

Przedmowa

Parę lat temu stanąłem przed bardzo interesującym wyzwaniem: miałem nauczyć projektantów technicznych absolutnego minimum elektroniki, aby mogli budować interaktywne prototypy obiektów, które projektują.

Na początku zacząłem, kierując się podświadomym instynktem, uczyć elektroniki w taki sam sposób, w jaki byłem jej uczony w szkole. Później stwierdziłem, że nie przynosi to zamierzonych efektów, i zacząłem przypominać sobie, jak piekielnie znudzony siedziałem w klasie, zalewany całą tą teorią bez wzmianki o jakimkolwiek praktycznym zastosowaniu.

W rzeczywistości, gdy byłem w szkole, znałem już elektronikę w bardzo empiryczny sposób: niewiele teorii, ale mnóstwo przydatnych doświadczeń.

Zacząłem myśleć o tym, jak naprawdę uczyłem się elektroniki:

- ◆ Rozkładałem na części dowolne urządzenia elektroniczne, które wpadały mi w ręce.
- ◆ Powoli poznawałem wszystkie elementy.
- ◆ Zaczynałem z nimi majstrować, zmieniając pewne wewnętrzne połączenia i patrząc, co dzieje się z urządzeniem: zwykle coś między wybuchem a smugą dymu.
- ◆ Zaczynałem budować pewne zestawy z czasopism elektronicznych.
- ◆ Łączyłem rozpracowane urządzenia oraz zestawy i inne obwody, znalezione w czasopismach, zmieniając ich zastosowanie, aby powstawały nowe rzeczy.

Jako małe dziecko, zawsze byłem zafascynowany odkrywaniem, jak rzeczy działają. Dlatego miałem zwyczaj rozbierania ich na części. Ta pasja rozwijała się, a celem stawały się różne nieużywane sprzęty domowe, które następnie rozkładałem na najmniejsze kawałki. W końcu ludzie zaczęli przynosić mi wszelkiego rodzaju urządzenia do rozmontowania. Moim największym

projektem w tym czasie były zmywarka i wczesny komputer, który pochodził z biura ubezpieczeniowego. Ten komputer miał dużą drukarkę, karty elektroniczne, magnetyczne czytniki kart i wiele innych rzeczy, które oczywiście były bardzo interesujące, a ich całkowite rozłożenie było wspaniałym wyzwaniem.

Po rozmontowaniu wielu urządzeń znałem elementy elektroniczne i z grubsza wiedziałem, do czego służą. Na dobitkę mój dom był pełen starych czasopism elektronicznych, które mój ojciec kupił zapewne na początku lat siedemdziesiątych. Spędziłem wiele godzin, czytając te artykuły i przyglądając się schematom obwodów – bez większego zrozumienia.

Czytanie w kółko artykułów wraz ze wzrostem wiedzy zdobywanej dzięki rozkładaniu sprzętu na części tworzyły powoli efekt spirali.

Wspaniały przełom nastąpił w pewne Boże Narodzenie, gdy mój ojciec dał mi zestaw pozwalający nastolatkom uczyć się na temat elektroniki. Każdy element był umieszczony w plastikowej kostce, która magnetycznie łączyła się z innymi kostkami, ustanawiając połączenie. Na wierzchu był umieszczony symbol elektroniczny. Nie zdawałem sobie wtedy sprawy, że ta zabawka, która powstała w latach sześćdziesiątych, była także charakterystyczna dla niemieckiego stylu projektowania, ponieważ jej projektantem był Dieter Rams.

Dzięki temu nowemu narzędziu mogłem szybko składać obwody i wypróbowywać ich działanie. Cykl prototypowania stawał się coraz krótszy.

Następnie budowałem radia, wzmacniacze, obwody wydające straszne hałasy i przyjemne dźwięki, czujniki deszczu i małe roboty.

Szukałem długo słowa, które podsumowuje sposób pracy bez specyficznego planu, zaczynający się od jednego pomysłu, a kończący się na całkowicie nieoczekiwanym wyniku. W końcu nazwałem to „majstrowaniem” (ang. tinkering). Odkryłem, że to słowo jest używane w wielu innych dziedzinach do opisu sposobu działania i określenia osób, które wytyczają drogę rozwoju. Na przykład pokolenie francuskich reżyserów, którzy powołali do życia „Nową falę”, było nazywane mianem „tinkerers”. Najlepsza definicja majstrowania, jaką kiedykolwiek spotkałem, pochodzi z wystawy w Exploratorium w San Francisco:

Majstrowanie polega na próbowaniu czegoś, gdy nie do końca wiemy, co chcemy osiągnąć. Kierunek wskazuje humor, wyobraźnia i zaciekawienie. Podczas majstrowania nie ma żadnych instrukcji – ale nie ma też błędów ani dobrych czy złych sposobów wykonania. Jest to rozpoznawanie, jak rzeczy działają i przerabianie ich.

Przyrządy, maszyny, niedopasowane obiekty działające w harmonii – to istota majstrowania.

Majstrowanie to zasadniczo proces łączący zabawę i dociekanie.

—www.exploratorium.edu/tinkering

Dzięki moim wczesnym eksperymentom wiedziałem, ile doświadczenia wystarczy do zbudowania obwodu działającego zgodnie z oczekiwaniami, gdy zaczynamy od podstawowych elementów.

Inny przełom przyszedł latem 1982 r., kiedy przyjechałem do Londynu z moimi rodzicami i spędziłem wiele czasu zwiedzając Muzeum Nauki. Właśnie otworzyli wtedy nowy dział komputerowy, gdzie dzięki seriom ćwiczeń z instruktorem nauczyłem się podstaw matematyki binarnej i programowania.

Wtedy odkryłem, że w wielu zastosowaniach inżynierowie nie muszą już budować obwodów z podstawowych elementów, ale zamiast tego mogą implementować znaczną część logiki przy użyciu mikroprocesorów. Oprogramowanie zastąpiło wiele godzin projektowania elektronicznego i pozwoliło skrócić cykl majstrowania.

Gdy wróciłem, zacząłem oszczędzać pieniądze, ponieważ chciałem kupić komputer i nauczyć się programować.

Moim pierwszym i najważniejszym projektem po tym wydarzeniu było użycie mojego fabrycznie nowego komputera ZX81 do sterowania spawarką. Wiem, że nie brzmi to jak bardzo ekscytujący projekt, ale był potrzebny i stanowił dla mnie wspaniałe wyzwanie, ponieważ właśnie nauczyłem się programować. W tym momencie stało się jasne, że pisanie wierszy kodu zabiera mniej czasu niż modyfikowanie złożonych obwodów.

Dwadzieścia parę lat później zacząłem myśleć, że to doświadczenie pozwala mi uczyć osoby, które niewiele pamiętają z lekcji matematyki, i zarazić je tym samym entuzjazmem i zdolnością do majstrowania, które miałem w młodości i zachowałem do tej pory.

—Massimo

1 Wstęp

Arduino to platforma programowania urządzeń o otwartych źródłach (open source), oparta na prostej płytce z wejściami i wyjściami (I/O) oraz środowisku projektowym, które implementuje język Processing (www.processing.org). Platforma Arduino może służyć do tworzenia samodzielnych obiektów interaktywnych lub może być połączona z oprogramowaniem na komputerze (takim jak Flash, Processing, VVVV lub Max/MSP). Płytki można złożyć ręcznie lub zakupić fabrycznie złożone, zaś otwarte środowisko programistyczne IDE (Integrated Development Environment) jest dostępne do pobrania za darmo z witryny www.arduino.cc.

Arduino różni się od innych platform na rynku, ponieważ ma następujące cechy:

- ◆ Jest to środowisko dla wielu systemów operacyjnych – Windows, Macintosh i Linux.
- ◆ Jest oparte na środowisku programistycznym Processing, łatwym do użycia środowisku projektowym IDE używanym przez artystów i projektantów.
- ◆ Arduino programujemy przez kabel USB, a nie port szeregowy. Ta cecha jest przydatna, ponieważ wiele nowoczesnych komputerów nie ma portów szeregowych.
- ◆ Sprzęt i oprogramowanie są otwarte (open source) – jeżeli chcemy, możemy pobrać schemat obwodu, kupić wszystkie elementy i wykonać własną płytkę, bez płacenia ani grosza twórcom Arduino.
- ◆ Sprzęt jest tani. Płytkę USB kosztuje około 20 euro (obecnie około 35 dolarów), a wymiana spalonego układu scalonego na płytce jest prosta i nie kosztuje więcej niż 5 euro. Dlatego możemy pozwolić sobie na błędy.

- ◆ Istnieje aktywna społeczność użytkowników, więc jest mnóstwo osób, które chętnie nam pomogą.
- ◆ Projekt Arduino powstał w środowisku edukacyjnym i właśnie dlatego jest wspierany dla osób początkujących, ponieważ pozwala im szybko uzyskać działające rozwiązania.

Niniejsza książka ma za zadanie pomóc początkującym poznać korzyści, jakie odniosą z nauki korzystania z platformy Arduino i działania zgodnie z jej koncepcją.

Grupa docelowa

Ta książka została napisana dla „pierwotnych” użytkowników Arduino: projektantów i artystów. Dlatego sposób wytłumaczenia pewnych rzeczy może doprowadzić niektórych inżynierów do szału. Faktycznie jeden z nich nazwał wstępne rozdziały mojego pierwszego brudnopisu „banałami”. To oddaje istotę rzeczy. Zmierzymy się z tym: większość inżynierów nie potrafi wyjaśnić, co robią, innym inżynierom, a co dopiero zwykłym ludziom. Dlatego teraz zagłębimy się w banały.

UWAGA: Arduino powstało na podstawie pracy dyplomowej Hernanda Barragana na platformie Wiring w szkole IDII w Ivrea. Pracą oprócz mnie kierował Casey Reas.

Po tym, jak Arduino zaczęło stawać się popularne, zauważyłem, jak wielu eksperymentatorów, hobbystów i majsterkowiczów różnego rodzaju zaczęło używać tej platformy do tworzenia pięknych i zwariowanych obiektów. Zauważyłem, że wszyscy oni są sami w sobie artystami i projektantami, więc ta książka jest również dla nich.

Arduino powstało, aby można było nauczać projektowania interaktywnego, dyscypliny projektowej, która umieszcza prototypowanie w centrum swojej metodologii. Istnieje wiele definicji projektowania interaktywnego, ale najbardziej podoba mi się następująca:

**PROJEKTOWANIE INTERAKTYWNE TO PROJEKTOWANIE
DOWOLNYCH DOŚWIADCZEŃ INTERAKTYWNYCH.**

W dzisiejszym świecie projektowanie interaktywne skupia się na tworzeniu znaczących doświadczeń między nami (ludźmi) i obiektami. Jest dobrą drogą, która pozwala zagłębić się w tworzenie pięknych – a może nawet kontrowersyjnych – doświadczeń między nami a technologią. Projektowanie interaktywne stymuluje proces tworzenia dzięki interaktywnemu procesowi opartemu na prototypach o coraz większej dokładności. To podejście – również stanowiące część pewnego rodzaju „konwencjonalnego” projektowania – może być rozszerzone, aby obejmowało prototypowanie przy użyciu technologii. W szczególności – prototypowanie z wykorzystaniem elektroniki.

Szczególnym polem projektowania interaktywnego, dotyczącym Arduino, jest programowanie urządzeń (lub projektowanie interakcji z urządzeniami).

Czym jest programowanie urządzeń?

Programowanie urządzeń korzysta z elektroniki do prototypowania nowych materiałów dla projektantów i artystów.

Obejmuje projektowanie interaktywnych obiektów, które mogą się komunikować z ludźmi przy użyciu czujników i elementów aktywnych sterowanych oprogramowaniem działającym w mikrokontrolerze (małym komputerze zawartym w jednym układzie scalonym).

W przeszłości korzystanie z elektroniki oznaczało konieczność nieustannej współpracy z inżynierami oraz tworzenie obwodów po jednym małym elemencie na raz. Te problemy powodowały, że kreatywne osoby trzymały się z dala od bezpośredniej zabawy z medium. Większość narzędzi była przeznaczona dla inżynierów i wymagała dużej wiedzy. W ostatnich latach mikrokontrolery stały się tańsze i łatwiejsze w użyciu, co pozwala na tworzenie lepszych narzędzi.

Dzięki Arduino dokonaliśmy postępu, który polegał na przybliżeniu tych narzędzi do nowicjuszy. Teraz tworzenie własnych projektów jest możliwe po zaledwie dwóch czy trzech dniach warsztatów.

Korzystając z Arduino, projektant lub artysta może bardzo szybko poznać podstawy elektroniki i czujników, a następnie zacząć budować prototypy przy bardzo małym nakładzie finansowym.

2 Droga Arduino

Koncepcja Arduino jest oparta na tworzeniu projektów, zamiast ich omawiania. Jest to nieustanne szukanie szybszych i potężniejszych sposobów budowania lepszych prototypów. Zbadaliśmy wiele technik prototypowania i wynaleźliśmy sposoby „myślenia przy użyciu rąk”.

Klasyczna inżynieria polega na prostym procesie przechodzenia z A do B. Droga Arduino zachwyca możliwością zagubienia się na trasie i znalezienia zamiast tego C.

Jest to proces majstrowania, który tak lubimy – zabawa ze środkami w otwarty sposób i znajdowanie nieoczekiwanego. W tym szukaniu sposobów budowania lepszych prototypów wybraliśmy także pewną liczbę pakietów oprogramowania, która umożliwia proces stałego zmieniania środków programistycznych i sprzętowych.

W następnych punktach zaprezentujemy pewne koncepcje, zdarzenia i osoby, które zainspirowały drogę Arduino.

Prototypowanie

Prototypowanie to najważniejsza koncepcja drogi Arduino: wykonujemy rzeczy i budujemy obiekty, które oddziałują z innymi obiektami, osobami i sieciami. Staramy się znaleźć prosty i szybki sposób prototypowania, a równocześnie możliwie najtańszy.

Wiele osób początkujących podchodzących do elektroniki po raz pierwszy myśli, że będzie się uczyć, jak budować wszystko od podstaw. Jest to strata energii: chcemy bardzo szybko sprawdzić, że coś działa, aby zmotywować się do podjęcia następnego kroku, a nawet zmotywować kogoś innego, aby nam za to zapłacił.

Dlatego wynaleźliśmy „prototypowanie oportunistyczne”: po co tracić czas i energię na budowanie od podstaw, proces wymagający czasu i głębokiej

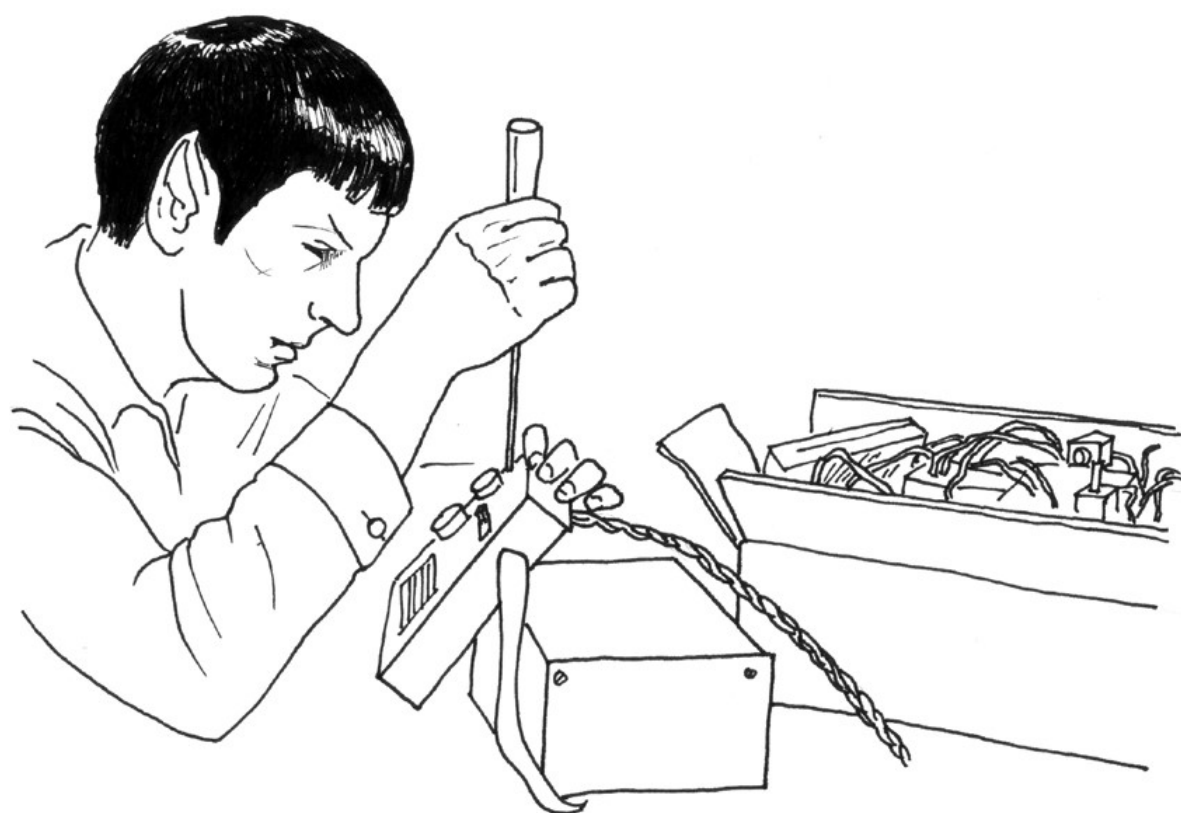
wiedzy technicznej, jeżeli możemy wziąć gotowe urządzenia i rozpracować je, aby zbadać ciężką pracę wykonaną przez duże firmy i dobrych inżynierów?

Naszym bohaterem jest James Dyson, który wykonał 5127 prototypów odkurzacza, zanim uznał, że otrzymał satysfakcjonujący projekt (www.international.dyson.com/jd/1947.asp).

Majstrowanie

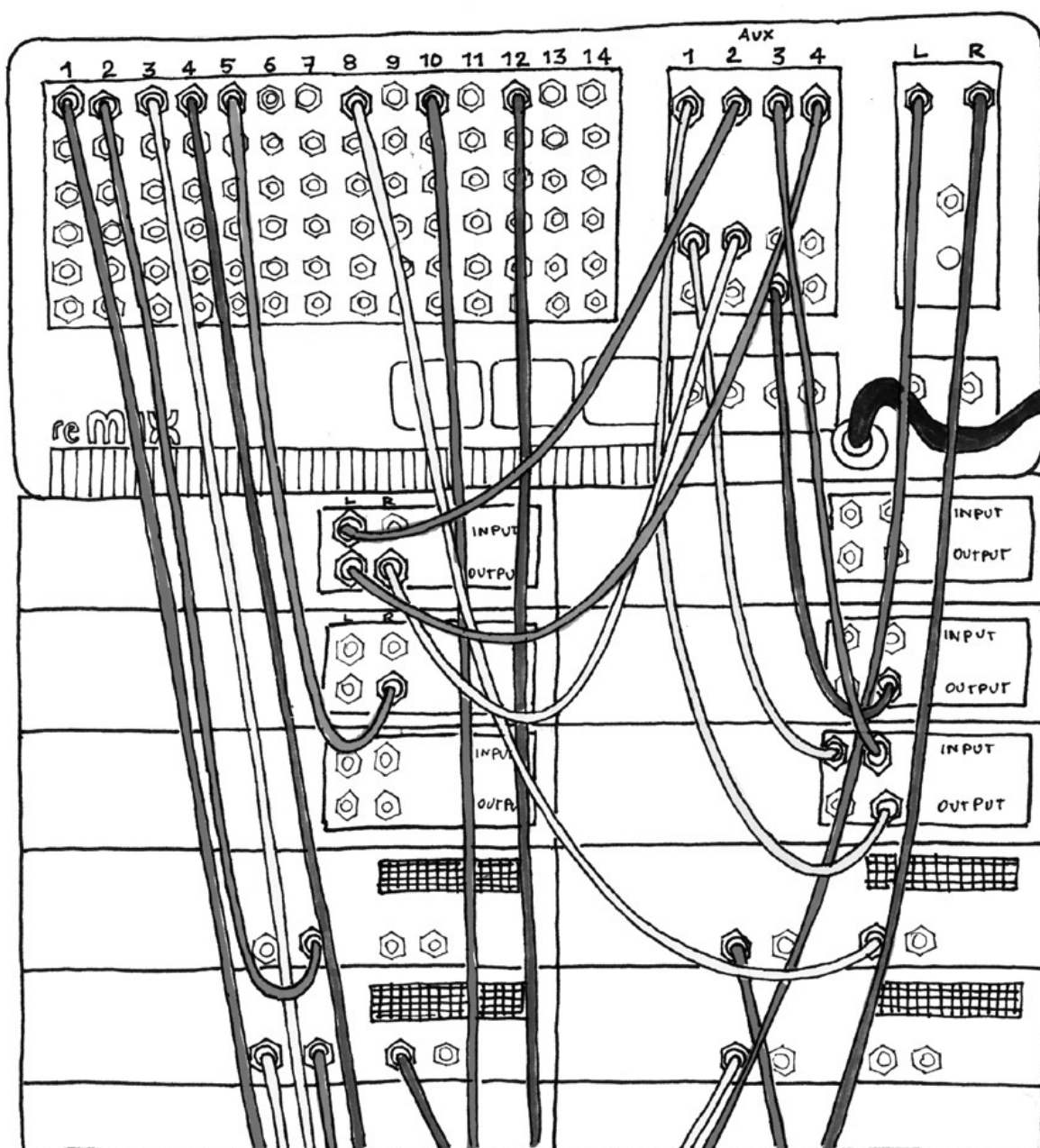
Wierzmy, że jest ono istotą działania z technologią, badania różnych możliwości bezpośrednio sprzętowych i programistycznych – czasami bez dobrze zdefiniowanego celu.

Ponowne używanie istniejącej technologii jest jednym z najlepszych sposobów majstrowania. Rozpracowywanie tanich zabawek lub starego, wyrzuczonego sprzętu i używanie ich do czegoś innego jest jednym z najlepszych sposobów osiągnięcia wspaniałych wyników.



Spinanie

Zawsze fascynowała mnie modularność i zdolność do budowania złożonych systemów za pomocą łączenia ze sobą prostych urządzeń. Dobrym przykładem tego procesu są syntezatory analogowe Roberta Mooga. Muzycy tworzą dźwięki, wypróbowując niekończące się kombinacje przez „spinanie ze sobą” różnych modułów przy użyciu kabli. To podejście sprawia, że syntetyzator wygląda jak stara centrala telefoniczna, ale w połączeniu z licznymi gałkami stanowi doskonałą platformę do majstrowania dźwiękiem i tworzenia innowacyjnej muzyki. Moog opisał to jako proces między „wsluchiowaniem się

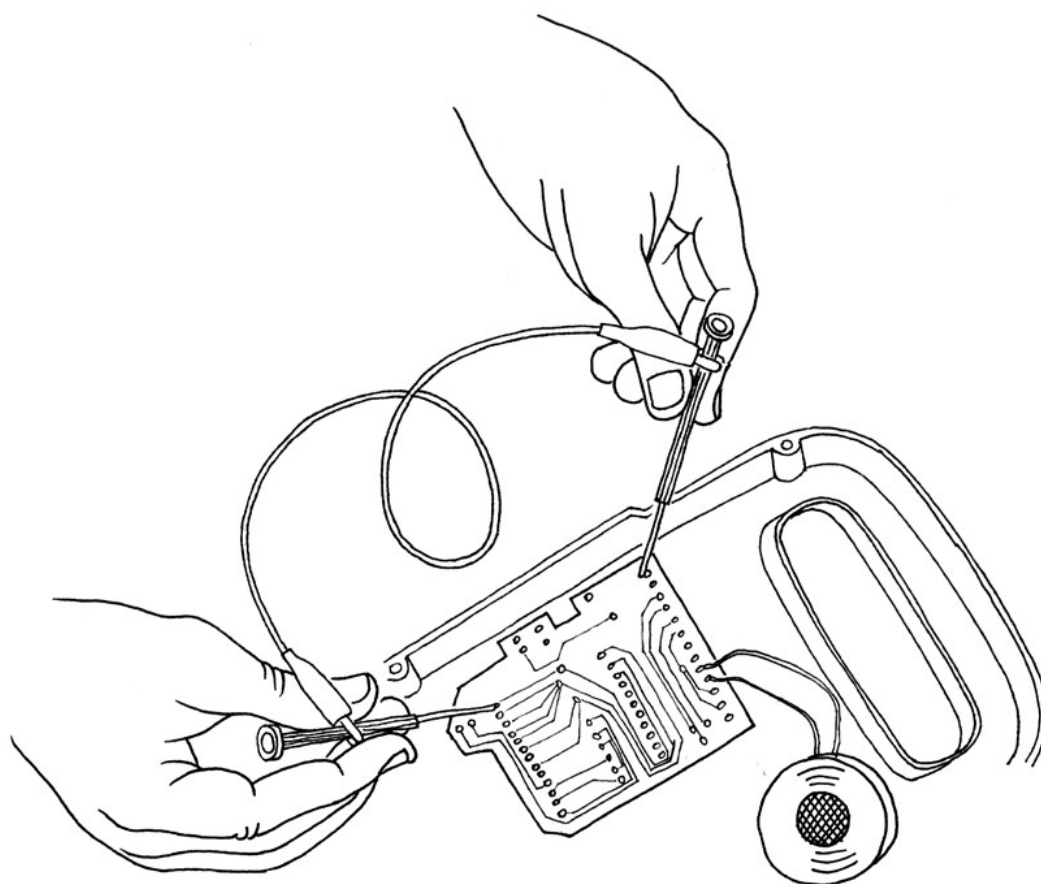


a odkrywaniem”. Jestem pewien, że większość muzyków na początku nie wie, do czego służą te setki galek, ale próbują i próbują, nieustannie doskonaląc własny styl bez tracenia płynności.

Redukcja liczby przerw jest bardzo ważna dla kreatywności – im bardziej płynny jest proces, tym lepiej się majstruje.

Ta technika została przeniesiona w świat oprogramowania dzięki środowiskom „programowania wizualnego”, takim jak Max, Pure Data czy VVVV. Te narzędzia mogą być wizualizowane jako „opakowania” dla różnych funkcjonalności, które dostarczają, pozwalając użytkownikowi „spinać” te opakowania ze sobą. Te środowiska pozwalają użytkownikowi eksperymentować z programowaniem bez ciągłego przerywania typowego dla zwykłego cyklu: „napisz program, skompiluj, do licha—wystąpił błąd, napraw błąd, skompiluj, uruchom”. Zalecam wypróbowanie tych środowisk osobom myślącym wizualnie.

Zwieranie obwodów



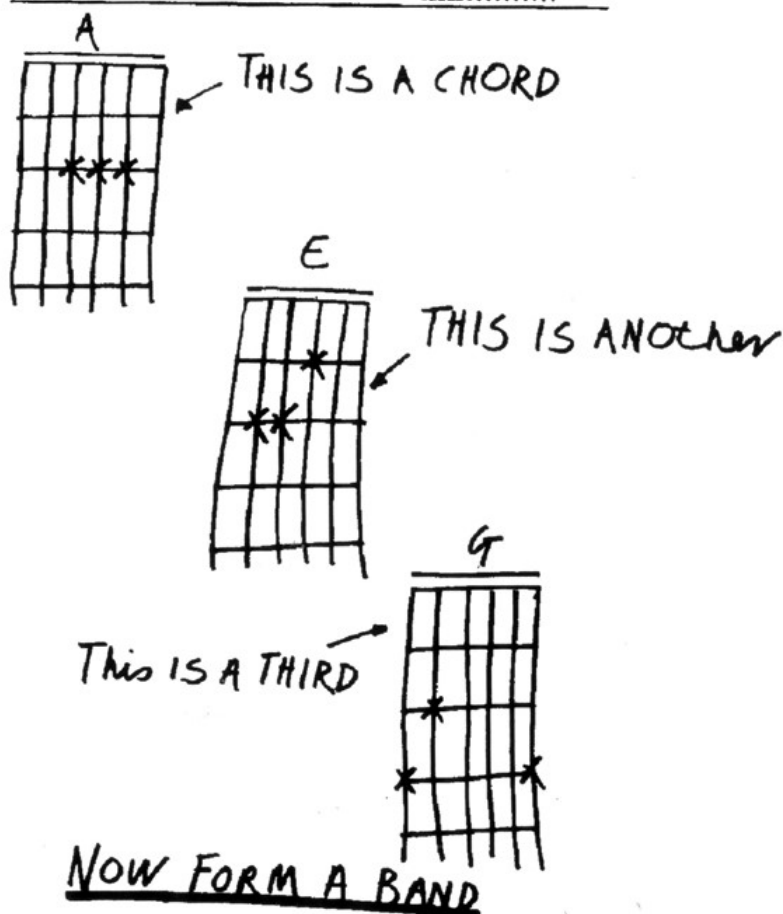
Łączenie obwodów jest jedną z najbardziej interesujących form majstrowania. Jest to kreatywne zwieranie elektronicznych urządzeń audio pracujących pod niskim napięciem i zasilanych z baterii, takich jak pedały efektów gitarowych, dziecięce zabawki i małe syntezatory, aby utworzyć nowe instrumenty muzyczne i generatory dźwięku. Istotą tego procesu jest „sztuka przypadku”. Zaczęło się to w roku 1966, gdy Reed Ghazala przez przypadek zważył zabawkowy wzmacniacz z metalowym obiektem w szufladzie swojego biurka, otrzymując w efekcie strumień niezwykle dźwięków. Tym, co mi się najbardziej podoba w osobach zajmujących się łączeniem obwodów, jest ich zdolność tworzenia najdziwniejszych urządzeń przez majstrowanie z technologią, bez istotnego rozumienia swojego działania od strony teoretycznej.

SNIFFIN' GLUE.. + OTHER ROCK 'N' ROLL HABITS FOR PUNKS! ①

NO.1 OF MANY, WE HOPES!

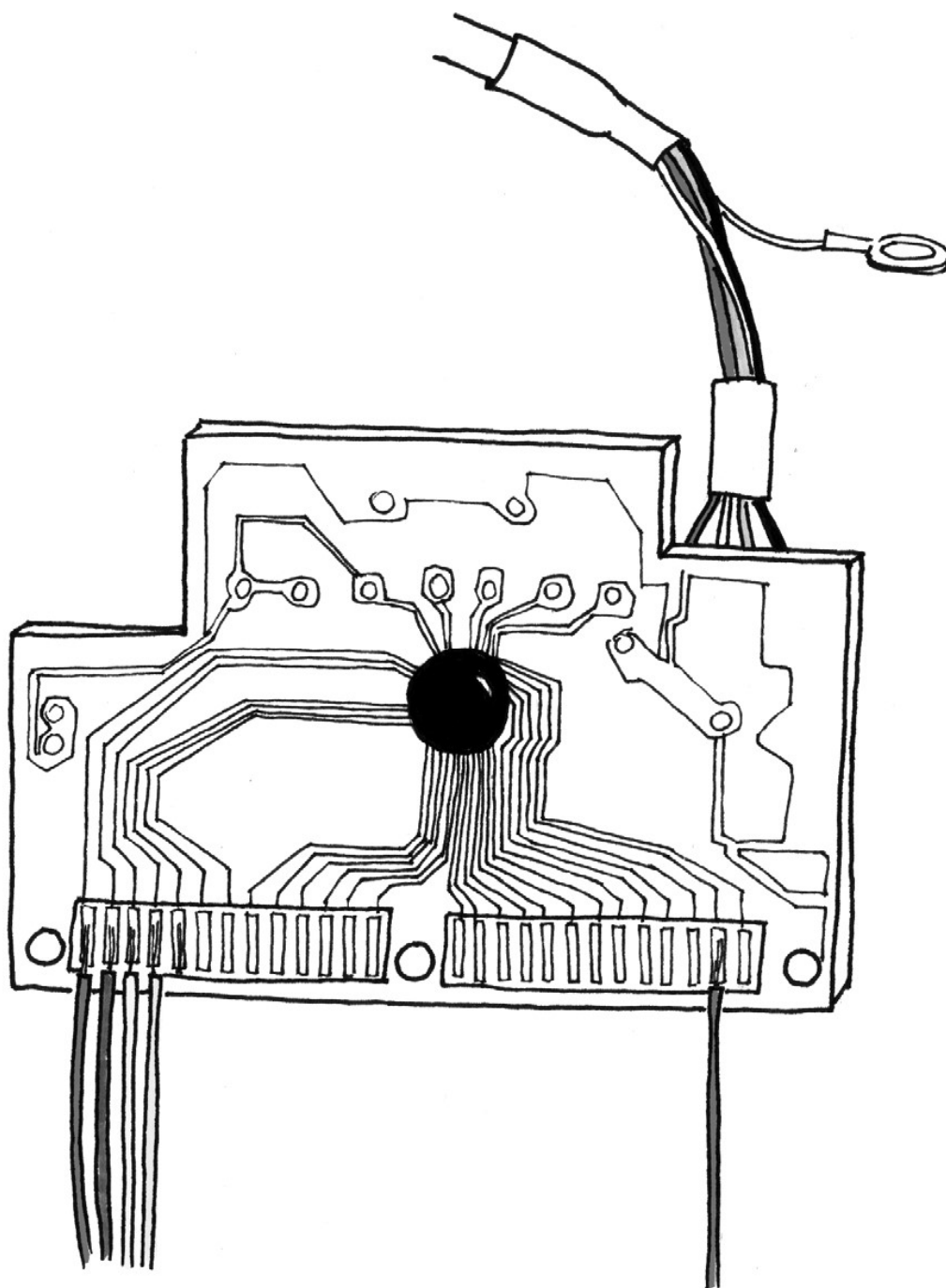
THIS THING IS NOT MEANT TO BE READ...IT'S FOR SOAKING IN GLUE AND SNIFFIN'.

PLAY'IN IN THE BAND...FIRST AND LAST IN A SERIES.....



To odrobinę jak w fanzinie *Sniffin' Glue* pokazanym na poprzedniej stronie: w erze punka znajomość trzech akordów gitarowych wystarczała do założenia zespołu. Nie pozwól ekspertom w jakiejś dziedzinie powiedzieć, że nigdy nie będziesz jednym z nich. Zignoruj to i zaskocz ich.

Rozpracowywanie klawiatury



Klawiatury komputerowe są nadal, po ponad 60 latach, głównym sposobem interakcji z komputerem. Alex Pentland, kierownik naukowy laboratorium Media Laboratory w MIT, napomknął kiedyś: „Wybaczcie porównanie, ale pisuary są mądrzejsze od komputerów. Komputery są izolowane od tego, co jest w ich otoczeniu.”¹

Jako majsterkowicze możemy zaimplementować nowe sposoby oddziaływania z oprogramowaniem przez zastąpienie klawiszy urządzeniami, które są wrażliwe na otoczenie. Rozłożenie na części klawiatury komputerowej odsłania nam bardzo proste (i tanie) urządzenie. Jej najważniejszym elementem jest mała płytką. Zazwyczaj jest to zgnięty lub brązowy obwód z dwoma zestawami styków na dwóch plastikowych warstwach, które przytrzymują połączenia między różnymi klawiszami. Jeżeli usuniemy obwód i użyjemy przewodu, aby połączyć dwa styki, zobaczymy literę pojawiającą się na ekranie komputera. Jeżeli pójdziemy dalej, kupimy czujnik ruchu i połączymy go z klawiaturą, zobaczymy, że klawisz jest wciskany za każdym razem, gdy ktoś przechodzi przed komputerem. Połączmy to z ulubionym oprogramowaniem, a komputer stanie się tak mądry jak pisuar. Nauka rozpracowywania klawiatury jest istotnym składnikiem prototypowania i programowania sprzętu.

Uwielbiamy śmieci!

Obecnie ludzie wyrzucają mnóstwo elementów technicznych: stare drukarki, komputery, dziwne urządzenia biurowe, sprzęt techniczny, a nawet mnóstwo militariów. Zawsze istniał duży rynek obejmujący tę nadwyżkę technologii, szczególnie wśród młodych i/lub biednych majsterkowiczów oraz tych, którzy dopiero zaczynają. Ten rynek stał się oczywisty w Ivrea, gdzie wynaleźliśmy Arduino. To miasto było siedzibą firmy Olivetti. Firma ta produkowała komputery od lat 60 XX w. W środku lat dziewięćdziesiątych wyrzucali wszystko na okoliczne złomowiska. Były one pełne części komputerowych, elementów elektronicznych i dziwnych urządzeń wszelkiego rodzaju. Spędzaliśmy tam niezliczone godziny, kupując najrozmaitsze ustrojstwa za grosze i rozpracowując je, aby tworzyć nasze prototypy. Kiedy można kupić tysiąc głośników za bardzo małe pieniądze, niewątpliwie wpadniemy w końcu na jakiś pomysł,

1 Cytowany w artykule Sary Reese Hedberg, „MIT Media Lab’s quest for perceptive computers,” *Intelligent Systems and Their Applications*, IEEE, lipiec/sierpień 1998.