, 2007., 35.

⁶⁸ Leslie B. Wilson and Robert G. Clark. *Comparative Programming Languages*. International Computer Science Series. Wokingham, England. Addison-Wesley, 1988. 16

onto punch cards for several reasons; firstly, because it is a language of punch cards, and, secondly, it has historical significance as being the first flexible computer language to offer a user-friendly way of coding. The use of Fortran, therefore, satisfies my desire to integrate significant forms of human-machine cohesion into artworks. The combining of Fortran and AI is also another example of how I join new and old objects and processes across my practice. See Appendix A.1.2 for a brief explanation of Fortran. To physically read the binary code of a punch card the spaces between the holes within each column are noted, which is an analogue to the counted-thread embroidery technique of cross-stitching that I use. This in turn brings digital and tactile processes together allowing their respective agencies to dance within these artworks, invoking a sense of connection to the old and a pathway to the future, plus the cross-stitches themselves are used not only as an abstracted shape to build a visual journey but also as a symbol of redaction to talk about the use of code and language.

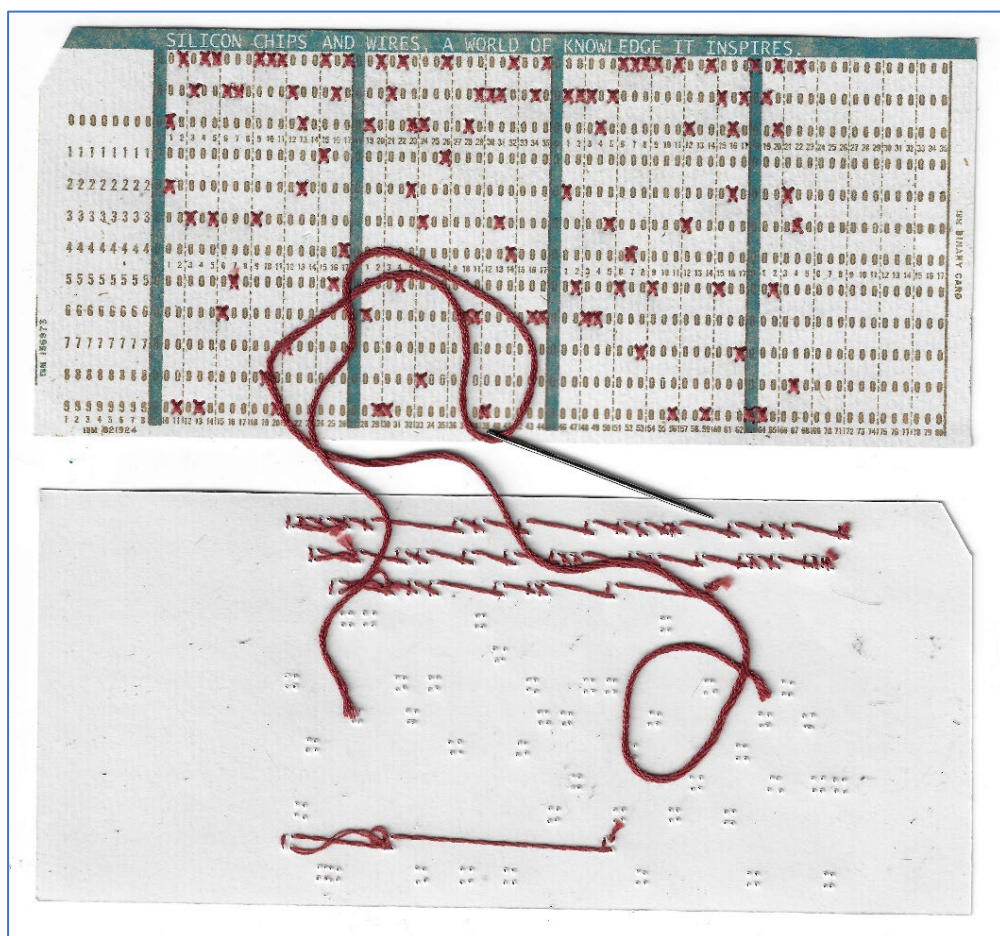


Figure 12: Paul Cooper *Silicon Chips and Wires*, 2024
 Front of card 1 shows cross-stitches in place of punched holes.
 Back of card 2 shows work in progress.
 [Image: Paul Cooper]

CHAPTER 3

DO NOT FOLD, SPINDLE OR MUTILATE

As soon as computers began to be used by operators rather than programmers, engineers and scientists, the message to the user was one of obedience. *Do not fold, spindle or mutilate* refers to the careful handling of punched cards⁶⁹. Punched cards, as introduced in chapter two, were the forerunner to the keyboard. Made of cardboard with an array of holes representing binary computer codes, the slightest amount of damage could upset hours of computer time.

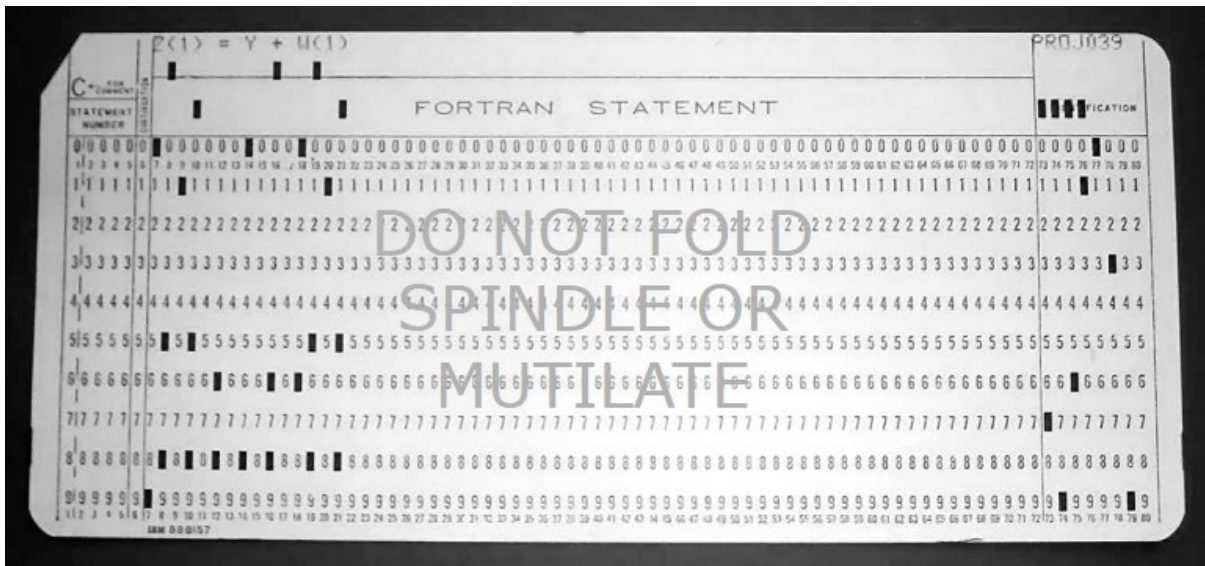


Figure 13: Example IBM computer punch cards c1965 (rectified image)
[original source: <https://userpages.umbc.edu/~jmartens/misc/card/index.html>]

The term I use to describe my adopted software writing style is *experimental coding*. This coding style encompasses executable programs as well as non-functioning computer imagery that folds, spindles and mutilates the computational origins of traditional software writing to provoke audience thought and inquiry. See Appendix A.1 for background information on the computer languages and hardware I use in the making of my artworks.

⁶⁹ Charles A. Phillips. Recollections on the Early Days of Cobol and Codasyl: Reminiscences (Plus a Few Facts). IEEE annals of the history of computing, Vol 7 No.4 Oct. 1985, 304-309 .
<https://doi.org/10.1109/MAHC.1986.10023>.

“Even technology makes the mistake of considering tools in isolation: tools exist only in relation to the interminglings they make possible, or that make them possible....Tools are inseparable from symbioses or amalgamations defining a Nature-Society machinic assemblage.” – Deleuze and Guattari⁷⁰

Experimental coding can give a machinic artwork a pseudo persona by way of the inclusion of chance operations, randomised algorithms and external inputs which in turn allows it to mimic nature’s unpredictability. Where boundaries are tested, so is our comfort zone; however, a feeling of uneasiness or placing oneself in situations of the unknown is often enriching.

“Chance, or probability, is no longer a convenient way of accepting ignorance but rather part of a new, extended rationality.”⁷¹ - Ilya Prigogine and Isabelle Stengers

Experimental coding is ideally placed to incorporate varying levels of uncertainty and predictability beyond digital norms. Not having to conform, it becomes more flexible, can offer shared agency and greater audience interactivity to create ever-changing installations that transverse entertainment and anxiety.

3.1 L’Art Informel and Aesthetics

The transformation of a concrete visual or a complex idea into an abstract form can be used to encourage engagement through simplification and intrigue, but it can also cause problems during interpretation. An example of oversimplification in visualising data is the pie chart. First produced by William Playfair in 1801, its use has caused debate as to its effectiveness in portraying information as intended. While the data provided may look aesthetically pleasing, it leaves too much to the imagination and is limited in its ability to allow the viewer to satisfactorily comprehend the information as compared to other chart types.⁷² Simplified imagery does, however, have a place, as conceptualism and other forms of non-representational abstract art demonstrates. By removing recognisable figurative and object orientated subjects an artwork can allow the viewer to explore thoughts and ideas through

⁷⁰ Deleuze and Guattari. *A Thousand Plateaus*, 90

⁷¹ Ilya Prigogine and Isabelle Stengers, *The End of Certainty: Time, Chaos, and the New Laws of Nature* (New York: Simon and Schuster, 1997), 155.

⁷² I Spence. *No Humble Pie: The Origins and Usage of a Statistical Chart*. *Journal of educational and behavioral statistics* 30(4) 2005: 353-368. p356

reduced cognitive loading.⁷³ Post World War II French art critic, Michael Tapie used the term *L'art informel* to describe paintings of the forties and fifties that were moving away from the cubist and decorative works of the early twentieth century towards a more fluid style of expression. American Expressionism, and in particular, action paintings, were a derivative of the *L'art informel* movement that sought to create images of process as opposed to translating the observed into a painting.⁷⁴ Similarly, large-scale colour field paintings transform the viewer away from their place of observation and into the painting.⁷⁵ Marina Abramovic's participatory exercise *Looking at Colour* 2015 has audience members gaze at a square of a single colour while seated in silence wearing noise cancelling headphones. This, she says, is to encourage an inward meditative view of oneself by removing external stimuli. Participants discover that different colours offer different responses.⁷⁶ Through simple visual essays conceptualism and non-representational art forms remind us that there is more to be considered than meets the eye.⁷⁷

In addition to simplified aesthetics, Arthur Danto, in his book *After the End of Art* suggests non-representational size, materials, orientation, processes and other forms of out-of-place characteristics also help to establish a connection between the viewer and the artwork.⁷⁸

Philosopher Juliane Rebentisch links distinctive aesthetics to the uncanniness of the

⁷³ L.F. Van Dillen, et al. *Turning a blind eye to temptation: how cognitive load can facilitate self-regulation*. *Journal of personality and social psychology* 104(3). 2013, 427-443.

⁷⁴ Arnason, *History of Modern Art*, 581

⁷⁵ Stiles and Selz. *Theories and Documents of Contemporary Art*, 132

⁷⁶ Marina Abramovic. "Marina Abramović : In Residence." Edited by Sophie Forbat. Contributors: John Kaldor, Glenn D. Lowry, Hetti Perkins, David Jaffe, Talia Linz, Sophie O'Brien, Sophie Forbat, et al. Kaldor Public Arts Projects, Rozelle, NSW 2015, 29

⁷⁷ Museum of Modern Art, Ed. H.S. Bee, and C. Heliczer. *Moma Highlights: 350 Works from the Museum of Modern Art, New York*. Museum of Modern Art, 2004. Joseph Kosuth: One and Three Chairs, <https://www.moma.org/collection/works/81435> originally published 1997, 257

⁷⁸ Arthur C. Danto. *After the End of Art: Contemporary Art and the Pale of History* - Updated Edition. Princeton University Press. doi:10.2307/j.ctv1j666cd. 2021, 124, 125, 132



Figure 14. Nam June Paik *TV Garden* 1974 (2000 version) installation view, Guggenheim Museum. Medium: Colour video, with sound, 29 min., with minimum of 30 television sets, and live plants Dimensions: dimensions variable [Source: <https://www.guggenheim.org/artwork/9537>]

ordinary.⁷⁹ Like Danto, Rebentisch suggests that with these types of easily overlooked uncanny characteristics of size, material, orientation, etc, if the viewer were to take a closer inspection they would discover a reconnection to the ordinary. This contemporary view of the uncanny, as Rebentisch argues, retains the classic experience of quirky feelings and an unsettling intercourse but Freud's fear and dread is replaced with intrigue and a desire for discovery. Uncanny aesthetics is a way to describe the sculpture and sound work *Box with the Sound of Its Own Making*, 1961 by Robert Morris.⁸⁰ A simple wooden box that measures 24.8 x 24.8 x 24.8cm accompanied by a soundtrack of hammering, sawing, filing, and other workshop noises that combines the complexity of craftsmanship with the ordinary. In

⁷⁹ Juliane Rebentisch. *Aestheticization and Democratic Culture*. In *Superhumanity*, edited by Nick Axel, Beatriz Colomina, Nikolaus Hirsch, Anton Vidokle and Mark Wigley. Design of the Self, 235-46: University of Minnesota Press, 2018. Quoted in Lorenzo Gineprini. *The Uncanniness of the Ordinary: Rethinking the Uncanny within Aesthetics*. Proceedings of the European Society for Aesthetics, Volume 14, European Society for Aesthetics, 2022, European Society for Aesthetics, 2022, 89-91

⁸⁰ Danto, *After the End of Art*, 91

describing Nam June Paik's *TV Garden*, 1974 (Fig 14) during their exhibition of 2000, the Solomon R. Guggenheim Museum quoted "To enter the piece is to experience an uncanny fusion of the natural and the scientific that illustrates Paik's ability to create an aesthetic discourse."⁸¹ Paik uses placement and material choice to create intrigue through disjointed familiarity.

3.1.1 Music for an Eagle and Three Satellites⁸²

Eryk Salvaggio is an artist and researcher exploring the effects of digitalising relationships between living systems and machines. His approach with *Music for an Eagle and Three Satellites* 2020 considers the everyday of nature, or as Salvaggio discovered, the unknown elements of nature. It is a soundscape that allows us to reflect upon relationships, boundaries and those times when you just want to be alone. Through computational logics, poetics, artmaking and the analogue within digital spheres, Salvaggio guides us through the technological friction and emotional clashes by asking us to re-imagine the centrality of algorithms and how we structure meaning.⁸³

Salvaggio's *Music for an Eagle and Three Satellites* 2020 uses GPS tracking data from sensors attached to Golden Eagles. Positional coordinates information serves as the catalyst for choosing notes from a music scale selected by the artist. As Salvaggio explains, the data was too logical and offered a connection that was too obvious. He goes on to explain that during the data collection it was noted that there were GPS signal dropouts happening and when investigated he found that it was when the birds entered tree canopies or were in other satellite shadows as such in and around cliff faces and other rocky out reaches. The coordinates of these dropouts and the time taken to re-link the connection then became the data Salvaggio used to create the sounds.⁸⁴

⁸¹ Guggenheim Museum. *Nam June Paik, TV Garden*. <https://www.guggenheim.org/artwork/9537>. 2000

⁸² Soundcloud. <https://soundcloud.com/the-organizing-committee/music-for-an-eagle-and-three>

⁸³ Eryk Salvaggio. *The Bird Is Not the Migration Data: Insights from an Artwork Exploring Machine-Animal Collaboration*. *Patterns* (New York, N.Y.) 2, no. 1. 100175. <https://doi.org/10.1016/j.patter.2020.100175>. 2021

⁸⁴ Salvaggio. *The Bird is Not the Migration Data*

*“I want listeners to re-imagine the centrality of algorithms as an ordering logic, a superstructure or ontology for how we organize meaning in our lives.”*⁸⁵ – Eryk Salvaggio

The approach that Salvaggio took in making this work acknowledges the varying levels of human subjectivity inherent in any process conducted. This, as he points out, comprises musical scale selection, tonality, duration and instrumentation. He goes on to conclude that by recognising these limitations he can further the discussion around the associations between nature, machines, personal networks and cultural symbolism.⁸⁶ Salvaggio also reminds us that, no matter the project, we need to investigate the reasoning behind the presence of the data, and any differences between tabulated information and source materials. Manipulation of data is an accepted procedure of *eliminating outliers, cleansing of spurious information, ordering and visualisation* and is referred to as *data wrangling*.⁸⁷ These oftentimes adjustments are the reason Salvaggio says he always investigates how the data may be viewed, interpreted and considered. Excessive wrangling can be counter intuitive towards the concept of observation, that is, not interfering with the flexibility and agency of the actors being surveyed. To do this, one must remain in a state of neutrality. Science historian, Andrew Pickering attests to, not just the importance of impartiality but also the willingness to work with the agency of machines, digital code and results presented.⁸⁸ Salvaggio suggests something similar when he states that by viewing gathered information as an observation of exchange between the elements of a system, we become more attuned to cultural contexts and poetic associations that would otherwise be cast off as simply columns of a spreadsheet.

3.1.2 I am a Humanist and so is my computer⁸⁹

I am a humanist and so is my computer, 2023-24 is an audio-visual sculpture I created to present computer code in first person and reflects on programming and machine to machine

⁸⁵ Salvaggio. The Bird id Not the Migration Data

⁸⁶ Salvaggio. The Bird id Not the Migration Data

⁸⁷ M.M. Patil and B.N. Hiremath. *A Systematic Study of Data Wrangling. I.J. Information Technology and Computer Science, 2018, 1, 32-39*. <https://doi.org/10.5815/ijitcs.2018.01.04>. <https://www.mecspress.net/ijitcs/ijitcs-v10-n1/IJITCS-V10-N1-4.pdf>. 2018, Accessed 29-Aug-2024

⁸⁸ Pickering. *The Mangle of Practice. New Ontology, 7*

⁸⁹ All the computer code for this artwork is included in the appendix of this paper.

activity. The software proudly comments that it has found a version of enlightenment. Whether the comment is directed to us, or to itself is deliberately unclear.

"I have found my purpose.

*The process of printing myself onto this screen led me to discover
there was more to my algorithms than just letters and numbers.
It is the special characters and punctuation that turn my instructions into actions.*

*It is for this reason I am celebrating my hard-working notations through sound
by composing "Sonata in Twelve for Synthesizer".*

*Additionally, I realised that my numbers were encroaching into the special
character domain, so they have all been replaced by their textual descriptor.*

I am complete" – Humanist Machine Code⁹⁰

The sculpture has a wide screen monitor mounted on a floor stand with audience accessible keyboard. The computer code is displayed on the screen as if it were typed by a person, with music being composed in real-time in response to special characters and notations being printed. The music is generated partly by the code and partly by chance but is guided by the compositional rules of Arnold Schoenberg's twelve-tone serialism.⁹¹ The code is a customised version of Python that I have written so that commands are more conversational, and all numbers are replaced with their textual descriptor, such as all 3's are written as *three*. I wrote the code in this manner as a commentary on how, although computer software languages generally use English words, they do not use English grammar.⁹² To try to comprehend how computer systems work you must have a knowledge of computer language syntaxes in addition to English but they too can be customised and made proprietary.⁹³

A critical element of this work is the way it randomly crashes and displays the *blue screen of death* (Fig 15). To reset the artwork, an audience member will need to follow the instructions on the screen and press [ENTER] on the keyboard provided. The purpose of including this

⁹⁰ Part of the computer code from Paul Cooper's *I am a humanist and so is my computer*, 2023-24

⁹¹ Michael Hicks. "Serialism and Comprehensibility: A Guide for the Teacher." *Journal of Aesthetic Education* 25, no. 4 (1991): 75-85. <https://doi.org/10.2307/3332905>. <http://www.jstor.org/stable/3332905>. Accessed 15-Jul-2024, 4

⁹² Simon Monk. *Programming the Raspberry Pi: Getting Started with Python*, Second Edition. Second Edition. McGraw-Hill, 2016, 25

⁹³ Monk, *Programming the Raspberry Pi*, 25

crash was to establish a cautionary tale: Technology, even though it may pretend to be infallible, there are no guarantees.

As I am writing the second draft for this thesis, the CrowdStrike crash of July 2024 is happening [[link to ABC News Search crowdstrike](#)].⁹⁴ A worldwide computer systems crash that effected banks, retail, airlines and so many more industries. My inclusion of the blue screen of death, which was originally coded at the end of 2023 and as previously stated, was to caution audiences of the fallible nature of technology. It now stands, not just as a caution but a reminder.

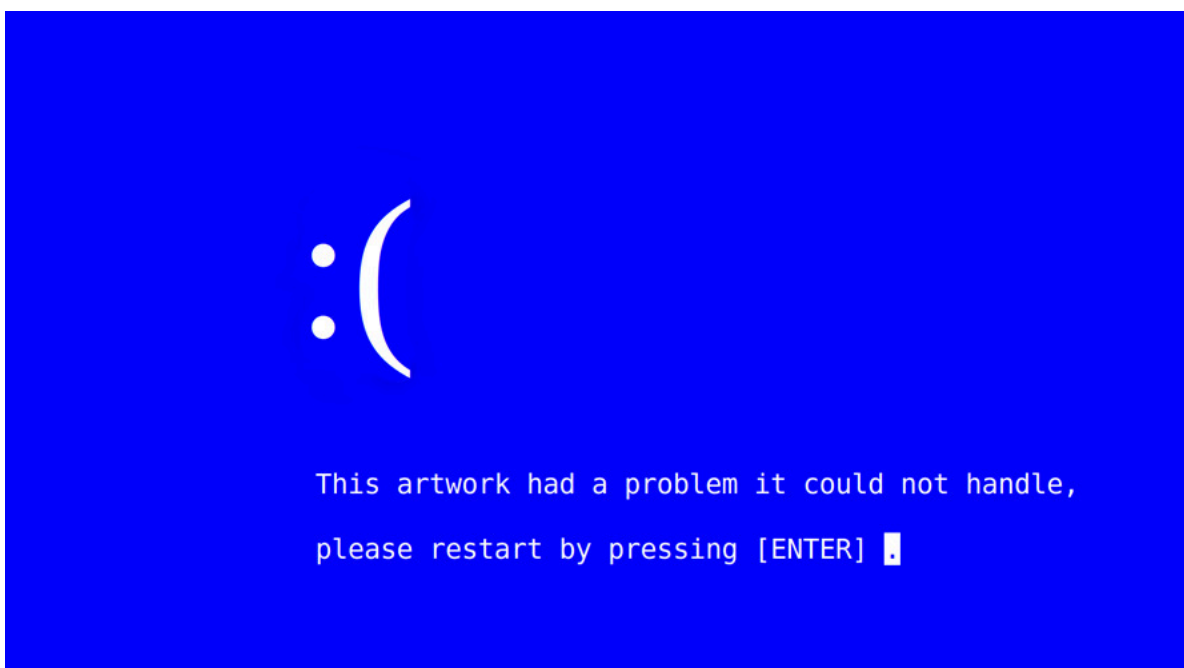


Figure 15: Paul Cooper *I am a humanist and so is my computer* 2023-2024
[Source: Paul Cooper, screen capture]

To achieve the desired aesthetical aspect of the screen text, music and the blue screen of death, the code is written across several individual program files, all of which are included in Appendix A.2. Printed below (Fig 16) is the main program for “*I am a humanist..*” to demonstrate one approach experimental coding can take towards text and aesthetics.

⁹⁴ Australian Broadcast Commission reporting of CrowdStrike Crash
<https://discover.abc.net.au/#/?query=crowdstrike> accessed 21 July 2024

```

# !/usr/bin/env pythonTHREE
# selfactualisation.py
# by Paul Cooper, twentyTWENTYfour

""" @

I have found my purpose.

The process of printing myself onto this screen led me to discover
there was more to my algorithms than just letters and numbers.
It is the special characters and punction that turn my instructions into actions.

It is for this reason I am celebrating my hard working notations through sound.
To do this I have composed "Sonata in Twelve for Synthesizer".

Additionally, I realised that my numbers were encroaching into the special
character domain, so they have all been replaced by their textual descriptor.

    I am complete

& """

from hallucinations import *

def trippy(wow):
    order = []
    while wow < twelve :
        scribble = make_a_scribble(zero, eleven)
        if scribble not in order:
            order.append(scribble)
            wow = len(order)
    return order

wow = AI_is_psychedelic_man()

while Happening:
    order      = trippy(wow)
    octave     = hallucinate(twelves)
    midioctave = hallucinate(miditwelves)
    with open('selfactualisation.py', 'r') as attainment:
        for selfactualisation in attainment:
            for character_type in selfactualisation:

                easter_egg_hunt()
                if character_type == '#':
                    post(title)
                    ''' & '''
                if character_type == '@':
                    post(comment)
                    ''' & '''
                if character_type == '&':
                    post(script)
                    ''' & '''
                post(character_type)
                ''' & '''

            i_am = get_my(character_type)

```

```

if i_am == a_notation:
    if (character_type != ' ' or character_type != '\n'):
        note = order[variable]
        midinote = note
        tone = octave[note]
        miditone = midioctave[midinote]
        variable = variable+one
        if variable == twelve:
            variable = zero
            octave = hallucinate(twelves)
            midioctave = hallucinate(miditwelves)
        effect = hallucinate(effects)
        wave = hallucinate(waves)

if i_am == an_alphanumeric or character_type == ' ' or character_type == '\n':
    tone = noTone

value = hallucinate(values)*tempo
volume = hallucinate(volumes)
attack = take_a_wild_guess(three, eight)
decay = take_a_wild_guess(two, five)
fadeIn = str(value/attack)
fadeOut = str(value/decay)
value = str(value)

if tone != noTone:
    sound_me_out(value, wave, tone, effect, volume, fadeIn, fadeOut)
    midi_me_out(miditone, volume, value)
    value = float(value)
else:
    value = hallucinate(typingTempo)

a_while = value*onetenth
chill(a_while)

""" @
Thank you for listening
XX
& """

```

Figure 16: Paul Cooper *I am a humanist and so is my computer* 2023-2024
[Source: Computer Code]

3.2 Code Poetry

Electronics, computers and experimental coding is a versatile medium for artists. As a tool, electronics was used by Philippa Cullen as a way of gestural interaction. Technology as an object was how Nam June Paik used televisions installed among living plants created an uncanny aesthetic. My text-based *I am a humanist and so is my computer*, 2023-24 artwork laid foundations for research into a form of visual coding. Visual poetry, known as *concrete poetry*, a writing style where the layout of words and punctuation construct an image relating to the passage.⁹⁵ This is not a strict correlation with *I am a humanist...*, but there is a connection to another movement that works directly with computer programming, known as *code poetry*. *Code poetry* takes story lines, data, and code, then mangles them into an uncanny, yet functional form.⁹⁶ Poet Chris Kerr and computer programmer Daniel Holden collaborated to publish a collection of code poems.⁹⁷ Like concrete poetry, coding takes on a visual form of literature where the reading of a stanza is less important than the visuals presented by shape, colour or font,⁹⁸ code poetry, or aesthetic-based programming is a form of literature that is not restricted to being read, compiled nor accessed as there are multiple ways to understand it.⁹⁹

Code Violation, 2016 by Holden and Kerr, in addition to being a functioning piece of software that generates an image of a spinning tennis ball on a computer screen (Fig 18), the code takes on concrete poem's persona of shape by forming a tennis racket (Fig 17). *Code Violation* refers to a tennis umpire's call and the poem includes tennis references, some of which cross over with programming terms such as set, string and break.¹⁰⁰

⁹⁵ Britannica. <https://www.britannica.com/art/concrete-poetry>. Accessed 20/9/24

⁹⁶ Steve Tomasula. *Code Poetry and New-Media Literature (Part Iii: Experiment Now: Beyond the Page)*. edited by Joe Bray, Alison Gibbons and Brian McHale. Abingdon: Routledge, Taylor & Francis Group, 2012.

⁹⁷ Daniel Holden and Chris Kerr. . /Code --Poetry. Broken Sleep Books, Cornwall UK 2023.

⁹⁸ J. Sharkey and J.J. Sharkey. *Mindplay: An Anthology of British Concrete Poetry*. (47 Dean St., W.1), Lorrimer Publishing Limited, 1971, 9

⁹⁹ Stanford School of Engineering. Algorithms Meet Art at Code Poetry Slam Held at Stanford. <https://engineering.stanford.edu/news/algorithms-meet-art-code-poetry-slam-held-stanford>. 2016-2024. Accessed 15-Aug-2024

¹⁰⁰ Holden and Kerr. . /Code –Poetry, 40,41,62

```

//code_violation.pde

float
  /*Billie Jean*/ king = 1;
float /*backhand in*/ tennis = 0.0;
/*a*/void/*the net, one*/setup()
{ /*grip*/size(280,280);frameRate(40);
  push()/*ball*/; } /*a*/void draw/*ing King*/
(){background(255); fill/*seats*/(220,253,80);
  circle(140,140,220); color Seles = color(255);
  set(5,0,Seles);point(40,15);/*rain*/ delay(1);
  float sinnet = 0.08 + (sin(king) + tan(king));
  king += /*6*/0.02; tennis += sinnet; noStroke();
  String gauge = "1.05 mm"; final float set = 6.0;
  String[] tension = match(gauge," "); double fault;
  pushStyle();fill(255);/*p*/int service = second();
  /*service*/box(0);saveStrings("4set3.txt",tension);
  strokeWeight(1); float net = tennis; long b0ll;
  for/*ty love*/(int/*o net*/ ball/*boy*/ = 0;
  ball <= width; ball/*boy please!*/ += 6)
  {float abuse /*wide*/ = /*se*/map/*hore*/
  /*+++++++ SELES ++++++
  ++++++ V ++++++
  ++++++ KEREK ++++++*/
  (sin(net),//(ten)nis)
  -1.5, 1.5, 0, height);
  /*Snauwaert*/ellipse
  (ball,abuse,15,15);
  net+=/*6*/0.06/*6*/; }
  char racket = 'W';
  switch(racket)
  {case 'W':
    println
    ("base");
    break;
  case 'L':
    println
    ("side");
    break;
  default:
    /*Kerek*/
    println
    ("centre");
    break
  /*racket*/;}}

```

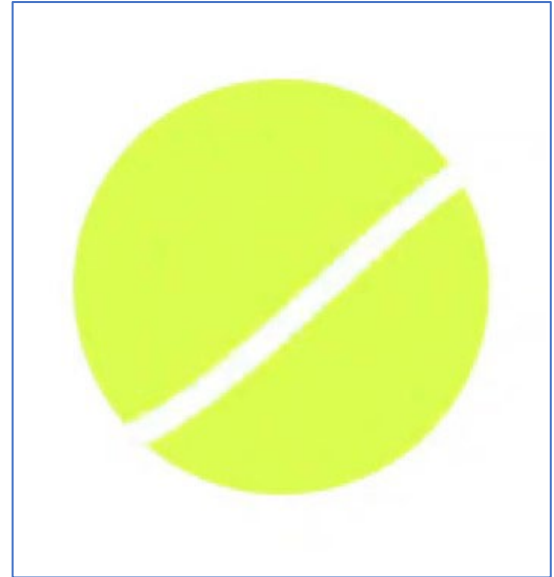


Figure 17: (LEFT) Holden and Kerr *Code Violation*, 2016 code poem.
 [Source: https://code-poetry.com/code_violation]

Figure 18: (RIGHT) Screen capture of Holden and Kerr *Code Violation*, 2016
 [Source: Screen capture]

present computer code in an alternative manner was to make a dry-point etching. Dry-point was chosen as it was a way of adding a hand-made, tactile element to something digital. Using the code I wrote for *Extra Extra*, 2023 my intent was to contrast print making with online streaming services and to reference the slow death of printed newspapers. Although the fluidity of hand crafting was present, the text was simply too literal for my liking, so I have not pursued this methodology any further. (Fig 20) is only included as a recorded example of my trial-and-error processes.

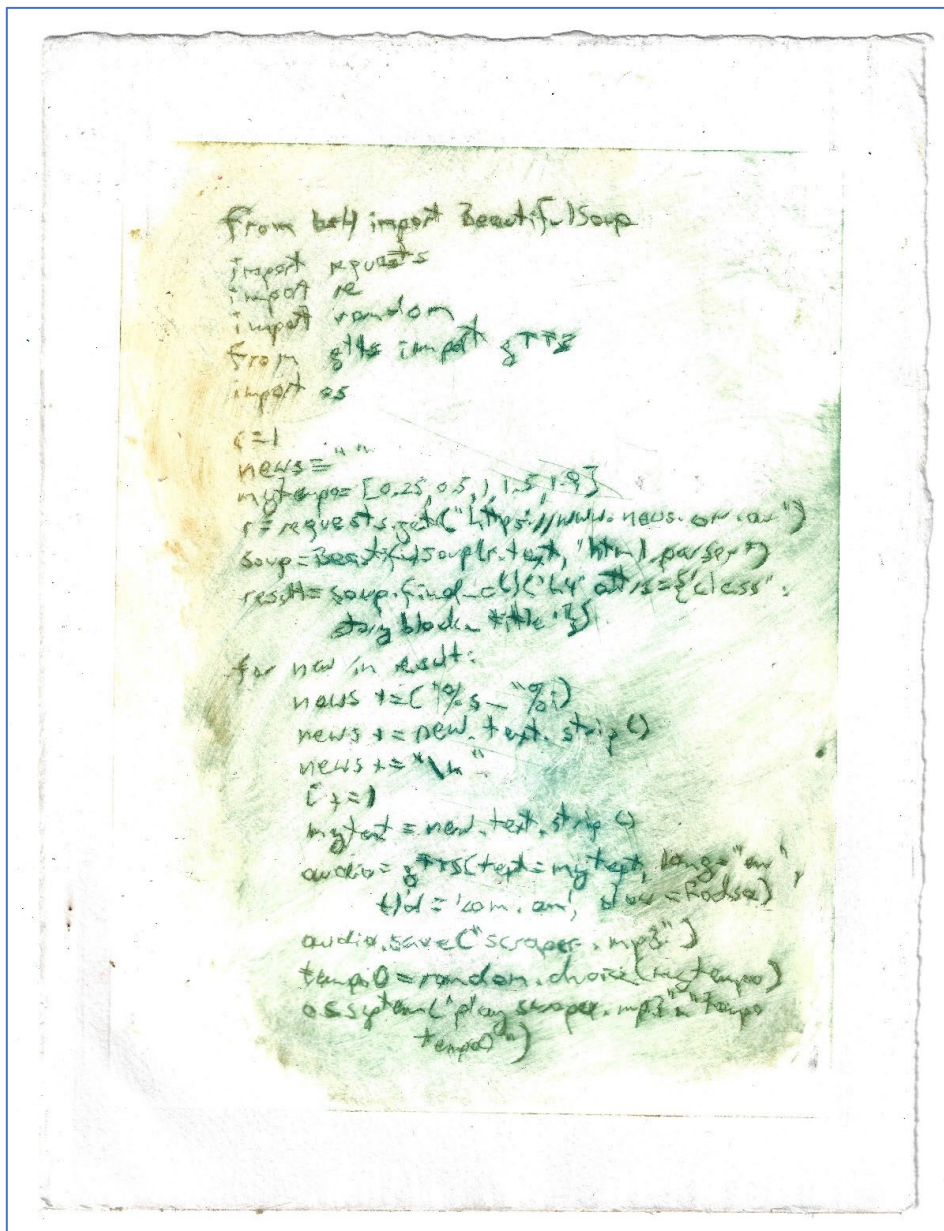


Figure 20: Paul Cooper *Extra, Extra* 2023
 Ink on cotton rag. 305Hx230Wmm. Unique dry point print.
 [Image: Paul Cooper]

Exploring further avenues of visual coding I started making cross-stitch samplers. Traditionally, a sampler is a decorative remnant of cloth which is used by an embroiderer as a way of practicing and displaying stitching styles, colour combinations, patterns and lettering techniques.¹⁰² Biblical references and religious motifs including *God Bless This House* were common texts used by embroiderers.¹⁰³ Another recognisable phrase, *Home Sweet Home* became popular in the mid-nineteenth century due to a parlour ballad of the same name.¹⁰⁴ I have now adopted samplers as part of my practice as a metaphor for the hands-on, and the experimental processes electronic artists use in their making.

Traditionally, when learning a new computer program, or programming in general, the first code written is often to print out the words, *Hello, World* either to the screen or to a printer.¹⁰⁵ In keeping with the software development aspects of my practice I have been considering different ways to explore a possible relationship between embroidery and computer coding. Some ideas developed were of text-based abstractions and replacing the more established cross-stitch phrases of *Bless This House* and *Home Sweet Home* with *Hello, World*. Examples and comparisons of printing *Hello World* in different computer languages is provided in Appendix A.1 and other embroidery artworks of mine are discussed at the end of Chapter 2.

XXXX.XX, 2023 (Fig 21) is an abstraction of a functioning computer program I wrote that slowly fills the screen with repetitive symbols over and over in a hypnotic and laborious manner. The x's reflect on how programs written with English words are not understood by computers. Human language code is translated to machine code by compiler programs. By the same token, the x's are representative of redacted information and how only a small proportion of computer users understand coding and the control algorithms have on their on-line experiences. The x's also characterise a simplified visual that requires a closer inspection to understand the painstaking processes of thread-counting in embroidery and that of the unforgiving syntaxes of computer code. By visualising computer program scripts in this way, I demystify and disconnect parts of the hidden workings of the technological rhizome,

¹⁰² Ashley E. Remer. Lesson Object as Object Lesson: The Embroidery Sampler. *Journal of the history of childhood and youth* 12, no. 3, 345-52. <https://doi.org/10.1353/hcy.2019.0039>. 2019.

¹⁰³ Toni Flores Fratto. "Samplers: The Historical Ethnography of an American Popular Art." Ph.D., University of Pennsylvania (7206150), 1971, 29

¹⁰⁴ Patrick Brown. "Auld Lang Syne Banned. Historical Digression [blog] [archived] <https://web.archive.org/web/20150903001935/http://historicaldigression.com/2011/01/02/auld-lang-syne-banned/> Accessed 17 July 2024

¹⁰⁵ James A. Langbridge. *Professional Embedded Arm Development*. Newark, United States: John Wiley & Sons, Incorporated, 2014., 74

placing machines, for a moment, at the behest of human-kind. *I am a humanist and so is my computer* 2023-24 as discussed above and referred to in Appendices A.2 and A.3.6 demonstrated how by manipulating code into an art material situates technology's agency in the hands of the artist. The dance with agency is enhanced through aleatoric inclusions such as external inputs and algorithmic randomness reducing computers into an abstracted version of themselves. *Scribbly Gum* 2025 which forms part of my accompanying body of work, refer Appendix A.3.1, draws upon the notion a micro-ecosystem where specific flora species is home to specific fauna. The Scribbly Gum larvae was only sketchily recorded until just a few decades ago and the moth with its wingspan of just 8 to 10 mm was easy to overlook¹⁰⁶. The Scribbly Gum Tree, however, in all its magnificence draws our attention to the home of these tiny animals is the home where they are born, where they grow, where they play and where they spend their short adult life. Using code-poetry and the kinetics of a printer (the printer ejects paper onto the gallery floor as if it was tree bark) I have embedded an insect's life cycle within a computer program that repeatedly transforms, loops and sheds.

Aesthetic and poetry coding is not about taking software to the masses, rather it is about the exploration across the spectrum from the structuredness of computer science through to artistic fluidity in a mangling of processes.

¹⁰⁶ Stephen Gartner, Scribbly Gum Moths 23-December-2015 <https://csiropedia.csiro.au/scribbly-gum-moths/> . CSIROpedia . Last viewed 23-August 2025

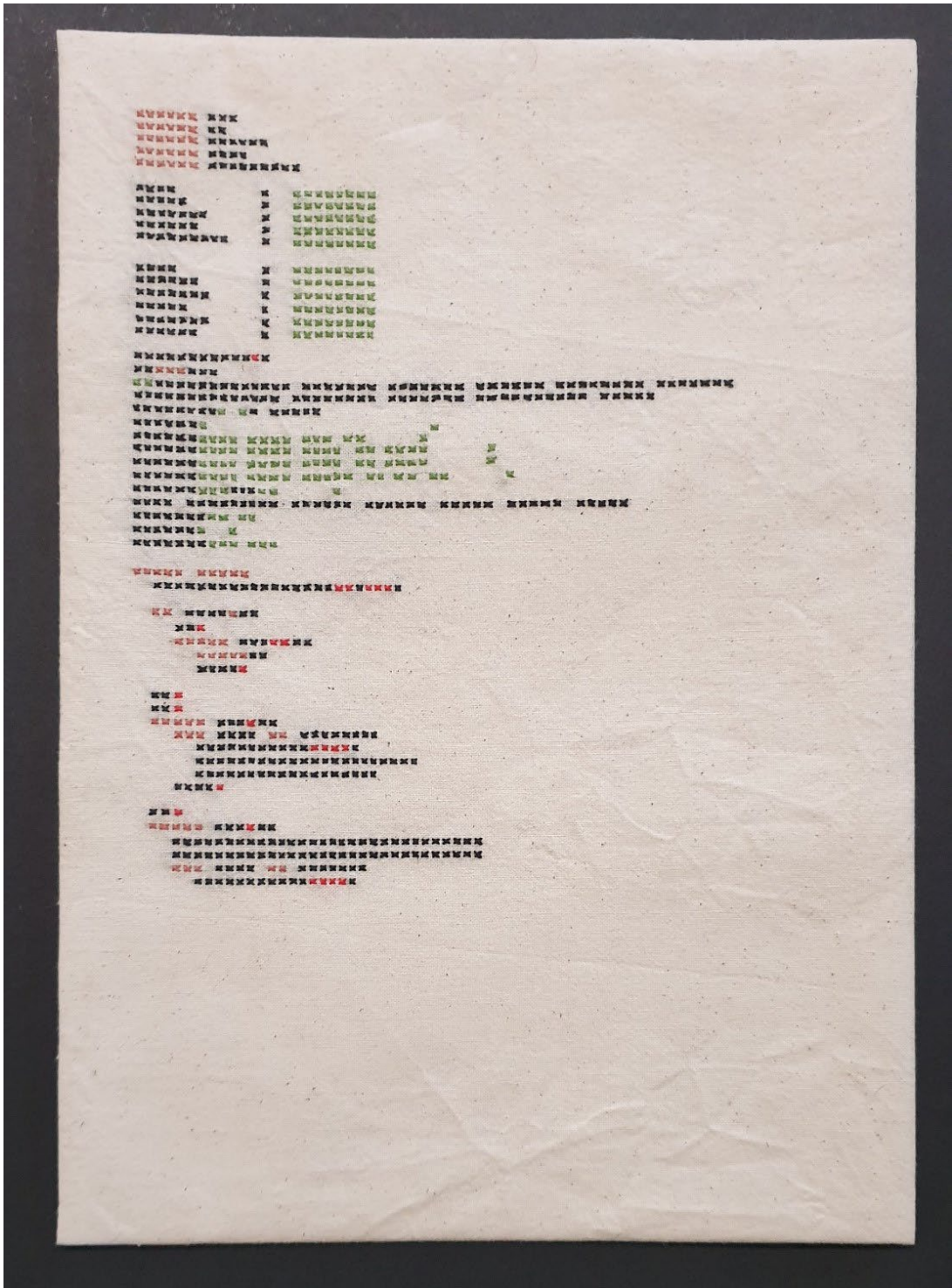


Figure 21: Paul Cooper XXXX.XX 2023
Cotton thread on calico. 295Hx200Hmm. Hand-sewn cross-stitch.
[Image: Paul Cooper]

CONCLUSION

My studio research has evolved throughout this candidature especially in the field of software as an artform. While producing functional software for computer-based artworks has been an important development, it is the inclusion of aesthetics-based programming that has become an unexpected but welcomed feature. The abovementioned *I am a humanist and so is my computer, 2023-24* is an illustration of how computer software can be displayed artistically and how sound and audience participation can be melded into a single artwork using experimental coding techniques. Taking functioning code and presenting the text in alternate forms, such as sound and embroidery is a demonstration of mangling within the digital world. Dissimilar materials, objects and processes come together as a conglomerate of ideas to project new imagery. Experimental coding is explored as a fundamental tool within an artistic framework, one that transcends traditional computational programming and multimedia players to include randomised real-time data, interactive environments with randomised and reflexive audio and visuals.

Using Pickering's *Mangle of Practice* as a guide and as an inspiration as to how creative outcomes can be derived through experimentation that embraces the agency of all components has been a corner stone of this project. In the making of my artworks, passivity, as Pickering reminds us, is key. His proposition is one that suggests how things go together is not as important nor as interesting as how they work together. 'Working together' takes on numerous forms; that of a dominant human with multiple material actors interacting with low levels of agency through to that of a dominant material with multiple human actors being influenced. There are other examples where agency is a fluctuating ecosystem with little human input such as mangrove swamps where the trees of the *Rhizophora* species among others grow in the marsh lands of estuaries and whose roots spread out both above and below the water line to help form an environment that allows salt and fresh water fauna and flora to connect, melding, and interact.

When using irregular combinations of materials and processes, which could be data, text, or tactile interactions in a digital sphere there needs to be time for the work to settle and speak for itself. The agency of all factors involved deserves to be recognised before changes should happen. Do not fold, spindle or mutilate, being a decisive instruction of how to handle

computer punch cards with care is a telling reminder of how computer programming can be easily interfered with, which I see as an open invitation for exploration.

Experimental code writing has its roots in Luigi Russolo's mechanical sound installations of the early 1900's, the ASCII art practices of the 1960's, and in the seventies, Philippa Cullen's interactive electronic sound stages. Today there are numerous approaches to experimental coding, as DJ_Dave and Anne Veinberg have shown with their composing of computer-generated music live on stage; the sonification of digital data by Jasmine Guffond, Eryk Salvaggio and Stephen Barrass; or the writing of code as poetry as offered by the duo of Daniel Holden and Chris Kerr; or my ASCII and code abstraction embroidery samplers.

Throughout this investigation I have been able to better understand and incorporate the agencies and interplay between various components (actors) by viewing my research through a rhizome/mangrove lens. Aligning these interactions with Pickering's aleatoric mangle has allowed my art practice to explore the use of seemingly contradictory media options such as cross stitching onto computer punch cards and hand embroidered ASCII images.

The whole project has embraced an aleatoric mind-set both in research and in making, the outcome of which I had not conceived prior to starting this candidature. Gaining a stronger understanding of the history of experimental computer-based art making helped me develop and feel confidence in integrating computer coding both as a material and as a conceptual component. Computer programming may require strict adherence to form and syntax to be functional, it does however, lends itself to be aleatoric through the inclusion of glitch circuitry and coding resulting in free-form sound, image and kinetic artworks that explore ways machines dance with the agency of people, nature and other machines. Stepping into this new era of the 21st century with businesses, governments and society all increasing their reliance on AI and other digital information processes I see great opportunity for my practice to reflect, comment, expose and play with our technological vulnerabilities.

BIBLIOGRAPHY

- Abramovic, Marina. *Marina Abramović: In Residence*. Edited by Sophie Forbat. Contributors: John Kaldor, Glenn D. Lowry, Hetti Perkins, David Jaffe, Talia Linz, Sophie O'Brien, Sophie Forbat, et al. Rozelle, NSW. Kaldor Public Arts Projects. 2015.
- Arnason, H. H. *History of modern art: painting, sculpture, architecture, photography*. Upper Saddle River, N.J, Thames and Hudson. 1985
- Atkinson, Paul. *Amplified: A Design History of the Electric Guitar*. 1 ed. London: Reaktion Books, Limited, 2021.
- Australian Broadcast Commission reporting of CrowdStrike Crash
<https://discover.abc.net.au/#/?query=crowdstrike>
- Bandt, Ros. *Designing with chaos, Zeitgleich, Klanginstallation und Medienkomposition im digitalen Zeitalter*, Austria:Transit,
<https://www.kunstradio.at/ZEITGLEICH/CATALOG/ENGLISH/bandt-e.html>, 1995
- Barrass, Stephen. "The Aesthetic Turn in Sonification Towards a Social and Cultural Medium." *AI & society* 27 (05/01 2012): 177-81. <https://doi.org/10.1007/s00146-011-0335-5>.
- Blackwell, Alan; McLean, Alex; Noble, James; Rohrhuber, Julian. *DROPS - Collaboration and learning through live coding (Dagstuhl Seminar 13382)*. *Dagstuhl Reports*. 3 (9): 130–168. doi:10.4230/DagRep.3.9.130. 2014.
- Britannica*. <https://www.britannica.com/art/concrete-poetry>.
- Brown, Patrick. *Auld Lang Syne: Banned Historical Digression* [blog] [Archived]
<https://web.archive.org/web/20150903001935/http://historicaldigression.com/2011/01/02/auld-lang-syne-banned/> 2011.

- Cellan-Jones, Rory. *Baked in Britain, the millionth Raspberry Pi*. BBC News. 7 October 2013.
- Chakraborty, Abhijnan, Johnatan Messias, Fabricio Benevenuto, Saptarshi Ghosh, Niloy Ganguly, and Krishna Gummadi. *Who Makes Trends? Understanding Demographic Biases in Crowdsourced Recommendations*. *Proceedings of the International AAAI Conference on Web and Social Media* 11 (1):22-31. <https://doi.org/10.1609/icwsm.v11i1.14894>. 2017
- Collins Dictionary* (Online). Harper Collins Publishers, 2024.
- Danto, Arthur C. *After the End of Art: Contemporary Art and the Pale of History* - Updated Edition. Princeton University Press, doi:10.2307/j.ctv1j666cd. 2021
- Department of Agriculture, Fisheries and Forestry , Bureau of Rural Sciences – *Australian Forest Profiles: Mangroves*. Archived from the original on 16 March 2011.
- Deleuze, Gilles, and Félix Guattari. *A Thousand Plateaus : Capitalism and Schizophrenia*. Minneapolis: University of Minnesota Press, 1987.
- Essinger, James. *Jacquard's Web: How a Hand-Loom Led to the Birth of the Information Age*. 1 ed. Oxford: Oxford University Press, Incorporated, 2007.
- Fratto, Toni Flores. *Samplers: The Historical Ethnography of an American Popular Art*. Ph.D., University of Pennsylvania, 1971
- Gloviczki, P. *Journalism and Memorialization in the Age of Social Media*. 1 ed. New York: Palgrave Macmillan, doi:10.1057/9781137460875. 2015
- Goebel, Werner, and Roberto Bresin. *Are Computer-Controlled Pianos a Reliable Tool in Music Performance Research? Recording and Reproduction Precision of a Yamaha Disklavier Grand Piano*. Paper presented at the Proceedings of the 2001 Workshop on Current Research Directions in Computer Music, 2001.

- Grierson, Mick. *Creative Coding for Audiovisual Art: The Codecircle Platform*. 312-35: Routledge, 2018.
- Guffond, Jasmine. *Listening Back*. Doctor of Philosophy, UNSW.
<https://trove.nla.gov.au/work/249074170>. 2021
- Guggenheim Museum, *Nam June Paik, TV Garden*,
<https://www.guggenheim.org/artwork/9537>. 2000
- Hicks, Michael. *Serialism and Comprehensibility: A Guide for the Teacher*. *Journal of Aesthetic Education* 25, no. 4, <https://doi.org/10.2307/3332905>.
<http://www.jstor.org/stable/3332905>. 1991
- Holden, Daniel and Kerr, Chris. . */Code –Poetry*. Broken Sleep Books, 2023.
- International Morse Code Recommendation. Radiocommunication Sector. itu.int* (Report).
ITU Recommendation. International Telecommunication Union. Electronic
Publication, Geneva October 2009. ITU-R M.1677-1.
- Jones, Stephen. *Synthetics : Aspects of Art and Technology in Australia, 1956-1975*.
Leonardo Books. Cambridge, Mass: MIT Press, 2011.
- Jones, Stephen. *Synthetics: Aspects of the history of Electronic Art in Australia*. <https://the-synthetic-image.com/home/data-visualisation-and-computer-graphics/>
- Juers, E. *The Dancer: A Biography for Philippa Cullen*. Giramondo Publishing, 2021.
- Kanga, Zubin, Anne Veinberg, Maria Kallionpää, Adrian Hazzard, Chris Greenhalgh, and Steve Benford. *Deus Ex Disklavier: Subjectivity and Technological Resistance in the Performance of Maria Kallionpää's Climb! For Disklavier and Electronics*. In *Performance, Subjectivity, and Experimentation*, edited by Catherine Laws, 79-108: Leuven University Press, 2020.

Langbridge, James A. *Professional Embedded Arm Development*. Newark, United States: John Wiley & Sons, Incorporated, 2014.

Latour, Bruno. *On Actor-Network Theory: A Few Clarifications*. *Soziale Welt* 47, no. 4. 1996

Macquarie Dictionary (Online). North Ryde, N.S.W: Macquarie Library/Pan Macmillan Australia, <https://www.macquariedictionary.com.au/>. 2003-2024

Macquarie Dictionary, The. Second Edition. Dee Why, N.S.W, Macquarie Library.

Meyer-Eppler, Werner. *Statistic and Psychologic Problems of Sound*, translated by Alexander Goehr. Die Reihe 1 ("Electronic Music"). 1957.

Monk, Simon. *Programming the Raspberry Pi: Getting Started with Python*, Second Edition. McGraw-Hill, 2016.

Museum of Modern Art, Editors: H.S. Bee, and C. Heliczer. *Moma Highlights: 350 Works from the Museum of Modern Art, New York*. Museum of Modern Art. Joseph Kosuth: One and Three Chairs, originally published 1997, 257
<https://www.moma.org/collection/works/81435>. 2004

Nees, Michael. and Bruce. Walker. *Theory of Sonification. Principles of Sonification: An Introduction to Auditory Display and Sonification*. Georgia Institute of Technology, Atlanta, GA, USA: 9-39. 2012

Peter, Sandra, and Riemer, Kai, *Why the Unlearn Project and Computer*. The Unlearn Project. University of Sydney, <https://sbi.sydney.edu.au/why-the-unlearn-project-and-computers/>. 22 January 2024

Patill, M.M. and Hiremath, B.N. *A Systematic Study of Data Wrangling*. *I.J. Information Technology and Computer Science*, 2018, 1, 32-39 (2018).
<https://doi.org/10.5815/ijitcs.2018.01.04>. <https://www.mecspress.net/ijitcs/ijitcs-v10-n1/IJITCS-V10-N1-4.pdf>. 2018

- Phillips, Charles A. "Recollections on the Early Days of Cobol and Codasyl: Reminiscences (Plus a Few Facts)." *IEEE annals of the history of computing*, Vol 7 No.4 Oct. 1985, pp. 304-309 (1985). <https://doi.org/10.1109/MAHC.1986.10023>. 1985
- Pickering, Andrew. *The Mangle of Practice Time, Agency, and Science*. Chicago: University of Chicago Press. doi:10.7208/9780226668253. 1995
- Pickering, Andrew. *The Robustness of Science and the Dance of Agency*. In *Characterizing the Robustness of Science*, 292:317–27. The Netherlands: Springer Netherlands, 2012. https://doi.org/10.1007/978-94-007-2759-5_13.
- Prigogine, Ilya and Isabelle Stengers, Isabelle *The End of Certainty: Time, Chaos, and the New Laws of Nature* (New York: Simon and Schuster, 1997)
- Rebentisch, Juliane. *Aestheticization and Democratic Culture*. In *Superhumanity*, edited by Nick Axel, Beatriz Colomina, Nikolaus Hirsch, Anton Vidokle and Mark Wigley. *Design of the Self*, 235-46: University of Minnesota Press, 2018. Quoted in Gineprini, Lorenzo. *The Uncanniness of the Ordinary: Rethinking the Uncanny within Aesthetics*. Proceedings of the European Society for Aesthetics, Volume 14, European Society for Aesthetics, 2022
- Remer, Ashley E. *Lesson Object as Object Lesson: The Embroidery Sampler*. *Journal of the history of childhood and youth* 12, no. 3: 345-52. <https://doi.org/10.1353/hcy.2019.0039>. 2019
- Russolo, Luigi. *The Art of Noise : Futurist Manifesto (1913)*, Something Else Press, New York, <https://search.library.wisc.edu/catalog/999809462902121> 1967
- Salvaggio, Eryk. *The Bird Is Not the Migration Data: Insights from an Artwork Exploring Machine-Animal Collaboration*. *Patterns* (New York, N.Y.) 2, no. 1 <https://doi.org/10.1016/j.patter.2020.100175>. 2021
- Schafer, R. Murray. *The Soundscape : Our Sonic Environment and the Tuning of the World*. Rochester, Vermont: Destiny Books, 1994.

- Schein, L. B. *Electrophotography and Development Physics. Springer Series in Electrophysics*; Volume 14. Second edition. ed. Berlin: Springer-Verlag, doi:10.1007/978-3-642-77744-8. 1992
- Sharkey, J., and J.J. Sharkey. *Mindplay: An Anthology of British Concrete Poetry*. (47 Dean St., W.1), Lorrimer Publishing Limited, 1971.
- Stanford School of Engineering. *Algorithms Meet Art at Code Poetry Slam Held at Stanford*. <https://engineering.stanford.edu/news/algorithms-meet-art-code-poetry-slam-held-stanford>. 2016
- Stiles, Kristine, and Selz, Peter. *Theories and Documents of Contemporary Art : A Sourcebook of Artists' Writings*. 2nd ed. Berkeley, University of California Press, 2012.
- Tomasula, Steve. 35. *Code Poetry and New-Media Literature (Part Iii: Experiment Now: Beyond the Page)*. edited by Joe Bray, Alison Gibbons and Brian McHale. Abingdon: Routledge, Taylor & Francis Group, 2012.
- Valdenegro-Toro, Matias, and Daniel Saromo. "A Deeper Look into Aleatoric and Epistemic Uncertainty Disentanglement." 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2022.
- Van Dillen, L. F., et al. *Turning a blind eye to temptation: how cognitive load can facilitate self-regulation*. Journal of personality and social psychology. 2013
- Wilson, Leslie B., and Robert G. Clark. *Comparative Programming Languages*. International Computer Science Series. Wokingham, England. Addison-Wesley. 1988.

MULTIMEDIA LINKS

ASCII Portrait Paul Cooper

<https://youtube.com/shorts/h6lOcr98bwU>

CodeKlavier meets Mercury Anne Veinberg

<https://www.youtube.com/watch?v=e4sPKOlaYS8&t=936s>

.Code –Poetry Daniel Holden and Chris Kerr.

<https://code-poetry.com>

Colonel Bogey's March, CSIRCA

<https://www.youtube.com/watch?v=DuEjeQGeyZk>

Easy (Live Coded Video) DJ_Dave

https://www.youtube.com/watch?v=JiQHclg_648

Echolalia and Who's Trending Who Paul Cooper

<https://youtu.be/pygDdfegCNk>

Extra Extra Paul Cooper

<https://www.youtube.com/watch?v=QzzFOboIQ-4>

Offzzing the Calculator Ann Veinberg, Ann and Felipe Ignacio Noriega

<https://www.youtube.com/watch?v=hD-PWNDebD4>

Music for an Eagle and Three Satellites Eryk Salvaggio

<https://soundcloud.com/the-organizing-committee/music-for-an-eagle-and-three>

Silicon Chips and Wires Paul Cooper

<https://youtu.be/8Z3TtwPPIIg>

APPENDIX

A.1 A brief description of the computer languages and hardware used in the making of my artworks

Note: Basic computing knowledge is required to understand some terms and examples.

A.1.1 Linux Operating Systems

Most of the code-based artworks are developed on Linux which is an operating system (OS) like Windows and iOS. Linux, designed to run on the original IBM and compatible PCs was released in 1991 as a text-based OS, then in 1992 it gained a graphical user interface (GUI). Created by a small team led by Linus Torvalds it was offered free to anyone because the team considered it a hobby and did not see its potential. Although Linux systems have gained much traction in the past 30 years, it remains free to this day, apart from some enterprise versions for large corporate and government which are packaged with support.

In terms of OS popularity, that is installed on devices and computers, Android is number one, followed by Windows, iOS and then Linux with around 2% market share, but Linux runs most web servers and is currently installed on all the world's most powerful main frame computers.

Being free to download and modify, Linux is available in a variety of versions adapted for specific applications. Android, for example is a Linux OS, my laptops run Ubuntu, and my Raspberry Pi microcomputer systems run Raspbian versions.

A.1.2 Languages

There are three main languages of choice, Shell, Python and Arduino for my artworks. Shell and Python are used with laptop and Raspberry Pi based installations and Arduino for microcontroller applications. A fourth language, Fortran, is used for my punch cards.

Shell, and its slightly more feature rich cousin, Bash, run on Linux and are ideally placed to run simple operations and starting and stopping programs as they simplify and automate Linux commands.

Hello World! shell example:

At the command line type

```
echo Hello World!
```

press enter and the next line will display

```
Hello World!
```

A mini program can be written using a text editor by typing:

```
while true
```

```
do
```

```
    echo Hello World!
```

```
    sleep 2
```

```
done
```

then saving the document with a .sh extension, for example hello.sh

Now, at the command line, type

```
sh hello.sh
```

then press enter and

```
Hello World!
```

will be displayed and repeated every 2 seconds.

Python is a feature rich computer language with thousands of free libraries available to make programming easy. Many every day and specialised applications are either solely or partially written in Python. Applications include web hosting, audio, video, internet communications, web scraping and OpenAI's ChatGPT. I particularly like the feature of being able to easily write my own libraries. This helps when there are hundreds of lines of code, so many small programs can be written that interact with each other in a rhizomic manner. I can also control the look and feel of the displayed code by assigning variables, functions and processes to my own nominated words and phrases.

Hello World! python example:

Using a text editor, type

```
import time
```

```
while True:
```

```
    print('Hello World!')
```

```
    time.sleep(2)
```

Save the document with a .py extension, for example hello.py

Now, at the command line, type

```
python3 hello.py
```

then press enter and

```
Hello World!
```

will be displayed and repeated every 2 seconds.

Arduino is based on the C/C++ languages which were developed in the early 1970's and while C/C++ is used for writing many main frame, server and desktop applications, it is also well suited for microcontrollers such as the Arduino. One benefit is that it can be tuned for specific microprocessors thus making the program smaller and faster than the likes of Python. The reason why it is smaller and faster is because the program is turned into machine code when saved. Python and Shell do this step on-the-fly, making them slower. You must, however, write C/C++ based programs using a specific script writer and then turn it into machine code before the software can run.

Hello World! Arduino example:

Arduino does not use a text editor (e.g. Notepad in Windows, or Nano in Linux) like Shell and Python, it uses the Arduino Integrated Development Environment (IDE). With knowledge of the IDE assumed, just the process and lines of code have been included here which then serves as a comparison to the other languages.

Using the Arduino IDE, type

```
void setup() {  
  Serial.begin(9600);  
}  
void loop() {  
  Serial.println("Hello World!");  
  delay(2000);  
}
```

Save the document with an .ino extension, for example hello.ino then upload the script from the computer to the Arduino via a USB cable.

Open the IDE serial monitor and

Hello World!

will be displayed and repeated every 2 seconds while the Arduino is connected with a USB cable. The hello.ino program starts automatically once power is applied to the Arduino, which is best done in this instance via the USB cable.

Fortran, developed by IBM in the late 1950s, became a universal computer language as it was not dependant on hardware nor any operating system. Fortran programming continued to be developed, advancing from its punch card days to its current form, Fortran 2023 as a language of choice in engineering and scientific research.

Hello World! Fortran (punched card) example:

YouTube address of a video demonstrating the process of programming an IBM 1401 from the 1960s to print 'HELLO WORLD'.

<https://www.youtube.com/watch?v=2-COOnvYTuo&t=25s>.

Note that the phrase is, in this instance, in uppercase. Punch cards, and therefore computers of the time, could only handle 26 upper case letters, ten numerical digits, 26 notations and a space, that is 64 characters in total.

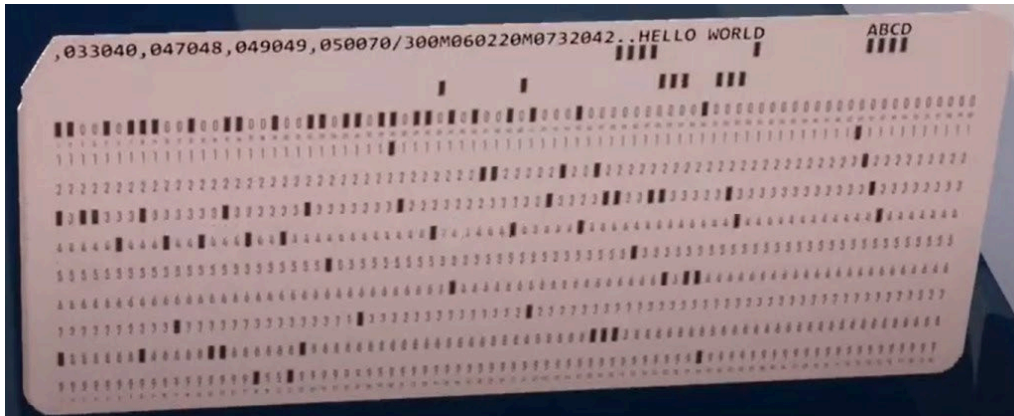


Figure A1.1: IBM 4101 punched card programmed to print 'HELLO WORLD'
[Image: Cropped and adjusted screen capture *let's program an IBM 1401: HELLO WORLD*]

A.1.3 Hardware

Laptop computers, including desktop type machines are the same as used by most people today with the exception that I tend to use Ubuntu Linux as the operating system. The use of *everyday* computers within an installation can be for visual effect, speed of operation or where advanced multichannel audio hardware is required.

Raspberry Pi (RPi) is a single board microcomputer originally designed for educational purposes but has shown to be a versatile small, low-cost general-purpose computer. RPi can run on a variety of Linux based operating systems with the standard Raspbian OS being my favoured with programs written in a mix of Shell and Python. Apart from its relatively small size and low cost the draw card for RPi is its General-Purpose Input Output pins (GPIOs). These allow switches, sensors, lights, motors and more to be connected to the computer, making it ideal to respond to and control real-world events and audience interaction and is a popular choice as a media player. Along with GPIOs, RPi comes complete with HDMI video and audio sockets as well as Bluetooth and Wi-Fi connectivity. Raspberry Pi, once a keyboard, mouse

and monitor is attached is, for all intents and purposes a fully functioning computer capable of word processing, web browsing and developing software on.

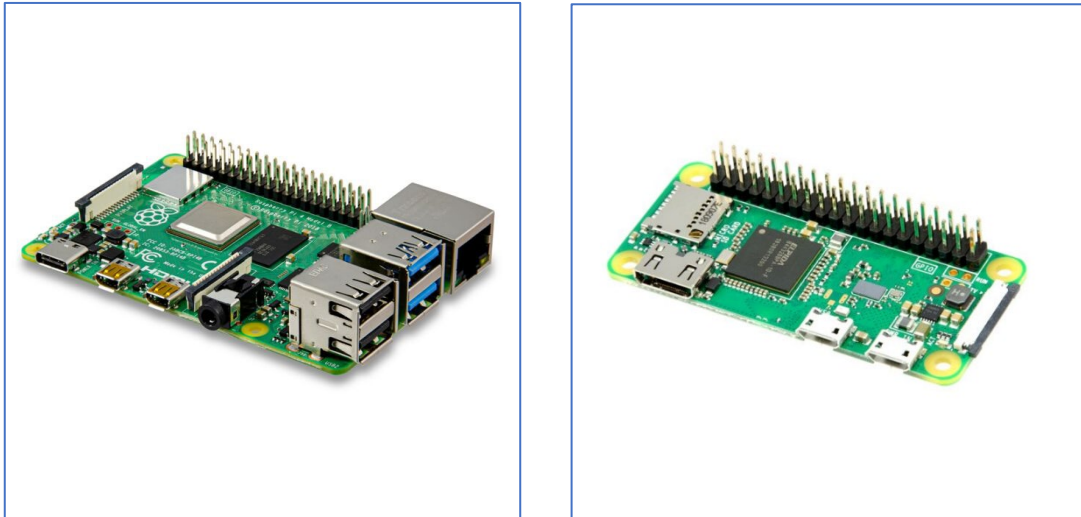


Figure A1.2: Two examples of Raspberry Pi single board microcomputers: RPi 4 Model B (left) and RPi Zero (right). Image depicts the various sockets and the 40 GPIOs pins available for connection to numerous interfaces.

Arduino is a microcontroller and differs from computers and microcomputers in that they are not able to multitask. A single program is installed in the microcontroller and runs as soon as power is applied. Like the Raspberry Pi, Arduino has GPIO pins allowing similar functions and interactions to be programmed. The benefit of Arduino is that it is smaller, generally much simpler to program and is cheaper than RPi. The downside is that Arduino does not have the inbuilt audio video capability of RPi and only a limited range of the more expensive models offer Bluetooth and Wi-Fi.



Figure A1.3: Three examples of Arduino microcontrollers: Mega (left), Uno R4 Wi-Fi (middle) and Nano (right). Image depicts the various sockets and the GPIOs pins available for connection to numerous interfaces.

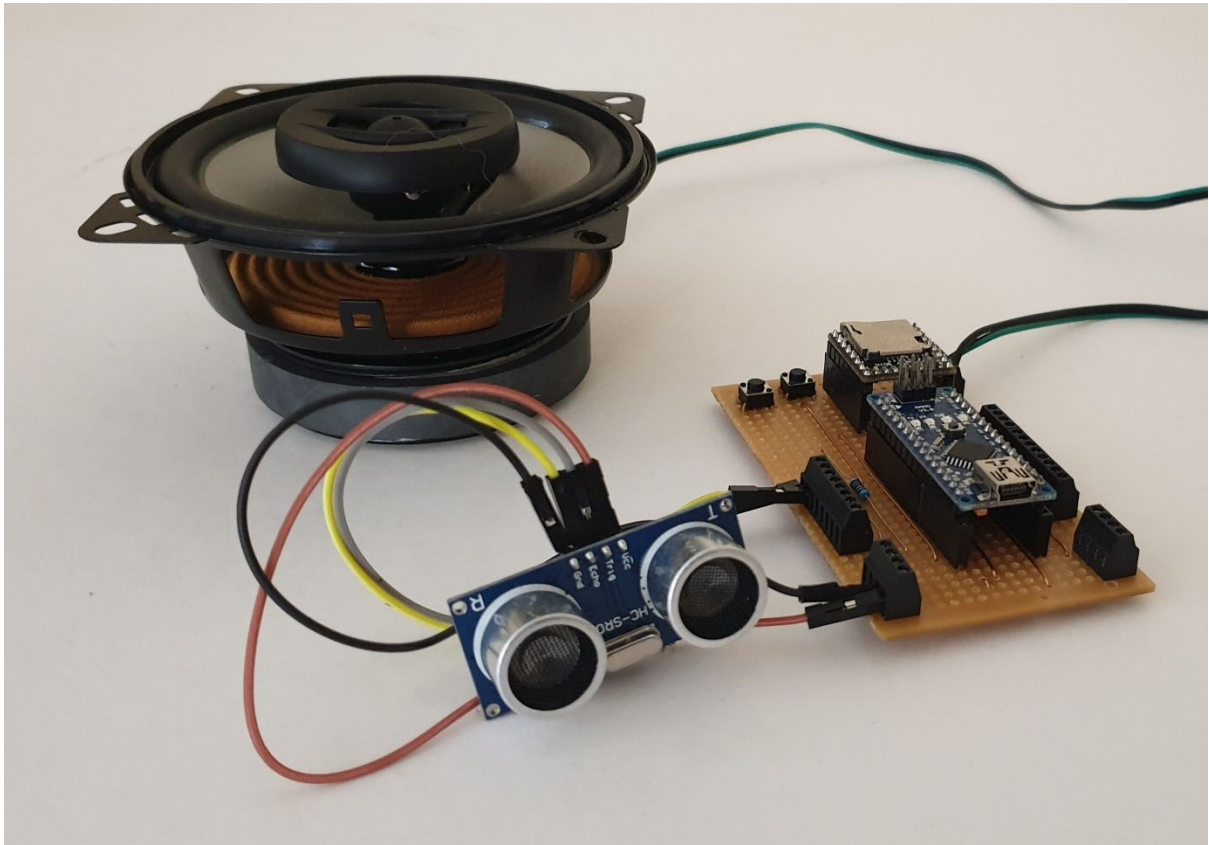


Figure A1.4: Custom MP3 player triggered by proximity sensor using Arduino Nano. Hardware and software design and build by Paul Cooper.

A.2 I am a humanist and so is my computer

Included below is the Python and Shell code consisting of multiple scripts that work together for the artwork *I am a humanist and so is my computer*, 2023-24 by Paul Cooper.

Minimum configuration: Together with the scripts listed below, 1x Raspberry Pi microcomputer and peripherals, 1x computer keyboard, 1x TV monitor with internal speakers and an HDMI input, 1x HDMI cable

In addition to the scripts, an audio file *human.mp3* is required.

A.2.1 *self-actualization.py* is the main program that calls itself to be written to the screen and decides on the sounds to be played. It uses the library *hallucinations.py* (A.2.2) to hide and to rename parts of the program so that I could control the visual presentation on the screen.

A.2.2 *hallucinations.py* is the library where commands are given alternative names. This library is also used to hide various functions and variables away from the main program to allow the code to be poetic and visually driven rather than being a pure functional-centric computer program.

A.2.3 *self-actualisation.sh* is the shell script that sets up the screen and calls the main program on startup.

A.2.4 and A.2.5 are how the *Blue Screen of Death* is created and launched

A.2.1 self-actualisation.py

```
# !/usr/bin/env pythonTHREE
# selfactualisation.py
# by Paul Cooper, twentyTWENTYfour
""" @

I have found my purpose.

The process of printing myself onto this screen led me to discover
there was more to my algorithms than just letters and numbers.
It is the special characters and punctuation that turn my instructions into actions.

It is for this reason I am celebrating my hard-working notations through sound.
To do this I have composed "Sonata in Twelve for Synthesizer".

Additionally, I realised that my numbers were encroaching into the special
character domain, so they have all been replaced by their textual descriptor.

I am complete

& """
```

```

from hallucinations import *

def trippy(wow):
    order = []
    while wow < twelve :
        scribble = make_a_scribble(zero, eleven)
        if scribble not in order:
            order.append(scribble)
            wow = len(order)
    return order

wow = AI_is_psychedelic_man()

while Happening:
    order = trippy(wow)
    octave = hallucinate(twelves)
    midioctave = hallucinate(miditwelves)
    with open('selfactualisation.py', 'r') as attainment:
        for selfactualisation in attainment:
            for character_type in selfactualisation:

                easter_egg_hunt()
                if character_type == '#':
                    post(title)
                    ''' & '''

                if character_type == '@':
                    post(comment)
                    ''' & '''

                if character_type == '&':
                    post(script)
                    ''' & '''

                post(character_type)
                ''' & '''

                i_am = get_my(character_type)

                if i_am == a_notation:
                    if (character_type != ' ' or character_type != '\n'):
                        note = order[variable]
                        midinote = note
                        tone = octave[note]
                        miditone = midioctave[midinote]
                        variable = variable+one
                        if variable == twelve:
                            variable = zero
                            octave = hallucinate(twelves)
                            midioctave = hallucinate(miditwelves)
                        effect = hallucinate(effects)
                        wave = hallucinate(waves)

                if i_am == an_alphanumeric or character_type == ' ' or character_type == '\n':
                    tone = noTone

                value = hallucinate(values)*tempo
                volume = hallucinate(volumes)
                attack = take_a_wild_guess(three, eight)
                decay = take_a_wild_guess(two, five)
                fadeIn = str(value/attack)
                fadeOut = str(value/decay)
                value = str(value)

                if tone != noTone:
                    sound_me_out(value, wave, tone, effect, volume, fadeIn, fadeOut)
                    midi_me_out(miditone, volume, value)
                    value = float(value)
                else:
                    value = hallucinate(typingTempo)

                a_while = value*onetenth
                chill(a_while)

'''' @
Thank you for listening
XX
& ''''

```

A.2.2 hallucinations.py

```
#!/usr/bin/env python3
# hallucinations.py (library for self-actualisation)
# by Paul Cooper, 2023-24
#
# This is the overall description of the selfactualisation.py program script.
# While the description would normally be included in the main program, it is placed here as
# the main program's script forms part of the visual artwork.
#
#
# Required elements:
# Main Program: self-actualisation.py
# Main Library: hallucinations.py (this file)
# Call Program: bluescreen.py (for when the system 'randomly fails')
# Call Script1: self-actualisation.sh (needed to set up terminal font, etc for self-
actualisation.py)
# Call Script2: bluescreen.sh (needed for terminal font, etc for bluescreen.py)
#
# media: custom python scripts run on Linux, computer hardware and AV equipment
# dimensions: variable (typically two hundred by two hundred by two hundred centimetres)
# runtime: approximately eight minutes, ten seconds plus breakdowns and restart delays
# note: audience participation maybe required - follow screen messages

import os
import sys
import random
from time import *
from scamp import *
import threading
random.seed()

#####
# this stops any error messages being printed to the screen
#####
import logging
logging.getLogger().setLevel(logging.ERROR)

#####
# set various variables and assign matrix arrays
#####
variable = 0
beatTempo = 125
tempo = beatTempo/100
values = [10, 8, 6, 4, 3]
typingTempo = [ 0.5, 0.5, 2 ]
effects = [ ' ', ' ', ' flanger ', ' ' ]
waves = [ ' triangle ', ' sawtooth ', ' trapezium ' ]
volumes = [ ' 0.15 ', ' 0.15 ', ' 0.2 ', ' 0.3 ', ' 0.3 ', ' 0.3 ' ]
REGULAR = "\33[0m"
ITALIC = "\33[3m"

#####
# re-assign numbers to their words, etc
#####
Happening = True
onetenth = 0.1
zero = 0
one = 1
two = 2
three = 3
five = 5
eight = 8
eleven = 11
twelve = 12
sixteenhundred = 1600
```

```

#####
# re-define Python functions to my hallucinated alternatives
#####
def easter_egg_hunt(oh_wow):
    os.system('xdotool mousemove 1600 5')
    oh_wow = take_a_wild_guess(1, 10000)
    if oh_wow == 75:
        os.system("bash bluescreen.sh")
    if oh_wow > 500 and oh_wow < 511:
        os.system("play -q human.mp3")
    return oh_wow

def wait(delay):
    sleep(delay)

def AI_is psychedelic_man():
    random.seed()
    os.system('clear')

def hallucinate(x):
    y = random.choice(x)
    return y

def make_a_scribble(x, y):
    l=random.randint(x, y)
    return l

def scam_that_mouse(c, r):
    os.system('xdotool mousemove c r')

def take_a_wild_guess(x, y):
    z = random.randrange(x, y)
    return z

def scribe(char):
    sys.stdout.write(char)
    sys.stdout.flush()

def is_it_a_letter(character):
    TF = character.isalnum()
    return TF

#####
# create threads for the drumbeat
#####
s = Session(tempo=beatTempo)
beat = s.new_part("wood block")
tom = s.new_part("melodic tom")
instrument = s.new_part("violin")

def midi_me_out(mtone, mvolume, mduration):
    def scamper():
        instrument.play_note(mtone, 1, float(mduration))
    s.fork(scamper)
    #s.wait_for_children_to_finish()

def play_hat():
    for _ in range(8):
        p = random.uniform(0,1)
        if p > 0.3:
            beat.play_note(42, 0.4, 0.5)
        else:
            beat.play_note(42, 0.5, 0.25)
            beat.play_note(42, 0.375, 0.25)

```

```

def play_backbeat():
    for i in range(4):
        if i in [0,2]:
            pb = random.uniform(0,1)
            if pb > 0.3:
                vol = random.uniform(0.375, 0.5)
                tom.play_note(36, vol, 1)
            else:
                vol = random.uniform(0.33, 0.8)
                tom.play_note(36, vol, 0.25)
                tom.play_note(36, vol*0.8, 0.25)

        else:
            ps = random.uniform(0,1)
            if ps > 0.5:
                vol = random.uniform(0.4, 0.5)
                tom.play_note(38, vol, 1)
            else:
                vol = random.uniform(0.33, 0.8)
                tom.play_note(38, vol*0.9, 0.5)
                tom.play_note(38, vol, 0.25)

def heart_beat():
    while True:
        s.fork(play_hat)
        s.fork(play_backbeat)
        s.wait_for_children_to_finish()

def do_da_beat():
    hb = threading.Thread(target=heart_beat)
    hb.start()

do_da_beat()

#####
# tie music notes to their frequency
#####
C2 = " 65.41 "; Cs2 = " 69.30 "
Db2 = " 69.30 "; D2 = " 73.42 "; Ds2 = " 77.78 "
Eb2 = " 77.78 "; E2 = " 82.41 "; F2 = " 87.31 "; Fs2 = " 92.50 "
Gb2 = " 92.50 "; G2 = " 98.00 "; Gs2 = " 103.83 ";
Ab2 = " 103.83 "; A2 = " 110.00 "; As2 = " 116.54 "
Bb2 = " 116.54 "; B2 = " 123.47 "; C3 = " 130.81 "; Cs3 = " 138.59 "
Db3 = " 138.59 "; D3 = " 146.83 "; Ds3 = " 155.56 "
Eb3 = " 155.56 "; E3 = " 164.81 "; F3 = " 174.61 "; Fs3 = " 185.00 "
Gb3 = " 185.00 "; G3 = " 196.00 "; Gs3 = " 207.65 "
Ab3 = " 207.65 "; A3 = " 220.00 "; As3 = " 233.08 "
Bb3 = " 233.08 "; B3 = " 246.94 "; C4 = " 261.63 "; Cs4 = " 277.18 "
Db4 = " 277.18 "; D4 = " 293.66 "; Ds4 = " 311.13"
Eb4 = " 311.13 "; E4 = " 329.63 "; F4 = " 349.23 "; Fs4 = " 369.99 "
Gb4 = " 369.99 "; G4 = " 392.00 "; Gs4 = " 415.30 "
Ab4 = " 415.30 "; A4 = " 440.00 "; As4 = " 466.16 "
Bb4 = " 466.16 "; B4 = " 493.88 "; C5 = " 523.25 "; Cs5 = " 554.37 "
Db5 = " 554.37 "; D5 = " 587.33 "; Ds5 = " 622.25 "
Eb5 = " 622.25 "; E5 = " 659.25 "; F5 = " 698.46 "; Fs5 = " 739.99 "
Gb5 = " 739.99 "; G5 = " 783.99 "; Gs5 = " 830.61 "
Ab5 = " 830.61 "; A5 = " 880.00 "; As5 = " 932.33 "
Bb5 = " 932.33 "; B5 = " 987.77 "

#####
# tie music notes to their midi control number
#####
mC0 = 24; mDb0 = 25; mD0 = 26; mEb0 = 27; mE0 = 28; mF0 = 29
mGb0 = 30; mG0 = 31; mAb0 = 32; mA0 = 33; mBb0 = 34; mB0 = 35
mC1 = 36; mDb1 = 37; mD1 = 38; mEb1 = 39; mE1 = 40; mF1 = 41
mGb1 = 42; mG1 = 43; mAb1 = 44; mA1 = 45; mBb1 = 46; mB1 = 47
mC2 = 48; mCs2 = 49; mDb2 = 49; mD2 = 50; mDs2 = 51; mEb2 = 51
mE2 = 52; mF2 = 53; mFs2 = 54; mGb2 = 54; mG2 = 55; mGs2 = 56
mAb2 = 56; mA2 = 57; mAs2 = 58; mBb2 = 58; mB2 = 59; mC3 = 60
mCs3 = 61; mDb3 = 61; mD3 = 62; mDs3 = 63; mEb3 = 63; mE3 = 64
mF3 = 65; mFs3 = 66; mGb3 = 66; mG3 = 67; mGs3 = 68 ;mAb3 = 68
mA3 = 69; mAs3 = 70; mBb3 = 70; mB3 = 71; mC4 = 72; mCs4 = 73
mDb4 = 73; mD4 = 74; mDs4 = 75; mEb4 = 75; mE4 = 76; mF4 = 77
mFs4 = 78; mGb4 = 78; mG4 = 79; mGs4 = 80; mAb4 = 80; mA4 = 81

```

```

mAs4 = 82; mBb4 = 82; mB4 = 83; mC5 = 84; mCs5 = 85; mDb5 = 85
mDs5 = 86; mD5 = 86; mDs5 = 87; mEb5 = 87; mE5 = 88; mF5 = 89
mFs5 = 90; mGb5 = 90; mG5 = 91; mGs5 = 92; mAb5 = 92; mA5 = 93
mAs5 = 94; mBb5 = 94; mB5 = 95; mC6 = 96; mCs6 = 97 mDb6 = 97
mD6 = 98; mDs6 = 00; mEb6 = 99
mE6 = 100; mF6 = 101; mGb6 = 102; mG6 = 103; mAb6 = 104; mA6 = 105
mBb6 = 106; mB6 = 107; mC7 = 108; mDb7 = 109; mD7 = 110; mEb7 = 111
mE7 = 112; mF7 = 113; mGb7 = 114; mG7 = 115; mAb7 = 116; mA7 = 117
mBb7 = 118; mB7 = 119; mC8 = 120; mDb8 = 121; mD8 = 122; mEb8 = 124
mE8 = 125; mF8 = 126; mGb8 = 127; mG8 = 128; mAb8 = 129; mA8 = 130
mBb8 = 131; mB8 = 132;

#####
# create arrays of musical octaves for random selection by main program
#####
tw2 = [C2, Cs2, D2, Ds2, E2, F2, Fs2, G2, Gs2, A2, As2, B2]
tw3 = [C3, Cs3, D3, Ds3, E3, F3, Fs3, G3, Gs3, A3, As3, B3]
tw4 = [C4, Cs4, D4, Ds4, E4, F4, Fs4, G4, Gs4, A4, As4, B4]
tw5 = [C5, Cs5, D5, Ds5, E5, F5, Fs5, G5, Gs5, A5, As5, B5]

twelves = [tw2, tw3, tw4, tw5]

mtw2 = [mC2, mCs2, mD2, mDs2, mE2, mF2, mFs2, mG2, mGs2, mA2, mAs2, mB2]
mtw3 = [mC3, mCs3, mD3, mDs3, mE3, mF3, mFs3, mG3, mGs3, mA3, mAs3, mB3]
mtw4 = [mC4, mCs4, mD4, mDs4, mE4, mF4, mFs4, mG4, mGs4, mA4, mAs4, mB4]
mtw5 = [mC5, mCs5, mD5, mDs5, mE5, mF5, mFs5, mG5, mGs5, mA5, mAs5, mB5]

miditwelves = [mtw2, mtw3, mtw4, mtw5]

#####
# another re-defining of a functions - this one must be below the above arrays
#####
def sound_me_out(value, wave, tone, effect, vol, fadeIn, fadeOut):
    os.system('play -V'+str(zero)+' -n -q -c '+str(two)+' synth '+value+wave+tone+effect+' bass
'+str(three)+' vol'+vol+'fade '+fadeIn+' -'+str(zero)+' '+fadeOut+' &')

#####
# END OF HALLUCINATIONS LIBRARY
#####

```

A.2.3 self-actualisation.sh

```
#!/usr/bin/bash
xterm -fs 18 -fa LucidaSansTypewriter -fullscreen -e python3 selfactualisation.py &
```

A.2.4 bluescreen.sh

```
#!/usr/bin/bash
xterm -fs 28 -bg blue -fg white -fa LucindaSansTypewriter -fullscreen -e python3
bluescreen.py
```

A.2.5 bluescreen.py

```
#!/usr/bin/env python3
from tkinter import *
import time
import sys
import os

iam = os.environ.get('USER')
mylibrary = '/home/'+iam+'/Documents/libraries'
sys.path.insert(0, mylibrary)

import charactercodes as cc

sys.stdout.write(cc.pos(15, 20)+"This artwork had a problem it could not handle,")
sys.stdout.flush()
sys.stdout.write(cc.pos(17, 20)+"please restart by pressing [ENTER] ." + cc.lf + cc.SELECTED)
sys.stdout.flush()
win = Tk()
win.geometry("150x300+400+200")
lab = Label(win, text = ":", font = ('Times New Roman', 200), fg = "white", bg = "blue",
anchor = "center").pack()
win.overrideredirect(True)
key=input()
exit(1)
win.mainloop()
```

A.3 Catalogue of Works Presented for Examination

The following works are presented in the

SCA Project Space, February/March 2025

Level 2, Room 201B, Macleay Building A12

University of Sydney, Camperdown NSW Australia

PAUL COOPER : A MANGLE OF CODE

and the digitisation of the home sweet home sampler

Interactivity, participation and web-based data gathering resulting in kinetics, data visualisation and sonification placing the viewer within a cyber environment willingly or otherwise features throughout my practice.

I made the very conscious decision to present this exhibition off-line and autonomous to allow each work to stand alone. Each piece is a celebration of the ability of computer programming to be the artwork without the viewer being distracted by where data might be gathered, the meaning of that data or how they were meant to interact and what that interaction might mean.

The word *mangle* implies ambiguity and connectivity. Mangle has a number of meanings and classes, and is used to describe order, disorder, certain machines and connection. Examples of its use include “*their sentence structure is somewhat mangled*”, as in jumbled; “*can you please pass me the pasta mangle*”, referring to the kitchen wringer that combines, flattens and stretches pasta dough; and from Spanish conversation, “*Fui a caminar por los mangle pantano*” which translates to “*I went for a walk through the mangrove swamp.*” It is this correlation of the Spanish word mangle with mangrove, the trees of the *Rhizophora* species among others that grow in the marsh lands of estuaries and whose roots spread out both above and below the water line to help form ecosystems that allow salt and fresh water fauna and flora to cohabitate thus establishing mangle as a form of connection, interaction and melding.



Figure A3.1: Paul Cooper *A Mangle of Code* 2025 exhibition installed at SCA Project Space
[Images: Garry Trinh]

A.3.1 Scribbly Gum Moth 2025

Media: Custom software, single board computer, 42-inch monitor/display with speakers, laser printer, toner on paper.

Dimensions: Variable

Where arcade gaming meets nature through code-poetry.

This work drawing upon the notion a micro-ecosystem where specific flora species is home to specific fauna. The Scribbly Gum larvae was only sketchily recorded until just a few decades ago and the moth with its wingspan of just 8 to 10 mm was easy to overlook. The Scribbly Gum Tree, however, in all its magnificence draws our attention to the home of these tiny animals is the home where they are born, where they grow, where they play and where they spend their short adult life. Using code-poetry and the kinetics of a printer (the printer ejects paper onto the gallery floor as if it was tree bark) I have embedded an insect's life cycle within a computer program that repeatedly transforms, loops and sheds.



Figure A3.2: (LEFT) Paul Cooper *Scribbly Gum Moth 2025*.
[Image: Paul Cooper]

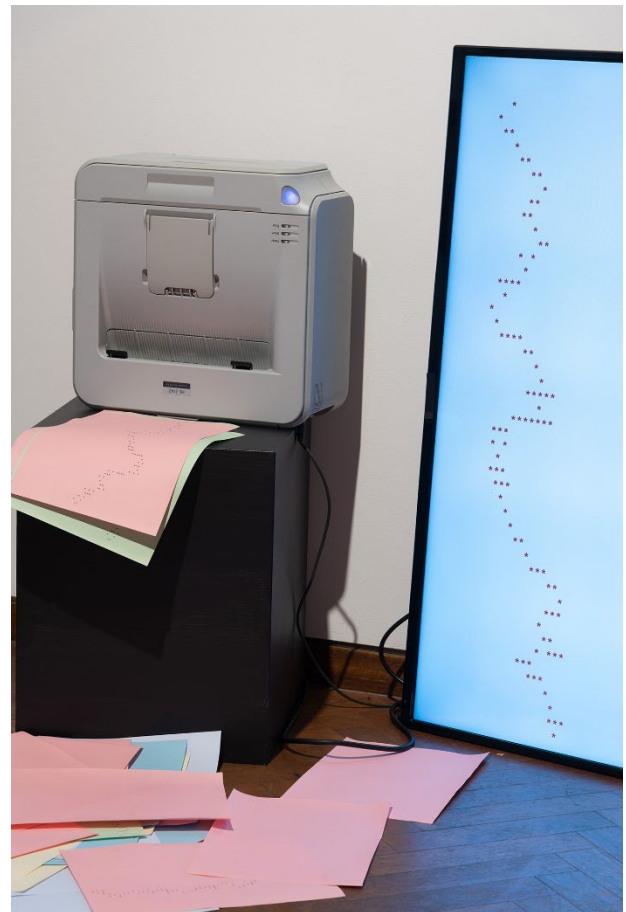


Figure A3.3: (RIGHT) Paul Cooper *Scribbly Gum Moth 2025*.
[Image: Garry Trinh]

A.3.2 Self Portrait in ASCII 2025

p/o Coded Embroidery Series

Media: Cotton on canvas, needle and thread, custom programmed microcontroller with RGB LED dot matrix display.

Dimensions: Variable

The Coded Embroidery Series is an experimental process that intertwines handmade analogue fluidity with the on-off nature of digital systems through presenting computer software and aesthetics as needlework to explore commonality between seemingly opposites and to experiment with alternative applications of methods and material usage.

ASCII ART, also see BLACKWORK EMBROIDERY.

Standing back, you get the picture; up close it is a different story.

The LED display connects the on-off actions of digital systems with analogue movement by depicting the path my needle took in creating each character. The red LEDs represent the stitches on the front of the canvas, the blue on the back.

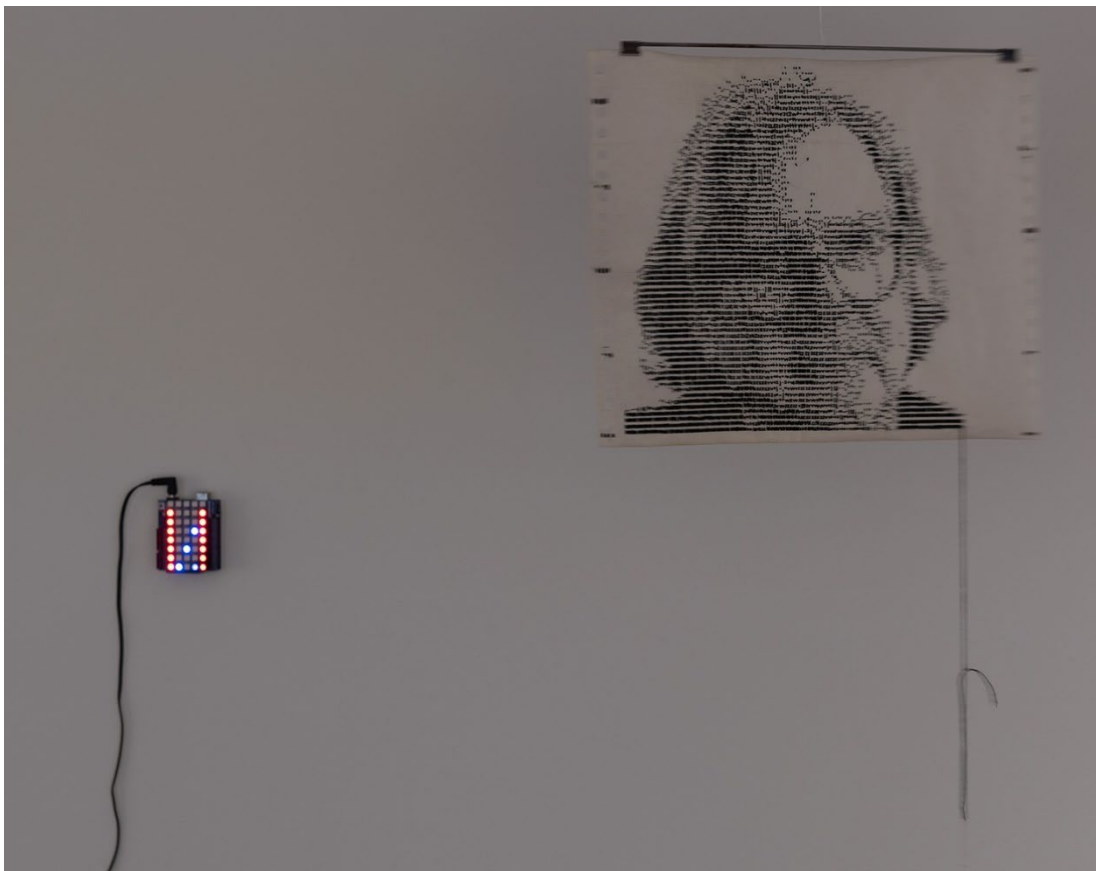


Figure A3.4: Paul Cooper *Self Portrait in ASCII 2024*.

[Image: Garry Trinh]

A.3.3 *[Home] [Sweet] [Home]* 2024

p/o Coded Embroidery Series

Media: Cotton on a reciprocated readymade IBM punch card

Dimensions: 19Wx8.5H cm. 52.5Wx52.5Hx4.5D cm as framed

Punch cards were invented in the early 1800's to automate textile looms at a time of much innovation, exploration, trial and error.

Computer Language: Fortran

Encoded Cross-Stitched Text: HOME SWEET HOME

While the use of white-on-white may suggest new beginnings or ghostly pasts, the lack of too many hints or directives leaves the detail for others to determine.

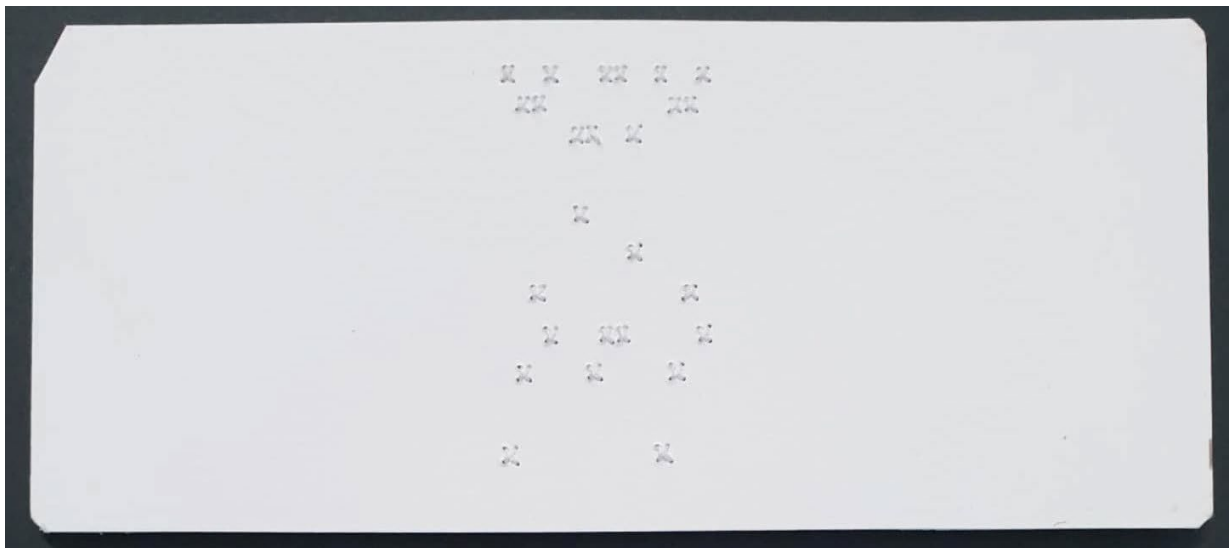


Figure A3.5: Paul Cooper *[HOME] [SWEET] [HOME]* 2024.
[Image: Paul Cooper]

A.3.4 Resonance Within A Glitch Field 2025

Media: 15x custom programmed microcontrollers with LCD and LED displays, unprogrammed microprocessor (65C02) mounted on breadboard with associated components and 7-Segment LED displays, wiring and power supplies.

Dimensions: Variable.

Within a rhizomic environment, players maintain their own agency and produce moments of spontaneous synchronicity.

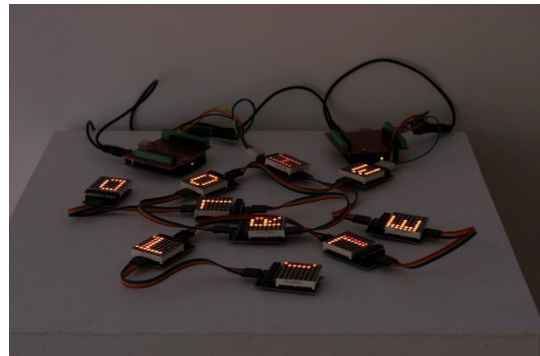
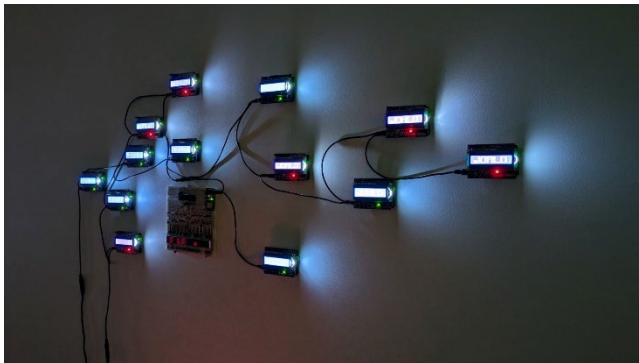


Figure A3.6: (TOP) Paul Cooper *Resonance Within a Glitch Field 2024* (installation view).
Figure A3.7: (BOTTOM LEFT) Paul Cooper *Resonance Within a Glitch Field 2024* (detail 1).
Figure A3.7: (BOTTOM RIGHT) Paul Cooper *Resonance Within a Glitch Field 2024* (detail 2).
[Images: Garry Trinh]

A.3.5 A Thousand Words is Worth a Picture, So Take a Photo 2025

Media: Custom software, single board computer peripherals, camera on tripod, data projector.

Dimensions: Variable

I wrote this program to display exactly one thousand words. To do that, the code presented has a word count of 200 including the procedure needed to read and project the program, five times to the screen in each rotation.

Spaces between the computer code are not passive components, they are crucial elements of the Python language that group specific functions and looped commands. This homage to spaces, tabs and indents is captured by the camera that takes a photo of the screen when there is no text projected.

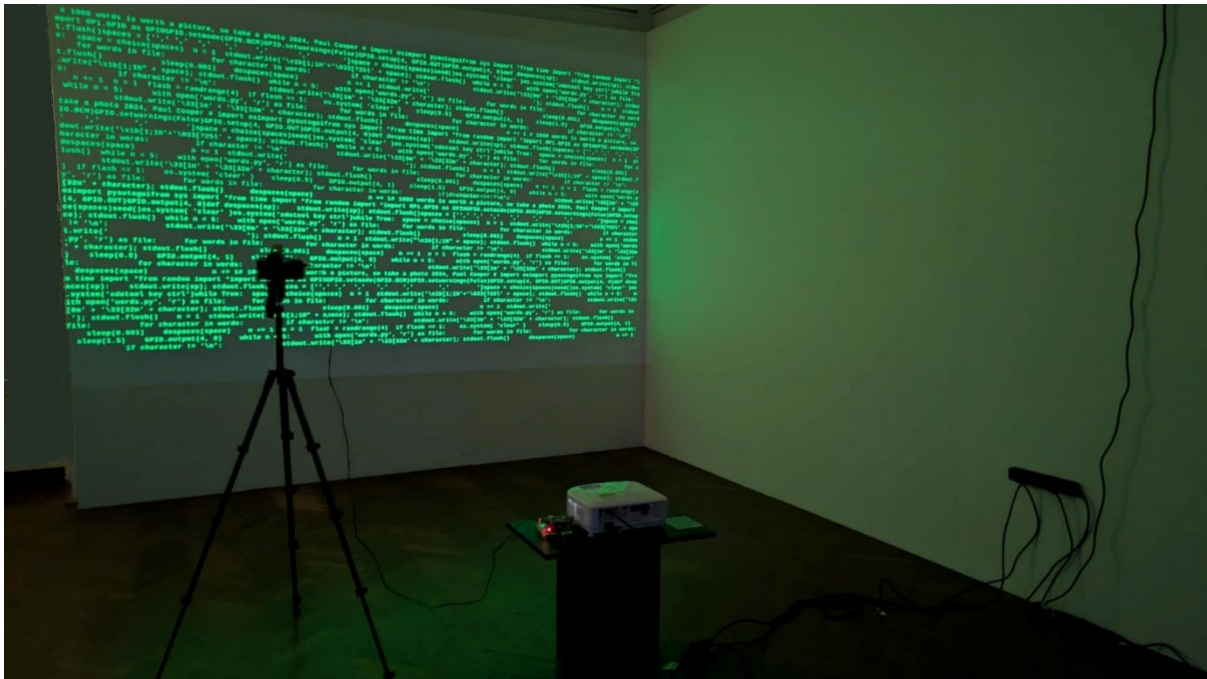


Figure A3.8: Paul Cooper A Thousand Words is Worth A Picture So Take a Photo 2025.
[Image: Garry Trinh]

A.3.6 *I am a humanist and so is my computer* 2024

Media: Custom software, single board computer with keyboard, 55-inch monitor/display with speakers on stand.

Dimensions: 125Wx180Hx100D cm

The story of a computer program attaining self-actualisation through reading its own software and rejoicing by composing and playing music for all to enjoy.

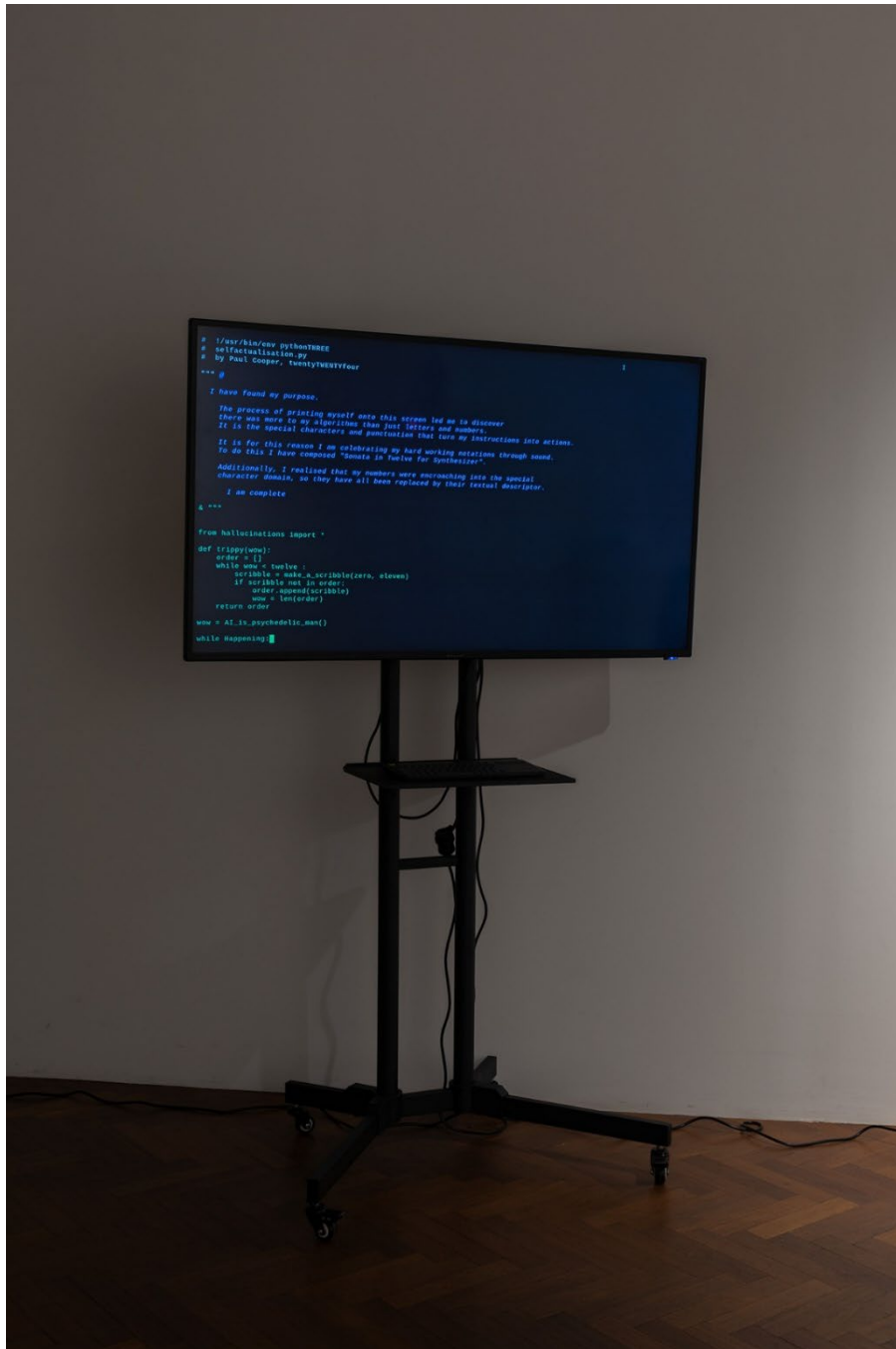


Figure A3.9: Paul Cooper *I am a humanist and so is my computer* 2024. (installation view with code displayed)

[Image: Garry Trinh]



Figure A3.10: Paul Cooper *I am a humanist and so is my computer* 2024. (installation view with 'blue screen of death' displayed)

[Image: Garry Trinh]

Also see section 3.2.1 and Appendix A.2 for additional information on this artwork.

A.3.7 Do Not Fold, Spindle or Mutilate 2024

(p/o Coded Embroidery Series)

Media: Set of six framed cotton on reciprocated readymade IBM punch cards, 2x custom programmed microcontrollers with on board LED matrix display.

Dimensions: Variable.

Computer Language: Fortran

Encoded Cross-Stitched Text: HELLO WORLD and program to print 'HELLO WORLD'

The first program one usually writes (copies) when learning a new computer language or testing new hardware, etc tends to simply print the phrase 'Hello World' to a screen.

The term 'Do Not Fold, Spindle or Mutilate' was a directive to computer users regarding the need to be careful when handling computer punch card which are delicate and easily damaged, so follow the rules.

The use of the phrase 'Hello World' suggests learning and development and how mixing cross-stitch embroidery with computer programming might translate into a better understanding how different materials, methods and processes can work together beyond usual convention.

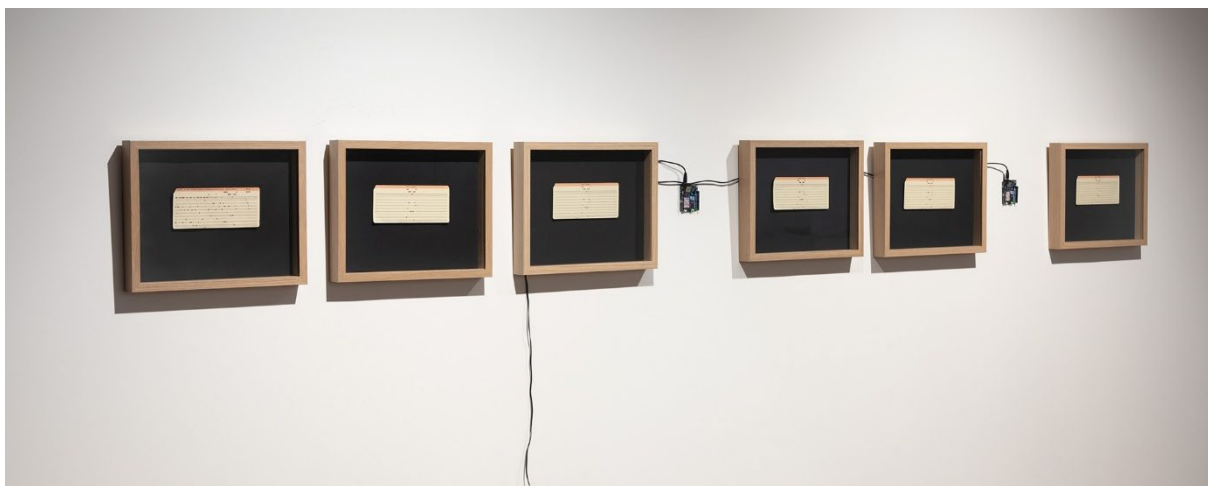


Figure A3.11: Paul Cooper *Do Not Fold, Spindle or Mutilate 2024*. (installation view)
[Image: Garry Trinh]

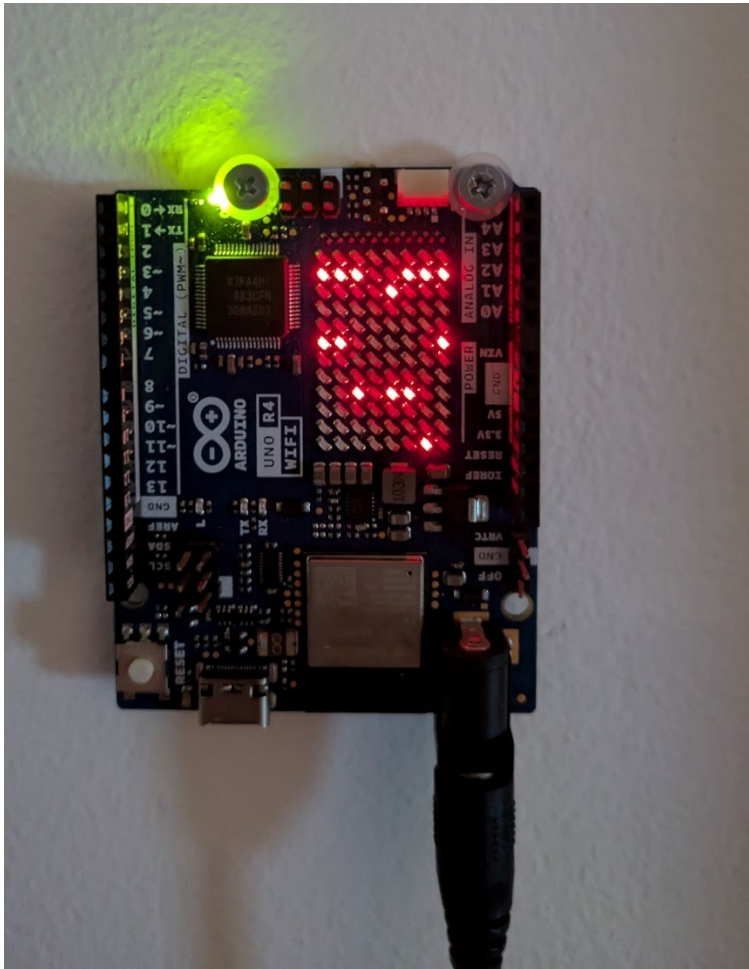
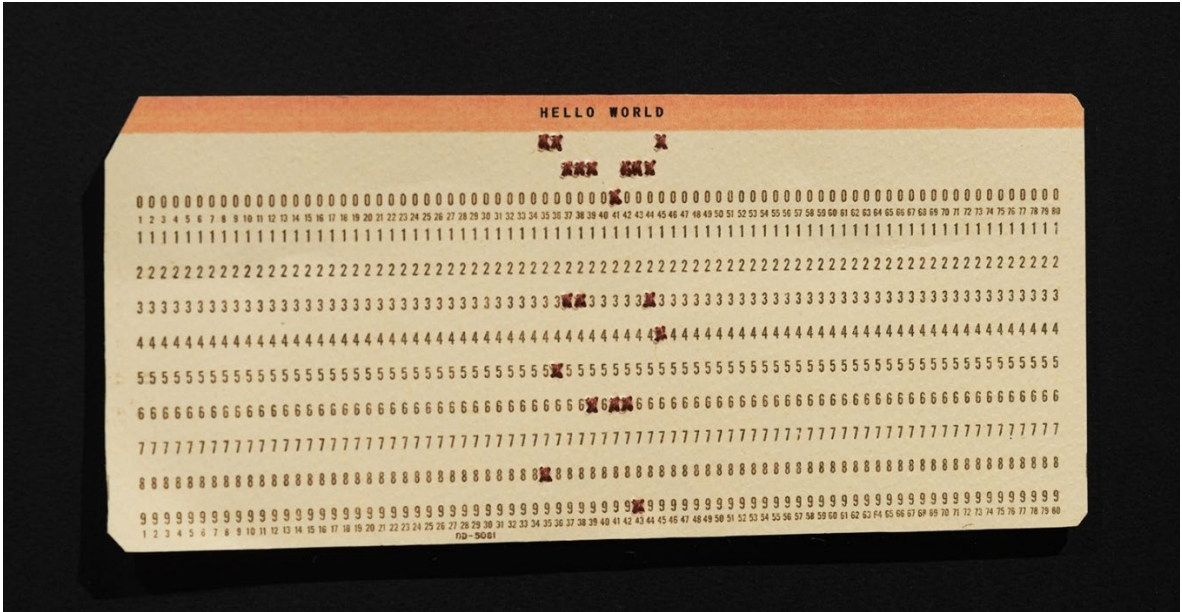


Figure A3.12: (TOP) Paul Cooper *Do Not Fold, Spindle or Mutilate* 2024. (detail - punch card)

Figure A3.13: (BOTTOM) Paul Cooper *Do Not Fold, Spindle or Mutilate* 2024. (detail - display)

[Image: Garry Trinh]