



pod redakcją

Mariana Nogi  
i Jerzego S. Nowaka

# POLSKA INFORMATYKA: WIZJE I TRUDNE POCZĄTKI

70<sup>lecie</sup>  
POLSKIEJ  
INFORMATYKI  
1948-2018



pod redakcją

---

Mariana Nogi  
i Jerzego S. Nowaka

# **POLSKA INFORMATYKA:**

**WIZJE I TRUDNE  
POCZĄTKI**

**70** lecie  
POLSKIEJ  
INFORMATYKI

---

1948-2018

pod redakcją

—  
Mariana Nogi  
i Jerzego S. Nowaka

# **POLSKA INFORMATYKA: WIZJE I TRUDNE POCZĄTKI**

POLSKIE TOWARZYSTWO INFORMATYCZNE

Warszawa 2017

*Recenzja:*

Prof. dr hab. Marek Greniewski

*Koordynator projektu:*

Bianka Piwowarczyk-Kowalewska

*Korekta:*

Bogusława Otfinowska

*Projekt okładki:*

Krzysztof Kanoniak

*Skład i łamanie:*

Michał Kośnik

Na okładce wykorzystano fotografie pochodzące ze zbiorów  
Narodowego Archiwum Cyfrowego.

Copyright © by Polskie Towarzystwo Informatyczne, Warszawa 2017  
Wszelkie prawa zastrzeżone.

Kopiowanie, przedrukowywanie i rozpowszechnianie niniejszej książki  
lub jej fragmentów bez pisemnej zgody wydawcy zabronione.  
Treść książki stanowi prywatną opinię i stanowisko Autorów.

Produkcja

PRESSCOM Sp. z o.o.

ul. T. Kościuszki 29

50-011 Wrocław

tel. 71 797 28 08

faks 71 797 28 16

e-mail: [wydawnictwo@presscom.pl](mailto:wydawnictwo@presscom.pl)

Wydawca

Polskie Towarzystwo Informatyczne

ul. Solec 38 lok. 103

00-394 Warszawa

tel: +48 22 838 47 05

fax: +48 22 636 89 87

e-mail: [pti@pti.org.pl](mailto:pti@pti.org.pl)

ISBN 978-83-60810-86-6 – oprawa miękka

ISBN 978-83-60810-95-8 – oprawa twarda

ISBN 978-83-60810-87-3 – wersja elektroniczna

# Spis treści

Słowo wstępne .....	7
1. Wrocławskie Zakłady Elektroniczne. Okres komputerów Odra 1300 .....	11
■ <b>Eugeniusz Bilski, Thanasis Kamburelis, Bronisław Piwowar</b>	
2. Maszyna matematyczna – co to właściwie jest? .....	37
■ <b>Marek Hołyński</b>	
3. Własne konstrukcje, licencje, klony .....	55
■ <b>Tomasz Kulisiewicz</b>	
4. Rodzina maszyn K-202 / Mera-400 / MX-16 .....	95
■ <b>Andrzej Ziemkiewicz, Elżbieta Jezierska-Ziemkiewicz</b>	
5. Historia rozwoju informatyki w hutnictwie żelaza i stali .....	115
■ <b>Andrzej Goleń, Stanisław Gembalczyk, Andrzej Musioł</b>	
6. Zakłady mechaniczno-Precyzyjne „Mera-Błonie” w Błoniu k. Warszawy (1953–2003) .....	171
■ <b>Jerzy Bezpalko, Marek Bielobradek, Zygmunt Pasek</b>	
7. Historia informatyki PZL Mielec 1960–2014 .....	207
■ <b>Włodzimierz Adamski</b>	
8. Historia projektu „System Zarządzania Bazą Danych RODAN” (1974–1990) ...	251
■ <b>Witold K. Staniszkis</b>	
9. Komputer Odra 1103 .....	277
■ <b>Jur Lesiński, Piotr Kociatkiewicz</b>	





## Słowo wstępne

Otwierając tom, przywołamy pierwsze zdania z artykułu dr. inż. Marka Hołyńskiego:

W czwartek, 23 grudnia 1948 r., w gmachu Fizyki Doświadczalnej przy ul. Hożej w Warszawie, z inicjatywy wybitnego topologa, profesora Uniwersytetu Warszawskiego, dyrektora świeżo organizowanego Państwowego Instytutu Matematycznego (PIM) Kazimierza Kuratowskiego spotkało się kilku przyszłych pionierów elektronicznych maszyn liczących. Byli to, oprócz inicjatora spotkania, prof. Andrzej Mostowski – matematyk zajmujący się głównie logiką matematyczną i algebrą, dr Henryk Greniewski – matematyk i logik oraz trzech młodzi inżynierowie po studiach na Politechnice Gdańskiej – Krystyn Bochenek, Leon Łukaszewicz i Romuald Marczyński, późniejsi profesorowie.

Profesor Kuratowski podzielił się z zebranymi swoimi wrażeniami z naukowego pobytu w USA. Był pod wrażeniem elektronicznych maszyn liczących, które widział za oceanem, i był przekonany, że chociaż jedna taka maszyna powinna być zbudowana w naszym kraju. W rezultacie tego spotkania zapadła decyzja powołania w ramach PIM Grupy Aparatów Matematycznych (GAM) w wyżej wymienionym składzie pod kierunkiem Henryka Greniewskiego.

Tak to się właśnie zaczęło – 23 grudnia 1948 r. uznajemy za początek historii polskiej informatyki. Potem było różnie. Z trudem zbudowano pierwszą elektroniczną maszynę cyfrową – bo tak je wtedy nazywano – XYZ. Zaczęto tworzyć ramy organizacyjne dla nowej dziedziny nauki i przemysłu – powstał Instytut Maszyn Matematycznych, niedługo później – Zjednoczenie Przemysłu Automatyki i Aparatury Pomiarowej MERA i liczne fabryki produkujące podzespoły, urządzenia peryferyjne i gotowe komputery. Polska została włączona do współpracy międzynarodowej, zarówno poprzez zakup licencji (Odra 1300, drukarki, pamięci dyskowe itp.), jak i podjęcie prac związanych z maszynami Jednolitego Systemu. Ukazały się liczne publikacje książkowe, w tym znakomite serie wydawnicze WNT i PWN – warto zauważyć, że w gronie autorów nie brakuje polskich specjalistów, w odróżnieniu od czasów obecnych. Społeczność informatyków dysponowała własnym miesięcznikiem popularnonaukowym „Informatyka” oraz licznymi biuletynami technicznymi („Zjednoczenie MERA”). W 1981 r. zawiązało się Polskie Towarzystwo Informatyczne. Od połowy lat 80. w kraju obserwowano zastosowania mikrokomputerów – polski przemysł próbował podjąć produkcję tych urządzeń, choć bez specjalnych sukcesów.

W 1989 r. przychodzi krach – polski przemysł komputerowy nie wytrzymuje zderzenia z gospodarką wolnorynkową, a w szczególności z napływem taniego, często używanego sprzętu komputerowego z zagranicy. Kadra – znakomicie wyszkolona w minionych latach – radzi sobie w tych warunkach, tworząc liczne firmy informatyczne – powstaje nowy przemysł informatyczny.

Konsekwencją tych wydarzeń jest likwidacja istniejących zakładów, rozproszenie kadr i bardzo często – zniszczenie archiwów. Zaczyna pojawiać się myśl o konieczności zachowania dorobku nauki i przemysłu komputerowego.

Pierwsze próby podejmuje PTI w 1988 r., organizując konferencję z okazji 40-lecia polskiej informatyki. Głos zabierają wtedy sami twórcy – byli jeszcze wśród nas. Dorobek konferencji publikuje w specjalnym wydaniu „Informatyka”<sup>1</sup>. Ten zestaw artykułów staje się na wiele lat kanonem wiedzy o historii polskiej informatyki.

Życie pokazuje, że to za mało – pojawiło się zbyt wiele opinii niemających pokrycia w faktach, ale trudnych do obalenia z powodu braku dokumentów i relacji. W takiej sytuacji w ramach Polskiego Towarzystwa Informatycznego zawiązała się grupa dyskusyjna zajmująca się historią polskiej informatyki. Pierwsze prezentacje i komunikaty wskazywały na potrzebę kontynuowania prac – grupa została przekształcona w Sekcję Historyczną PTI. Rozpoczęło się poszukiwanie materiałów, odtwarzanie kontaktów itp. Dość szybko okazało się, że brak czasopisma popularnonaukowego był i jest wyraźną przeszkodą w informowaniu o dziejach polskiej informatyki. Uruchomiony portal historyczny stał się w tej sytuacji najbogatszym obecnie źródłem takiej wiedzy w kraju, tworząc za zgodą autorów i posiadaczy dokumentów cyfrowe archiwum historii informatyki polskiej. Warto też odnotować pierwszą publikację historyczną PTI z 2011 r. – *Wczoraj, dziś i jutro polskiej informatyki*.

W konsekwencji tych działań w 2016 r. PTI ogłosiło konkurs wydawniczy na opracowania z historii polskiej informatyki. Plonem konkursu jest kilkanaście artykułów omawiających historię instytucji i wybranych przedsięwzięć oraz dwie publikacje książkowe. Pewnym rozczarowaniem jest brak inicjatywy stworzenia całościowego opracowania historii polskiej informatyki – jak widać, brak materiałów utrudnia opracowanie takiej syntezy.

Nadesłane artykuły po recenzjach są drukowane w dwóch odrębnych tomach. Jeden tom poświęcony jest szeroko rozumianemu przemysłowi informatycznemu, drugi – wybranym aplikacjom i zastosowaniom informatyki. Wydawca przedstawia te publikacje jako początek obchodów 70. rocznicy polskiej informatyki przypadającej na grudzień 2018 r.

Otwierając niniejszy tom poświęcony głównie technicznemu aspektowi polskiej informatyki, Czytelnik ma szansę zapoznać się z następującymi relacjami:

- E. Bilski, T. Kamburelis i B. Piwowar przedstawiają osobistą relację z pracy w WZE Elwro; dość długo czekała ona na druk, ale mamy okazję zapoznać się z opiniami twórców pierwszych komputerów serii Odra 1300 i R-32. Do ich relacji dołączamy kopie porozumień zawartych z firmą ICL z lipca 1967 r. – po raz pierwszy w kraju.
- M. Hołyński kreśli zarys historii Instytutu Maszyn Matematycznych – jest to szczególnie zasłużona placówka funkcjonująca praktycznie od początków informatyki w Polsce.
- T. Kulisiewicz podjął się trudnego zadania, omawiając – po raz pierwszy w Polsce – zarys historii Jednolitego Systemu. Zdaniem redaktorów jest szansa, że wreszcie znikną tzw. legendy miejskie związane z tym tematem. Odwołanie się do szeregu sprawozdań dawnego Komitetu Nauki i Techniki pokazuje, że Polska była żywotnie zainteresowana podjęciem współpracy, licząc na duży eksport urządzeń komputerowych do krajów RWPG.

---

1 „Informatyka” 1989, nr 7–8.

- A. Ziemkiewicz i E. Jezierska-Ziemkiewicz w żywy i barwny sposób opisali koncepcje architektoniczne słynnego minikomputera K-202. Redaktorzy tomu są zdania, że pozwoli to wreszcie zamknąć wszelkie dyskusje na temat walorów technicznych tego komputera.
- Zespół autorski A. Goleń, S. Gembalczyk i A. Musioł prezentujący dawny CIBEH i Hutę im. Lenina przedstawił szeroki zarys informatyzacji polskiego hutnictwa żelaza i stali. Wraz z przedstawieniem historii rozwoju informatyki w hutnictwie autorzy pokazali złożoność tej branży w jej historycznym rozwoju, odwołując się również do czasów przedwojennych.
- Zespół autorski byłych pracowników Mera-Błonie (J. Bezpałko, M. Bielobradek, Z. Pasek) przygotował z kolei skrócony zarys historii Zakładów. W końcu lat 80. była to największa fabryka drukarek komputerowych w Europie i dziwi nieco fakt, że tak łatwo doprowadzono do jej likwidacji.
- W. Adamski podjął się trudnej roli omówienia dorobku projektowania inżynierskiego w budowie samolotów na przykładzie Zakładów PZL Mielec, kreśląc przy okazji zarys historii informatyki w tej firmie.
- W. Staniszkis opisuje dzieje powstania istotnej aplikacji komputerowej, czyli bazy danych RODAN – był to jedyny przypadek podjęcia się tak trudnego zadania w Polsce.
- Przegląd artykułów kończy krótki komunikat o nietypowym komputerze Odra 1103, będącym odpowiednikiem urządzeń Aritma DP-100 czy EW-80, czyli kalkulatora zamykającego cykl obliczeniowy maszyn licząco-perforacyjnych.

Czytelnikowi należy się jeszcze jedno wyjaśnienie – w omawianym okresie nazwy zakładów produkcyjnych ulegały dość częstym zmianom, co nie zawsze znajduje odzwierciedlenie w treści artykułów. Poczyniona uwaga dotyczy także wielkości produkcji – Autorzy podają dane występujące w dostępnych materiałach. Na podstawie szeregu dokumentów ujawnionych w 2016 r. konieczne będzie zweryfikowanie tych danych.

Życzymy ciekawej lektury i zapraszamy do sięgnięcia po część drugą publikacji.

Redaktorzy



mgr inż. Andrzej Ziemkiewicz, mgr inż. Elżbieta Jezierska-Ziemkiewicz

## Rodzina maszyn K-202 / Mera-400 / MX-16

### Spis treści

1. Przedmowa .....	97
1.1. Źródła wiedzy o K-202 .....	97
1.2. Konkluzja .....	98
2. Architektura maszyny K-202 .....	98
2.1. Architektura globalna maszyny.....	98
2.2. Organizacja pamięci .....	99
2.3. Rejestry procesora.....	102
2.4. Wykonanie rozkazu .....	102
2.5. Lista rozkazów .....	103
2.6. Rozkazy warunkowe i patent Jacka Karpińskiego.....	103
2.7. Parę uwag porównawczych.....	104
2.8. Podsumowanie .....	105
2.9. Śmierć maszyny .....	105
3. Maszyna Mera-400.....	106
3.1. Architektura globalna .....	106
3.2. Organizacja pamięci .....	107
3.3. Zarządzanie dostępem do szyny .....	108
3.4. Rejestry maszyny .....	108
3.5. Rozkazy maszyny.....	109
3.6. Podsumowanie .....	109
4. Maszyna MX-16.....	109
4.1. Pamięć (Mega).....	109
4.2. Architektura .....	110
4.3. Podsumowanie .....	110
5. Oprogramowanie .....	111
6. Zakończenie.....	111
6.1. IMM .....	112
6.2. Elwro.....	112



## 1. Przedmowa

Od 40 lat maszyna K-202 wzbudza zamęt w mediach i w internecie.

Piszą o niej różni autorzy, a zdania są skrajnie różne. Jedni uważają, że „wyprzedzała o 10 lat...”, a inni, że inne maszyny były równie dobre, szybkie itd., a wszyscy zaznaczają, że nie znaleźli żadnej dokumentacji (czyli nic o niej nie wiedzą), więc opierają się na pogłoskach lub np. raporcie komisji Kilińskiego. Istotnie, raport komisji Kilińskiego nie jest dokumentacją i nie można dowiedzieć się z niego zbyt wiele o maszynie.

Jeżeli ktoś nie ma żadnych dokumentów źródłowych pozwalających się wypowiadać, to dlaczego u licha zabiera głos? Jest to trochę niepoważne. Rzetelny autor podjąłby raczej wysiłek, żeby dotrzeć do dokumentów. Być może zawiniło tu nadmierne zaufanie ludzi do Google’a. Zadajemy pytanie, a jeżeli Google natychmiast nie poda odpowiedzi, uznajemy, że odpowiedź nie istnieje, i nie szukamy dalej. Czyż nie tak? Ale „prawdziwa wiedza zaczyna się tam, gdzie Google nie sięga”, jak mówi żartobliwe powiedzenie.

A nie jest prawdą, że takich źródeł nie było.

### 1.1. Źródła wiedzy o K-202

Są co najmniej trzy miejsca, w których można było znaleźć informacje o maszynie.

#### 1.1.1. Dokument Maszyna cyfrowa K-202. Organizacja logiczna

Jest to opis maszyny, który przygotowałem właściwie dla programistów. Każdy programista w zespole miał jeden egzemplarz i na pewno chętnie by go udostępnił. Opis ten został zeskanowany i umieszczony w internecie przez pana Ryszarda Zenkera. Jest powszechnie dostępny pod adresem: <http://www.zenker.poznan.pl/k-202/dokumentacja/k-202-opis.pdf>.

Dokument ten (3,2 MB) zawiera kompletne dane o maszynie, począwszy od postaci informacji, poprzez schemat blokowy, rejestry, ogólne zasady wykonywania rozkazów, pełną listę rozkazów, system przerwán itd., aż do najdrobniejszych szczegółów. Krótko mówiąc, jest to pełny *Principles of operation* maszyny.

Sprawdziłem ostatnio, że adres ten ciągle działa.

#### 1.1.2. Dokumentacja techniczna

Zespół konstruktorów posiadał pełny komplet schematów maszyny. Nie było problemu, żeby do nich dotrzeć i sprawdzić na schematach. Każdy na pewno chętnie by je udostępnił.

#### 1.2.3. Projektanci maszyny

Ostatnim źródłem są wreszcie jeszcze żyjący projektanci maszyny. Kto mógłby lepiej wyjaśnić kwestie i odpowiedzieć na pytania niż ci, którzy ją projektowali?

Z przykrością muszę stwierdzić, że przez te wszystkie lata żaden z dyskutantów nie zwrócił się ani do mnie, ani do Eli, chociaż mogli i powinni byli, zanim zaczęli pisać byle co. Chętnie odpowiedzielibyśmy na wszystkie pytania, a co najmniej wysłali dokumentację.

## 1.2. Konkluzja

Z tego, co napisano powyżej, wynika jasno, że autorzy mogli mieć pełną wiedzę o maszynie, ale z tego nie skorzystali. Dlaczego? Może nie chodziło im wcale o prawdę. Zatem o co? Żeby zaistnieć na forum?

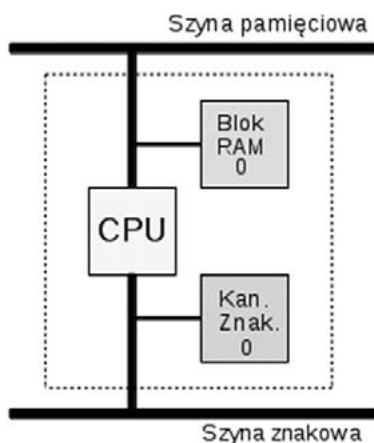
Wszystko jedno, uważam, że nie można nadal pozwalać na pisanie i wygłaszanie byle czego. Postanowiłem więc zapłacić lukę i przedstawić tę maszynę i jej następną wersję, czyli maszyny Mera-400 i MX-16. Warto to zrobić choćby dlatego, że maszyny te miały różne znakomite rozwiązania. Będę się starał, aby opis ten był możliwie zwięzły i zawierał tylko rzeczy najważniejsze. Pełna dokumentacja znajduje się w wyżej wymienionym dokumencie.

Wszystko, co będzie napisane poniżej, znamy z własnego doświadczenia, a nie z pogłosek.

## 2. Architektura maszyny K-202

### 2.1. Architektura globalna maszyny

Kiedy spotkaliśmy się z Jackiem Karpińskim na początku października 1970 r., zaproponował nam pracę nad minikomputerem K-202, i przedstawił jego koncepcję (rys. 1a).



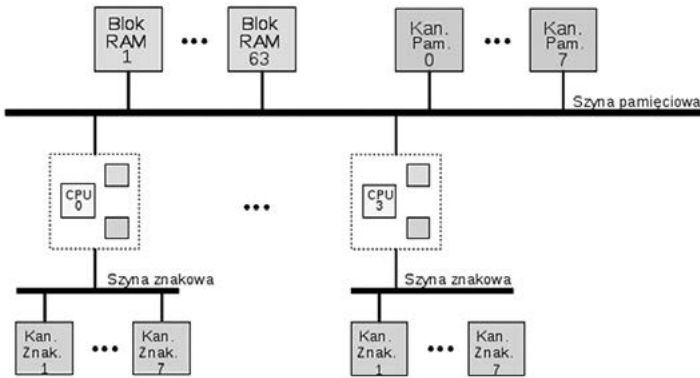
Rysunek 1a. Moduł procesora K-202

Komputer ma dwie szyny: pamięciową, do której dołączane są bloki pamięci i kanały pamięciowe (DMA), oraz znakową, do której dołączane są kanały znakowe.

Koncepcja jak koncepcja, nawet zgrabna, zależy, co się z niej zrobi.

Zaproponowaliśmy rozszerzenie szyny pamięciowej o bity zawierające numer procesora, tak aby system mógł być wieloprocessorowy, co ostatecznie dało schemat:





Rysunek 1b. Schemat blokowy K-202 (teoretyczna konfiguracja max)

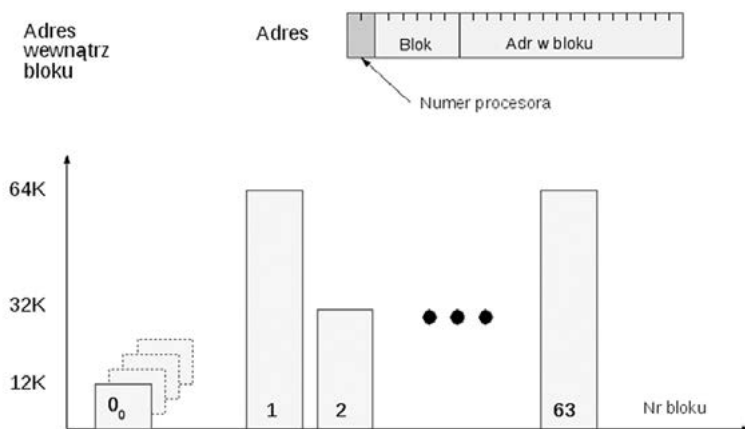
To już wyglądało znacznie poważniej. Można skonfigurować naprawdę wielki system. Oczywiście należało jeszcze rozwiązać mnóstwo problemów technicznych – jaka ma być adresacja i organizacja pamięci, jakie ma być sterowanie maszyny itd.

## 2.2. Organizacja pamięci

Pamięć miała organizację słowową, o długości słowa 16 bitów. Adres pamięci składał się z 16-bitowego adresu wewnątrz bloku i 6-bitowego numeru bloku, czyli w sumie 22 bity, co tworzyło przestrzeń adresową 4 Mśłów (8 MB). Blok pamięci o numerze 0 był blokiem systemowym, dostępnym tylko dla systemu operacyjnego. Każdy procesor miał swój prywatny blok zerowy. Bloki 1–63 były dostępne dla wszystkich procesorów.

Adres wewnątrz bloku był wysyłany na szynę adresową przez program. Numer bloku był dopisywany automatycznie z rejestru numeru bloku, do którego nie miał dostępu program użytkowy, a tylko system operacyjny, który przypisywał numer bloku do programu.

Najłatwiej wyobrazić sobie pamięć maszyny jako dwuwymiarową, gdzie pierwszy wymiar to numer bloku (przypisany do programu), a drugi to adres wewnątrz bloku. Przedstawiam to na poniższym rysunku.



Rysunek 2. Pamięć komputera K-202

Było to rozwiązanie unikalne i miało szereg zalet. Po pierwsze, praktycznie nic nie kosztowało. Nie był potrzebny żaden układ translacji adresów logicznych na fizyczne, jaki jest stosowany w innych maszynach. Po drugie, nie był potrzebny żaden układ kontrolujący, czy program nie wykracza poza przydzieloną mu przestrzeń adresową i czy nie wchodzi w cudzą, bo było to fizycznie niemożliwe. Zapewniało to doskonałą ochronę pamięci programu przed zepsuciem przez inny program w wyniku jakiegoś błędu. Po trzecie, czas dostępu do pamięci nie był wydłużany przez operacje translacji i sprawdzania.

Dla małej maszyny, jaką było K-202, było to rozwiązanie idealne, choć miało jedną słabą stronę. Nie było dynamicznej alokacji pamięci. Podział pamięci na bloki i złożenie bloków ze stron pamięci (mających po 4 kśłów) był statyczny, dokonywany za pomocą zworek zakładanych na styki w łączówce. Niektórzy wytykali to jako kompromitujące. Dlaczego? Inne minikomputery też nie posiadały dynamicznej alokacji i nikt im tego nie wytykał. Poza tym było to przez nas traktowane jako rozwiązanie przejściowe, które miało zostać zaraz ulepszone w następnej wersji. Istotnie, zostało to zrobione w maszynie Mera-400.

Trzeba tu przypomnieć, że podczas projektowania panował wielki pośpiech. Warunek postawiony przez Jacka Karpińskiego był następujący: „Musimy tę maszynę zrobić szybko, bo musi być wystawiona na Targach Poznańskich w czerwcu, a później na ekspozycji Olympia w Londynie, bo jeżeli nie zdążymy, to inni zajmą rynek”. Oczywiście była to prawda. Jeżeli chcieliśmy mieć eksport na Zachód, to trzeba było się liczyć z nieubłaganymi prawami gospodarki wolnorynkowej. Wystarczy przypomnieć historię mikroprocesorów. Intel wypuścił swój pierwszy 8-bitowy 8080 i zajął rynek. Firma Zylog wyprodukowała swój Z80 zaledwie 6 miesięcy później i już na niego nie weszła. Historia powtórzyła się z mikroprocesorami 16-bitowymi. Motorola wypuściła procesor 68000 pół roku po 80286 Intela i już zastała rynek zajęty. A Zylog ze swoim Z8000 wypuszczonym jeszcze 6 miesięcy później nie przebił się w ogóle, chociaż miał najpiękniejszą architekturę spośród tych trzech mikroprocesorów.

Zatem na zaprojektowanie, wykonanie i uruchomienie maszyny mieliśmy zaledwie osiem miesięcy. Zdążyliśmy i był to chyba rekord w szybkości opracowania i uruchomienia maszyny, ale trzeba było to robić szybko i prosto, a skomplikowane rozwiązania odkładać do drugiej wersji. Dynamiczna alokacja musiała poczekać. Należy tu okazać zrozumienie, a nie krytykować.

### 2.2.1. Sterowanie pamięci

Sterowanie pamięci było całkowicie asynchroniczne. Nie było żadnego centralnego zegara taktującego. Oznacza to, że blok pamięci, do którego nastąpiło odwołanie, wysyłał sygnał, że ma gotową informację, natychmiast gdy była ona gotowa. Nie musiał czekać na następny impuls zegarowy. Przy sterowaniu synchronicznym, jeżeli pamięć mogłaby dostarczyć informację np. w czasie 2,5 taktu zegarowego, to i tak następna operacja mogłaby się zacząć dopiero w 3. takcie, więc pół taktu zegarowego musi być zmarnowane. Ogólnie można powiedzieć, że przy identycznych wszystkich innych parametrach maszyna sterowana asynchronicznie będzie szybsza.

Ten system miał dwie zalety. Po pierwsze, nie było żadnych przestojów w oczekiwaniu na impuls zegarowy i związanej z tym straty czasu. Po drugie, nie był wymagany żaden określony timing.

Jak ważna była ta ostatnia cecha, miałem okazję przekonać się osobiście. Kilkanaście lat temu kupiłem sobie komputer PC. Miał zegar 2 MHz, 640 GB dysku i 512 MB RAM-u. Dziś to niewiele, ale wtedy to był „top model”. Uważałem tylko, że te 512 MB RAM-u to trochę mało, więc kiedy minęły dwa lata i skończyła się gwarancja, rozkręciłem obudowę i zobaczyłem, że oprócz baretki 512 MB jest drugi wolny slot – na drugą baretkę. Poszedłem do sklepu i kupiłem baretkę 512 MB, wstawiłem do maszyny i włączyłem. Maszyna w ogóle nie wystartowała. Myślałem, że może coś zepsułem, więc wyjąłem tę nową baretkę i maszyna ruszyła. Wyjąłem więc starą baretkę i wstawiłem nową – maszyna też ruszyła. Z obydwoma naraz nie chciała. Zacząłem więc przyglądać się nadrukowi na układach scalonych pamięci i zauważyłem, na co uprzednio nie zwróciłem uwagi, że stare były serii „i2”, a nowe – serii „i3”, czyli trochę nowsze, szybsze i o trochę innym timingu. Wyjaśniło się. Procesor, mając dwa bloki pamięci o różnych timingach, był tym tak ogłupiony, że nie chciał ruszyć z miejsca. Jest to jedna z pułapek synchronicznego sterowania. Każdy może powtórzyć ten eksperyment.

Maszynie K-202 nic takiego nie groziło. Można było dołączyć bloki pamięci o rozmaitych czasach cyklu i wszystko działało świetnie. Typowy czas cyklu w używanych przez nas pamięciach ferrytowych produkcji irlandzkiej firmy Data Products wynosił 1  $\mu$ s (czyli czas dostępu 0,5  $\mu$ s). Asynchroniczne było nie tylko sterowanie pamięcią, ale również szynami – pamięciową i znakową oraz całe sterowanie w procesorze. Dzięki temu maszyna była niewrażliwa na jakość elementów. Jeżeli udało się dostać partię elementów na układach Schottky’ego, to tym lepiej. Maszyna była wtedy szybsza. Jeżeli nie, to też nie było tragedii. Była wtedy wolniejsza, ale działała równie pewnie.

### 2.2.2. Zarządzanie dostępem do szyny

W każdej maszynie posiadającej szynę, do której dołączone są różne urządzenia, powstaje konieczność zarządzania tą szyną, tak aby nie było konfliktu przy próbie równoczesnego dostępu do niej przez dwa lub więcej urządzeń. Zazwyczaj jest to wykonywane przez układ arbitrażowy, który przydziela dostęp do szyny i zwykle zainstalowany jest w procesorze. Ale w systemie wieloprocesorowym? Tu sprawa się komplikuje.

Ponadto, na ile urządzeń ma być przewidziany taki skupiony układ arbitrażowy? 2, 4 czy 16? Zaprojektowany na zbyt wiele urządzeń jest niepotrzebną rozrzutnością, zaprojektowany na zbyt mało też jest niedobry, bo co zrobić, aby dołączyć więcej? Przerabiać układ, kupić większy model maszyny? Dodatkową niewygodą jest konieczność ciągnięcia indywidualnych kabli z tego układu arbitrażowego do urządzeń. Przy dołączeniu nowego urządzenia trzeba prowadzić dodatkowe nowe indywidualne kable.

Dlatego zdecydowaliśmy, że w maszynie nie będzie układu scentralizowanego. Przez naszego przyjaciela, dr. Andrzeja Karczmarewicza, został zaprojektowany rozproszony układ przydziału dostępu do szyny, który dał się podzielić na segmenty w taki sposób, że każdy klient szyny (procesor, kanał pamięciowy itd.) posiadał jeden jego segment i dołączał się do szyny wraz ze swoim segmentem. Rozwiązanie było unikalne, nowatorskie i działało znakomicie.

Wygodą była ogromna. Aby dołączyć do szyny nowe urządzenie, nie była potrzebna żadna, nawet najdrobniejsza przeróbka. Wystarczało włączyć wtyczkę kabla urządzenia do gniazdka szyny i działało.

### 2.3. Rejestry procesora

Procesor K-202 miał 8 głównych rejestrów. Rejestr nazywany R0 był rejestrem stanu procesora. Jedna jego część przechowywała stan procesora i do tej części programy użytkowe nie mogły pisać. Druga część była dostępna do użytku programów użytkowych.

Pozostałe rejestry, nazywane R1–R7, były rejestrami uniwersalnymi do dowolnego użytku. Osiem rejestrów to zupełnie nieźle jak na minikomputer i K-202 nie ma się tu czego wstydzić. Nawet wielkie maszyny, takie jak IBM 360/370, miały tylko 16 rejestrów uniwersalnych, a kosztowały miliony.

Każdy rejestr miał 16 bitów i mógł przechowywać liczbę stałoprzecinkową pojedynczej długości lub informację logiczną. Liczby podwójnej precyzji były przechowywane w parze dwóch kolejnych rejestrów. Liczby zmiennoprzecinkowe zajmowały trzy kolejne rejestry.

Poza wspomnianymi rejestrami procesor miał kilka dodatkowych rejestrów, licznik rozkazów, rejestr numeru bloku pamięci, rejestr przerwań itd., o czym nie będę już tutaj wspominać. Są one dokładnie opisane w dokumentacji.

### 2.4. Wykonanie rozkazu

Przy wykonaniu rozkazu najważniejszą operacją było wyznaczenie efektywnego argumentu. Zaczynało się ono od wyznaczenia pierwotnego argumentu, który mógł być

zawartością rejestru, miejsca pamięci lub być umieszczony bezpośrednio w rozkazie. Następnie mógł on podlegać modyfikacjom. Modyfikacja polegała na indeksacji rejestrów i/lub pośrednim adresowaniu. Możliwe były trzy kolejne modyfikacje argumentu. W sumie był to aparat bardzo bogaty, który umożliwiał działanie na skomplikowanych strukturach danych. Na koniec wykonywana była operacja określona przez kod operacji rozkazu.

## 2.5. Lista rozkazów

Lista rozkazów zawierała rozkazy dwuargumentowe, jednoargumentowe i bezargumentowe. Rozkazy dwuargumentowe mogły być typu rejestr–rejestr lub rejestr–pamięć i zawierać operacje arytmetyczne, logiczne lub zmiennoprzecinkowe. Maszyna miała różne bardzo wygodne rozkazy. Wymienię trzy przykłady tych ciekawszych. Pierwszy przykład to rozkazy tzw. przesyłania grupowego. Za pomocą jednego rozkazu można było przesłać z/do pamięci zawartość dwóch, trzech lub siedmiu rejestrów uniwersalnych. Drugi to rozkazy przesyłania z/do pamięci według adresu wskazanego rejestrów z automatyczną dekrementacją/inkrementacją tego rejestru. Ułatwiała to programowi zorganizowanie stosu w pamięci, używając jako wskaźnika stosu któregoś z rejestrów uniwersalnych.

Trzecim był rozkaz „ekstrakod”, dziś powiedzielibyśmy „system call”. Służył on do rozszerzania listy rozkazów. Za pomocą tego rozkazu program użytkowy mógł zażądać wykonania funkcji, której ciało znajdowało się w systemie operacyjnym.

Reszta listy rozkazów nie odbiegała w zasadzie od standardu.

## 2.6. Rozkazy warunkowe i patent Jacka Karpińskiego

Od lat czytam i słyszę różne aluzje na temat patentów Jacka Karpińskiego w maszynie K-202 i o różnych, rzekomo groźnych ich konsekwencjach. To jest chyba moment, aby wreszcie tę sprawę wyjaśnić. Jacek Karpiński mógł mieć nawet 50 patentów i nie o to chodzi. Chodzi o to, ile z nich było użytych w maszynie K-202. Słyszysz się o dwóch, a kto potrafi powiedzieć, czego one dotyczyły? Otóż tylko jeden patent Jacka Karpińskiego był użyty w K-202. Dotyczy on rozkazów warunkowych.

Rozkazy warunkowe służą do robienia rozgałęzień w programie. Metoda jest następująca:

1. Ustalamy jakiś warunek, np. za pomocą rozkazu porównania, który ustawia wskaźniki „mniejszy/równy/większy”;
  2. Następnie stosujemy rozkaz skoku warunkowego, np. „skocz gdy mniejszy”.
- Jeśli warunek nie jest spełniony, program kontynuuje, a jeśli jest – wykonuje skok do innej gałęzi programu, wskazanej adresem skoku.

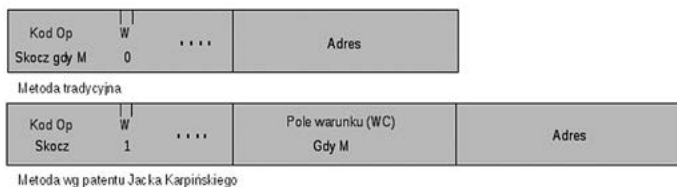
Każda maszyna posiada w liście rozkazów komplet skoków warunkowych:

- „skocz gdy mniejszy”,
- „skocz gdy równy”,
- „skocz gdy większy” itp.

Maszyna K-202 również posiadała taki komplet skoków warunkowych, ale dodatkowo miała inny mechanizm, określony patentem, który mówił, że:

- Każdy rozkaz ma bit, zwany bitem W, który mówi, czy rozkaz jest warunkowy, czy nie.
- Jeżeli ten bit ma wartość „0”, to rozkaz nie jest warunkowy.
- Jeżeli ten bit ma wartość „1”, to rozkaz jest warunkowy, a warunek jest wskazany w następnym słowie rozkazu, nazywanym polem warunku (WC).

W rezultacie rozgałęzienie w programie można było zrobić na dwa sposoby. Pokazuje to rysunek 3.



Rysunek 3. Porównanie metod tworzenia rozkazów warunkowych – tradycyjnej i wg patentu Jacka Karpińskiego

Funkcjonalnie obie metody są równoważne, ale wątpię, czy programiści użyli kiedykolwiek metody według patentu, bo program jest wtedy dłuższy o jedno słowo, a wykonanie wolniejsze o jeden cykl pamięci. Co gorsza, patent ten wymaga, aby każdy rozkaz w maszynie miał bit W. Zabiera to jeden bit z kodu operacji. Krótko mówiąc, zmniejsza to dwukrotnie liczbę rozkazów maszyny.

Najważniejsze jest to, że nie był to patent, który wymagał później jakiegoś omijania, i że to omijanie mogło być w czymś szkodliwe dla maszyny. Wręcz przeciwnie. Zamiast cokolwiek omijać, wystarczyło z niego zrezygnować.

Ostatnia uwaga – teoretycznie patent mógł mieć tę zaletę, że warunkowe mogły być nie tylko rozkazy skoku, ale i inne. Powiem, że szczerze szukałem przypadku, w którym mogło to być lepsze, ale nie znalazłem. Jeżeli ktoś znajdzie taki przykład, to postawię mu szampana.

Drugi patent Jacka Karpińskiego dotyczył sposobu zajmowania szyny. Jednak w K-202 patent ten nie został użyty. Użyliśmy układu zaprojektowanego przez dr. Andrzeja Karczmarewicza, o czym wspominałem już wyżej.

## 2.7. Parę uwag porównawczych

Jeżeli chce się porównać tę maszynę z innymi, należy brać pod uwagę inne komputery 16-bitowe, które pojawiły się w tym samym czasie. Było ich dosyć sporo – Varian, Nova, Super-Nova itd. Osobiście najbardziej podobała mi się maszyna Varian. Wszystkie zniknęły z rynku prawie równie szybko, jak się pojawiły, zatem ostatecznie jedynym prawdziwym konkurentem była maszyna DEC PDP 11/20. Została ona sprzedana w liczbie ponad 0,5 mln egzemplarzy. Sens ma porównanie K-202 jedynie z tą maszyną.

Firma DEC nie lubiła się chwalić parametrami maszyny PDP 11/20. Niemniej w dokumencie firmowym *PDP 11/20 Processor Handbook* można znaleźć informację, że czas

dośćępu do pamięci dla odczytu rozkazu wynosił 1  $\mu$ s (czyli czas cyklu – ok. 2  $\mu$ s), a następnie czas wykonania operacji typu rejestr-rejestr w procesorze wynosił 0.9  $\mu$ s, czyli w sumie 1.9  $\mu$ s, czyli dwa razy wolniej niż w K-202.

Jeżeli chodzi o pamięć, to maszyna PDP 11/20 standardowo miała pamięć 28 KB (adresacja była bajtowa, a nie słowowa). Można było zamówić opcję tzw. memory management. Był to dołączany do procesora układ logiczny, który pozwalał podwoić pamięć metodą przełączania między dwoma blokami. W każdym razie pamięć nie mogła przekroczyć 64 KB, ponieważ adres był 16-bitowy i nie było mechanizmów, które mogły to rozszerzyć.

W rzeczywistości to maksimum było nieosiągalne. Każde odwołanie do adresu powyżej 60 KB było bowiem interpretowane przez sprzęt jako odwołanie do urządzeń wejścia/wyjścia.

Maszyna PDP 11/20 nie posiadała arytmometru zmiennoprzecinkowego, który pojawił się dopiero w modelu 11/40.

Można w sumie powiedzieć, że ten najważniejszy konkurent K-202 był dwa razy wolniejszy, a pod względem architektury dużo uboższy.

## 2.8. Podsumowanie

Maszyna K-202 była bardzo dobrą maszyną, zawierającą dużo interesujących rozwiązań technicznych, i naprawdę trudno pokazać maszynę tej klasy istniejącą na rynku w tym samym czasie (rok 1971), która mogłaby z nią konkurować.

Miała również pewne inne miłe cechy. Była odporna na wstrząsy. Marynarka Wojenna zainstalowała jedną na kutrze torpedowym, gdzie są duże wibracje podłoża. Maszyna działała cały czas – jak na wytrząsarce, ale niezawodnie.

Dzięki sprawnej wentylacji nie wymagała klimatyzowanych pomieszczeń i mogła pracować gdziekolwiek.

## 2.9. Śmierć maszyn

W różnych wypowiedziach można przeczytać, że historia K-202 skończyła się, kiedy firmy angielskie wycofały się ze współpracy, bo rzekomo partia dostarczonych maszyn była wadliwa. Wszyscy, którzy tak piszą, dają tylko świadectwo, że nie mają pojęcia o realiach tamtych czasów. Formalnie niemożliwe było dokonanie dostawy maszyny, jeżeli nie przeszła ona pomyślnie testów. Maszyny te jednak przeszły pomyślnie testy. Co się zatem stało? Było to w czasie, gdy Jacek Karpiński był już w nielascie i władze „wtykały mu patyk w szprychy”. Zakazano wysyłki tych maszyn z matrycami pamięci, a zespół otrzymał polecenie, aby te matryce z maszyn wymontować. Pojechały więc bez pamięci. Pomysł iście machiaweliczny – później, jak wszyscy zapomną, jak to było, będzie można mówić, że to maszyny były wadliwe, co właśnie obecnie obserwujemy. Świadectwo z pierwszej ręki może dać tu Jurek Dżoga, który te maszyny uruchamiał. W rzeczywistości maszyny nie były wadliwe, tylko zostały celowo zdekompletowane. Mówiąc jaśniej, był to po prostu sabotaż. Nic dziwnego, że Anglicy nie chcieli takich maszyn. Władze szybko zarządziły zamknięcie projektu K-202.

Można powiedzieć, że śmierć maszyny to było zabójstwo, a obecnie niektórzy próbują mówić, że winna była ofiara. To nieładnie kopać leżącego, ale niektórzy znajdują w tym jakąś niezdrową przyjemność.

Ktoś napisał, że np. maszyna była budowana z matryc ferrytowych, na które było embargo. To nieprawda. Jak wiadomo, embargo było na układy scalone wielkiej skali integracji (LSI), natomiast nie było żadnego embargo na układy małej i średniej skali integracji ani na ferryty. Embargo na ferryty? To chyba jakiś żart – przecież elementy do K-202 były zamawiane oficjalnie i legalnie, a nie przemycane.

Jeszcze ktoś inny pisał, że według umowy strona polska nie mogła sprzedawać maszyn bez przyzwolenia strony angielskiej i rozliczenia się z każdej maszyny. To są już zwykle bajeczki. Tak się składa, że czytałem tę umowę i nic takiego z niej nie wynika. Strona angielska miała prawo rozprowadzać maszyny poza granicami Polski, a strona polska – wewnątrz. Nie musiała się przy tym przed Anglikami z niczego rozliczać i się nie rozliczała.

Mogę zrozumieć, że ktoś może mieć jakieś osobiste porachunki z tą maszyną, ale nie upoważnia go to do używania nieczystych metod.

### 3. Maszyna Mera-400

Maszyna K-202 była dobra, ale pierwsza wersja urządzenia nigdy nie jest doskonała. Zawsze znajdzie się coś, co można ulepszyć. Po wyrzuceniu Jacka Karpińskiego zespół, który pozostał, skonstruował maszynę Mera-400, którą należy traktować jako drugą wersję K-202, poprawioną i uzupełnioną.

Opiszę po kolei, jakie rzeczy zostały zmienione i w jaki sposób.

#### 3.1. Architektura globalna

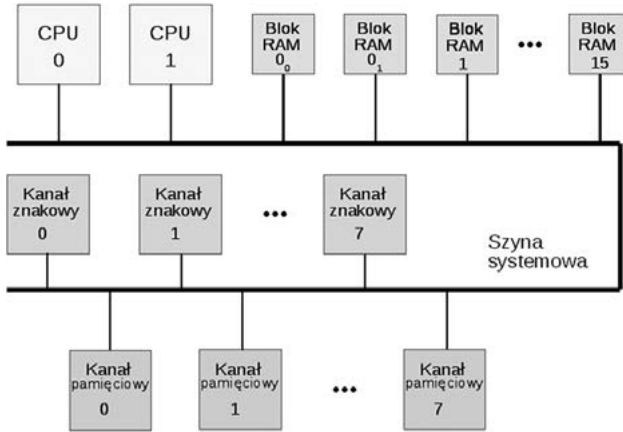
Doświadczenia wyniesione z K-202 pozwoliły wyciągnąć następujące wnioski:

- Posiadanie do 4 procesorów w jednym zestawie maszyny jest niecelowe, bo wówczas wąskim gardłem systemu staje się wspólna szyna, toteż liczba procesorów została ograniczona do 2.
- Posiadanie do 64 bloków pamięci było zbyt duże. Nikt nigdy nie potrzebował takiej pamięci, toteż liczba ta została ograniczona do 17 (dwa bloki zerowe – po jednym dla każdego procesora i 15 bloków użytkowych).
- Szyna znakowa prywatna dla każdego procesora okazała się niewygodna. Wady tego rozwiązania były następujące:
  - Jeżeli jakieś urządzenie, np. drukarka, było dołączone do szyny znakowej jednego z procesorów, to inne procesory nie miały do niej dostępu.
  - Konieczność obsługi każdego znaku z urządzenia przez procesor powodowała duży narzut systemowy.
  - Stosowanie DMA (*Direct Memory Access*) było niemożliwe.



Zatem szyna znakowa została zlikwidowana, a wszystkie kanały, zarówno pamięciowe, jak i znakowe, zostały dołączone do jednej szyny systemowej.

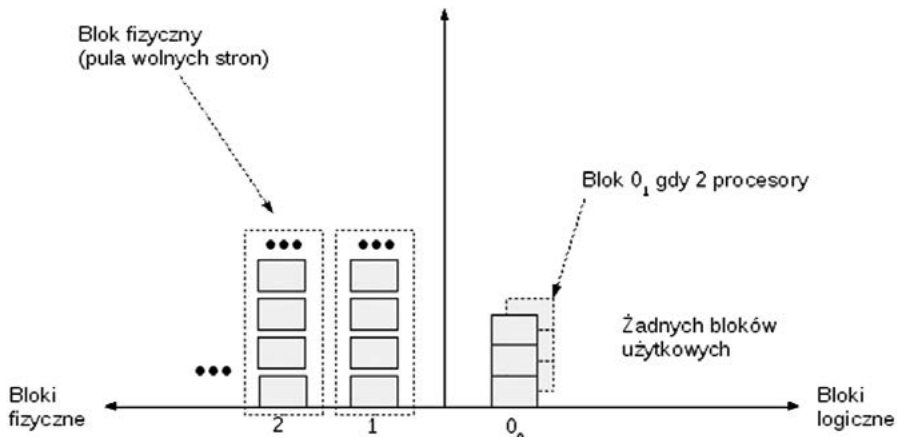
Ostatecznie schemat blokowy maszyny był następujący:



Rysunek 4. Schemat blokowy Mera-400 (konfiguracja max)

### 3.2. Organizacja pamięci

Tu zmiany były największe. Maszyna Mera-400 została wyposażona w doskonałą dynamiczną alokację, zarówno w segmentację, jak i stronicowanie. Stan początkowy pamięci po włączeniu zasilania jest przedstawiony na rysunku 5.



Rysunek 5. Pamięć Mera-400 – stan początkowy

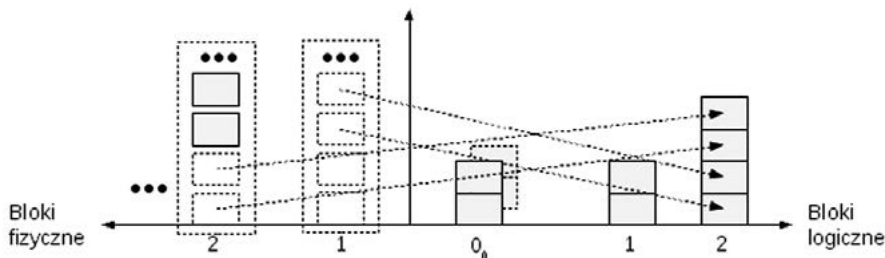
Po lewej stronie jest pamięć fizyczna. Składa się ona z bloków, z których każdy może zawierać do 16 stron o pojemności 4 kółbów każda. Pamięć fizyczna nie reaguje na żadne

rozkazy odczytu/zapisu, jest tylko pulą wolnych stron. I o to chodziło, bo programy nie działają na pamięci fizycznej, tylko na pamięci logicznej. Po prawej stronie jest pamięć logiczna, która zawierała 2 lub 3 strony „zaluowane” na stałe w bloku 0, aby system operacyjny mógł zacząć działać.

Pierwszą czynnością systemu operacyjnego było zbadanie pamięci fizycznej, aby mógł wiedzieć, czym dysponuje. Następnie mógł sobie dobrać odpowiednią liczbę stron pamięci, tyle, ile potrzebował do własnego działania.

W czasie normalnej pracy system operacyjny tworzy i usuwa procesy. Do utworzenia procesu system operacyjny musi utworzyć dla niego blok (segment) w pamięci logicznej. Każda strona pamięci była wyposażona w mały kawałek pamięci asocjacyjnej, do której system operacyjny specjalnym rozkazem mógł wpisać numer bloku logicznego i początkowy adres strony logicznej (to znaczy 4 najstarsze bity adresu). Dopiero tak zainicjalizowana strona zaczynała reagować na rozkazy odczytu/zapisu. Rysunek 6 pokazuje stan, w którym utworzone zostały dwa procesy. Strony pamięci, które były składane w blok logiczny, nie musiały pochodzić z tego samego bloku fizycznego.

Po zakończeniu procesu system operacyjny „rozmontowywał” jego blok pamięci i zwracał strony do puli wolnych stron.



Rysunek 6. Pamięć Mera-400 – dwa bloki użytkowe (procesy)

### 3.3. Zarządzanie dostępem do szyny

Opracowany został nowy rozproszony układ zarządzania dostępem (patent Andrzeja Karczmarewicza i Elżbiety Jezierskiej-Ziemkiewicz), zrealizowany jako ciąg semaforów, który działał doskonale.

### 3.4. Rejestry maszyny

Dodane zostały nowe rejestry zwiększające możliwości maszyny:

- Stos systemowy i systemowy wskaźnik stosu. W K-202 brak stosu systemowego utrudniał realizację wieloprotocowego systemu operacyjnego.
- Wektor stanu procesu. Składał się z zawartości licznika rozkazów, rejestru stanu procesora (R0) oraz ewentualnych rejestrów uniwersalnych procesora i był przechowywany automatycznie na stosie systemowym przy przełączaniu kontekstu (np. przy przerwaniu). Ułatwiała to realizację wieloprotocowego systemu operacyjnego.

- Rejestr masek przerwań dla urządzeń zewnętrznych. W K-202 była 1-bitowa maska blokująca przerwanie ze wszystkich urządzeń. W Mera-400 rejestr masek miał 10 bitów, co umożliwiało znacznie łatwiejsze i bardziej selektywne zarządzanie urządzeniami.
- Rejestr zegara czasu rzeczywistego. Jego dodanie umożliwiało aplikację *real-time*.

### 3.5. Rozkazy maszyny

Najważniejszą zmianą było zlikwidowanie nieszczęsnego bitu W. Zwolniony bit został dołączony do kodu operacji, co podwoiło liczbę możliwych kombinacji i pozwoliło zwiększyć liczbę rozkazów z 94 do ponad 130.

Chyba najważniejszymi z nowych rozkazów były rozkazy wykonujące operacje na argumentach podwójnej precyzji.

Dodane zostały nowe rozkazy służące do dynamicznej alokacji pamięci. Dodane zostały rozkazy działania na ciągach bajtów, z bajtową adresacją. Rozszerzony został zbiór rozkazów służących do komunikacji międzyprocesorowej.

Rozkazy K-202 zostały zachowane, co umożliwiło przejście całego oprogramowania K-202.

### 3.6. Podsumowanie

Mera-400 była doskonałą maszyną. Jak na minikomputer była nadspodziewanie mocna. Na przykład na Politechnice Gdańskiej jedna maszyna obsługiwała cały wydział, na którym, na kilkudziesięciu terminalach, studenci robili swoje ćwiczenia.

Wyprodukowano ponad 600 egzemplarzy w zakładach ERA, które mogły produkować ją nadal, ale zamknęły produkcję, bo chciały robić coś innego.

## 4. Maszyna MX-16

Kiedy ERA zamknęła produkcję Mera-400, zespół postanowił kontynuować prace. Były one prowadzone w firmie polsko-polonijnej AMEPOL. Powstała nowa maszyna nazwana MX-16, która była kontynuacją Mera-400.

Jako procesor zachowany został procesor Mera-400, który uważaliśmy za wystarczająco dobry do czasu opracowania następnego. Tym następnym miał być procesor według projektu SOLID, ale ogłoszenie stanu wojennego zamknęło prace, więc nie będę o tym pisać.

Natomiast wszystko, co było wokół procesora, zostało zmienione.

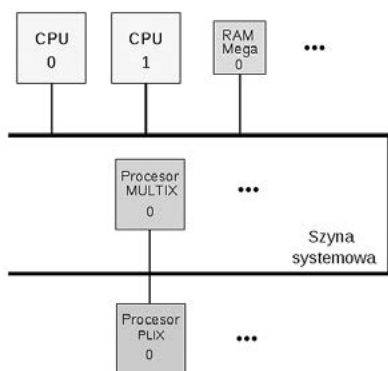
### 4.1. Pamięć (Mega)

Zespół nadążał za rozwojem technologii, więc skoro tylko były dostępne układy scalone pamięci, można było zrezygnować z pamięci ferrytowych (choć mogły być nadal używane). Skonstruowany został moduł pamięci półprzewodnikowej zawierający 1 MB w jednym bloku fizycznym, a później 2 MB w jednym bloku fizycznym.

Kanały zewnętrzne dla urządzeń zostały zaprojektowane jako mikroprocesorowe podsystemy peryferyjne zbudowane na Intel 8085. Pozwoliło to wyposażyć je w większą inteligencję i odciążać procesor oraz uprościć architekturę maszyny.

## 4.2. Architektura

Schemat blokowy maszyny przedstawiam na rysunku 7.



Rysunek 7. Schemat blokowy MX-16

Na schemacie:

- RAM Mega – oznacza moduł pamięci o pojemności 1 MB/2 MB;
- MULTIX – oznacza peryferyjny podsystem multipleksorowy znakowy z możliwością dołączania do 256 urządzeń zewnętrznych, prowadzący równoczesną transmisję z/do wielu urządzeń do niego dołączonych;
- PLIX – oznacza peryferyjny podsystem urządzeń pamięciowych; jest to procesor selektorowy z jedną transmisją aktywną w danym momencie, poprzez który pracowały dyski typu Winchester, pamięci taśmowe itp.

Oprócz tych modułów opracowany został procesor pomiarowy IEC. Procesor ten wyposażony był w dwuwęściową pamięć, która była dołączana do szyny głównej systemu jak normalny moduł pamięci operacyjnej. Zamierzano go wykorzystać do sterowania urządzeniami pomiarowymi z interfejsem IEC i dla aparatury medycznej. W dalszej perspektywie było dołączanie aparatury graficznej i innych urządzeń wymagających przesyłania dużych zbiorów danych.

## 4.3. Podsumowanie

Wszystkie własności Mery-400 zostały zachowane, a podsystem urządzeń peryferyjnych został wzbogacony.

## 5. Oprogramowanie

Ten artykuł dotyczy sprzętu, a nie oprogramowania. Niemniej należy tu umieścić parę uwag na temat oprogramowania.

Oprogramowanie K-202, ponieważ było projektowane od zera, na początku było ubogie. Zespół programistów był nieliczny. Zmieniło się to, gdy komputery zaczęły trafiać do użytkowników. Klub użytkowników włączył się do prac nad rozwijaniem oprogramowania, co było kontynuowane w czasie Mery-400 i MX-16.

Ostatecznie oprogramowanie było dosyć bogate i zawierało:

- pierwszy System Operacyjny SOK1 i assembler ASSK opracowany w Zakładzie Mini komputerów ERA przez zespół w składzie Teresa Pajkowska i Karol Doktor. Dla Mery-400 były to SOM1 i ASSM;
- pierwszy wielodostępny i wieloprogramowy systemem operacyjny CROOK rozwijany na Politechnice Gdańskiej<sup>1</sup>;
- języki BASIC i FORTRAN IV opracowane przez grupę programistów z Politechniki Poznańskiej pod kierunkiem Janusza Gocałka;
- z innych języków oprogramowania, które były w różnych fazach zaawansowania, należy wymienić CSL, CEMMA, BICEPS, MOST-2, COMIT;
- kompilator języka C – Politechnika Gdańska;
- kompilator języka LOGLAN – II UW.

Opracowano kilka systemów operacyjnych (oprócz wymienionych wyżej):

- SOM 3 – Wojciech Szanser, Jerzy Skolimowski – ZDM IMM,
- SOM 5 – Stanisław Chrobot – PIAP (system specjalizowany dla automatyki),
- SOM 7 – Piotr Findeisen, Paweł Gburzyński – II UW.

Klub użytkowników opracował mnóstwo pakietów użytkowych. Było to w sumie dosyć bogate oprogramowanie.

## 6. Zakończenie

Opisane tu maszyny K-202, Mera-400 i MX-16 były dobrymi maszynami. Jednak patrząc z perspektywy czasu, wydaje się nam, że to, czy były lepsze od innych, czy nie, nie było takie ważne. To nie miało wielkiego znaczenia i wszystkie przepychanki na ten temat też są bez znaczenia. Najważniejsze było rozwijanie polskiej myśli technicznej.

Żadne zakupy obcych licencji czy produktów nie zastąpią rozwijania własnej techniki. Rozwijając własne projekty, szkoli się i zdobywa doświadczenie kadra inżynierów

---

1 Z. Czerniak, K-202, MERA-400 i CROOK. *Krótką historią pewnego projektu*, [w:] *Spółczesność i informacyjność. Krok naprzód, dwa kroki wstecz*, red. P. Sienkiewicz, J.S. Nowak, Katowice, Polskie Towarzystwo Informatyczne 2008, s. 457–462.

i programistów, rośnie ogólny poziom kompetencji. Kupując obcy produkt czy licencję, można mieć w pierwszym momencie doraźny zysk, ale na dłuższą metę jest to szkodliwe, bo własny rozwój zamiera (a jak to jest w branży samochodowej?).

Wystarczy spojrzeć na przykłady IMM i Elwro (tu będzie mała dygresja).

## 6.1. IMM

Od swojego powstania do środkowych lat 60. IMM rozwijał własną linię maszyn i można powiedzieć, że radził sobie dobrze, bo potrafił rozwiązać wszystkie problemy, które się pojawiały.

Potrzebne były pamięci ferrytowe? W IMM został utworzony dział pamięci ferrytowych, który opracował metodę wypiekania rdzeni i mógł produkować własne matryce ferrytowe. Potrzebne były pamięci taśmowe? Utworzony został dział pamięci taśmowych, który opracował pamięci taśmowe serii PT. Służyły one później również innym maszynom, nie tylko IMM. Potrzebne były pamięci bębnowe (pamięci dyskowe nie były jeszcze znane)? Utworzony został dział pamięci bębnowych, który opracował dobrze działające pamięci bębnowe.

Krótko mówiąc, IMM był samowystarczalny, ponieważ potrafił opracować wszystko, co było mu potrzebne, i nad tym wszystkim panował.

Przy okazji tego rozwoju została wyszkolona ogromna rzesza inżynierów i programistów.

Wszystko zespuło się w momencie, gdy IMM został zmuszony do zaniechania rozwoju własnych projektów i kopiowania maszyny IBM 360. Był to koniec własnego rozwoju i, w dalszej perspektywie, zmierzch IMM. Obecnie wegetuje on, żyjąc z wynajmowania pomieszczeń swojego budynku różnym firmom i organizowania kursów szkoleniowych.

## 6.2. Elwro

Od samego początku Elwro rozwijało własne maszyny Odra, począwszy od pierwszego modelu 1000, aż do 1204. I bardzo dobrze.

Następnie uznano, że oprogramowanie jest zbyt ubogie, i zaczęto rozglądać się za kontrahentem, od którego można by je kupić.

Firma ICT była firmą elektroniczną o bogatych tradycjach sięgających jeszcze czasów przedwojennych. W 1960 r. rozpoczęła ona produkcję maszyn serii ICT 1900. Z tą firmą zaczęło pertraktować Elwro. Zanim te pertraktacje się zakończyły, firma ICT została kupiona przez firmę ICL, która przejęła ją z całym dobrodziejstwem inwentarza i z systemem przemianowanym na ICL 1900. Zatem Elwro podpisało ostatecznie umowę z firmą ICL. Umowa dotyczyła zakupu oprogramowania z prawem sprzedaży. Natomiast Elwro musiało samo zbudować maszynę, zgodnie z *Principles of Operation* firmy ICL.

Firma ICL przesłała do Elwro taśmy magnetyczne z kompletnym oprogramowaniem systemowym i użytkowym, zatem w pierwszym momencie Elwro miało doraźny zysk, bo otrzymało bogate oprogramowanie.

Maszyna musiała być zbudowana dokładnie według angielskiej specyfikacji, inaczej nie działałoby oprogramowanie. Nie było tu miejsca na rozwijanie własnych pomysłów.

W tym momencie skończył się rozwój własnych projektów w Elwro. Zostało ono z produktem, którego koncepcja pochodziła z 1960 r. i z każdym rokiem coraz bardziej się starzała. Tak zaczął się zmierzch Elwro.

Nie możemy oprzeć się wrażeniu, że gdyby pieniądze wydane na zakup z ICL zostały przeznaczone na własny rozwój, być może Elwro wyszłoby na tym lepiej i przeżyło.

Jeżeli ktoś nie jest jeszcze przekonany, że warto rozwijać własne, a nie kopiować lub kupować cudze, przedstawię jeszcze jeden przykład.

Kiedy Dijkstra opublikował swoją sławną pracę, w której sformułował koncepcję semafora (środek lat 60.), zrobiła ona dużo szumu. Był to pierwszy oparty na porządnej podstawie teoretycznej mechanizm programowy służący do synchronizacji i ochrony międzyprocesowej. Określała ona również warunek konieczny, aby to działało. Było nim istnienie w maszynie operacji atomowej i niepodzielnej, którą nazwano rozkazem *Test-and-Set* (nie będę wyjaśniał szczegółów, bo nie jest to miejsce na wykład z podstaw informatyki). W IMM publikacja ta również zrobiła dużo szumu. Projektanci natychmiast zaimplementowali odpowiedni rozkaz w maszynie, a koledzy programiści natychmiast uwzględnili go w systemie operacyjnym. Było to możliwe tylko dlatego, że IMM w pełni panował nad swoją maszyną.

W Elwro, przy oprogramowaniu kupionym od ICL, coś takiego było niemożliwe. Chociaż Elwro samo opracowało sprzęt i mogło zaimplementować taki rozkaz, byłoby to zupełnie bezużyteczne, bo kupiony system operacyjny go nie znał i nie umiałby go użyć (system był kupiony oczywiście w postaci binarnej – pojęcie *open source* było nieznane).

Miłym aspektem rodziny maszyn K-202/Mera-400/MX-16 było to, że wszystkie one były rozprowadzane w kraju i służyły z pożytkiem krajowej gospodarce.

Maszyny SM natomiast prawie w całości szły do Związku Radzieckiego. Zakład produkujący ERA był oczywiście zadowolony, że ma eksport. Ale w skali kraju? Z ZSRR kupowano procesory (skopiowane z PDP 11/40), z nich montowany był system, dokładane były różne polskie urządzenia (pamięci zewnętrzne, drukarki itd., co ogołacało z nich rynek krajowy), po czym gotowe maszyny wysyłano do ZSRR. Niemożliwe było kupno w kraju drukarki, bo było ich za mało, ale można było kupić system komputerowy zawierający drukarkę (dziwoląg gospodarki socjalistycznej). Zakłady kupowały więc system oparty na Momiku (bo najtańszy), z drukarką, po czym wyrzucały Momika, a drukarkę dołączały tam, gdzie trzeba. To nie jest dowcip. Rozmawialiśmy z kimś, kto pracował jako serwisant systemów Momik i opowiadał, że wezwany do naprawy drukarki sprzedanej z Momikiem zastawał drukarkę dołączoną do innej maszyny, a Momik leżał gdzieś w kącie. Jeżeli ktoś mnie zapewni, że „towarzysze radzieccy” płacili nam więcej za kompletne systemy niż my im za te procesory, to przestanę podejrzewać, że był to raczej drenaż gospodarczy kraju przez Rosjan niż korzystny eksport.

W powyższej uwadze proszę nie doszukiwać się jakiejś osobistej wycieczki w stronę Momika. Nic przeciw niemu nie mamy. W kraju były potrzebne wszystkie maszyny, także te najmniejsze.

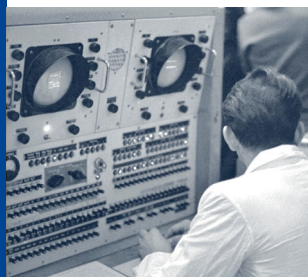
Nie bez powodu tak podkreślamy polską myśl techniczną. Przecież Polskie Towarzystwo Informatyczne, mające taki przymiotnik w nazwie, ma z definicji propagować polską

myśl techniczną w informatyce. Ten aspekt powinien być mocniej zaznaczany na forum, ponieważ niektóre z maszyn produkowanych w Polsce były owocem polskiej myśli technicznej, a w innych była to głównie myśl angielska, amerykańska lub jeszcze inna.



W czwartek, 23 grudnia 1948 r., w gmachu Fizyki Doświadczalnej przy ul. Hożej w Warszawie, z inicjatywy wybitnego topologa, profesora Uniwersytetu Warszawskiego, dyrektora świeżo organizowanego Państwowego Instytutu Matematycznego (PIM) Kazimierza Kuratowskiego spotkało się kilku przyszłych pionierów elektronicznych maszyn liczących. Byli to, oprócz inicjatora spotkania, profesor Andrzej Mostowski – matematyk zajmujący się głównie logiką matematyczną i algebrą, doktor Henryk Greniewski – matematyk i logik, a także trzech młodzi inżynierowie po studiach na Politechnice Gdańskiej – Krystyn Bochenek, Leon Łukaszewicz i Romuald Marczyński, późniejsi profesorowie.

Profesor Kuratowski podzielił się z zebranymi swoimi wrażeniami z naukowego pobytu w USA. Był pod wrażeniem elektronicznych maszyn liczących, które widział za oceanem, i uważał, że chociaż jedna taka maszyna powinna być zbudowana w naszym kraju. W rezultacie tego spotkania zapadła decyzja o powołaniu w ramach PIM Grupy Aparatów Matematycznych (GAM) w wyżej wymienionym składzie, pod kierunkiem Henryka Greniewskiego.



ISBN 978-83-60810-87-3

