



Systemy czasu rzeczywistego

Zastosowania systemów czasu rzeczywistego w wojskowości

Autor: Michał Jereczek

Spis treści

1	Wprowadzenie	2
1.1	Podstawowe pojęcia	2
1.1.1	Klasyfikacja	2
2	Historia	3
2.1	Projekt Whirlwind	3
2.1.1	Pamięć ferrytowa	3
2.1.2	Whirlwind assembler	3
2.2	Projekt SAGE	4
2.2.1	Univac 1103A	4
3	Rozwój języków programowania	4
3.1	Fortran	5
3.1.1	Wady	5
3.1.2	Zalety	5
3.1.3	Przykładowy program	5
3.2	CMS-2	6
3.2.1	Naval Tactical Data System	6

1 Wprowadzenie

1.1 Podstawowe pojęcia

Aby móc rozpocząć rozważania na temat zastosowania systemów czasu rzeczywistego w wojskowości niezbędnym jest wyjaśnienie podstawowych pojęć oraz koncepcji związanych z tego typu systemami.

1. System - W najogólniejszym znaczeniu tego zagadnienia, można powiedzieć, że system jest przyporządkowaniem pewnego zbioru wejść do pewnego zbioru wyjść.
2. System czasu rzeczywistego - Jest to system, dla którego krytycznym zagadnieniem jest czas. Poprawność działania takiego systemu jest ściśle zależna od poprawnego odebrania danych wejściowych w ściśle określonych ramach czasowych. System czasu rzeczywistego powinien zapewnić przewidywalne i możliwie bezpieczne zachowanie wobec niepowodzeń.

1.1.1 Klasyfikacja

Systemy czasu rzeczywistego można klasyfikować ze względu na skutki niezamieszczenia się w ramach czasowych, ogólnie przyjęto podział na trzy kategorie:

1. System o łagodnych ograniczeniach czasowych (ang. soft) - Jest to system, w którym przegapienie nawet wielu ram czasowych nie skutkuje poważnymi konsekwencjami, co najwyżej spadkiem wydajności i frustracją użytkowników. W zastosowaniach militarnych szczególnie ważne są systemy komunikacji, gdzie podstawową funkcjonalnością nie jest jej jakość, lecz niezawodność. Przykładem jest system audio-video, gdzie strata nawet ciągu danych nie sprawia, że cel (odebranie i zrozumienie komunikatu) nie zostanie osiągnięty (przekaz straci tylko na jakości).
2. System o mocnych ograniczeniach czasowych (ang. firm) - System, w którym przegapienie ciągu okien czasowych nie spowoduje awarii systemu, jednak spowodowane tym opóźnienie w reakcji systemu może doprowadzić do katastrofy. Wojskowe systemy szybkiego reagowania potrzebują predykcji zmian pogodowych, jest to niezwykle ważne, do poprawnego zaplanowania misji. Przykładowy system, który analizuje dane z stacji meteorologicznych w wyniku zbyt dużej ilości informacji do przetworzenia poprawnie przewidzi nastąpienie gwałtownej burzy, jednak przez opóźnienia informacja ta może zostać podana zbyt późno aby była możliwa odpowiednia reakcja.
3. System o ostrych ograniczeniach czasowych (ang. hard) - Klasa systemu, gdzie nie zmieszczenie się w określonych ramach czasowych nawet wobec jednej operacji jest katastrofalna w skutkach. Tego typu systemy są szczególnie ważne dla wojska. Przykładem są wszelakiego rodzaju systemy kontrolujące pociski rakietowe, gdzie nawet minimalne opóźnienie między naciśnięciem przycisku do wystrzału, a samą operacją wystrzelenia może doprowadzić do nie trafienia w cel.

2 Historia

2.1 Projekt Whirlwind

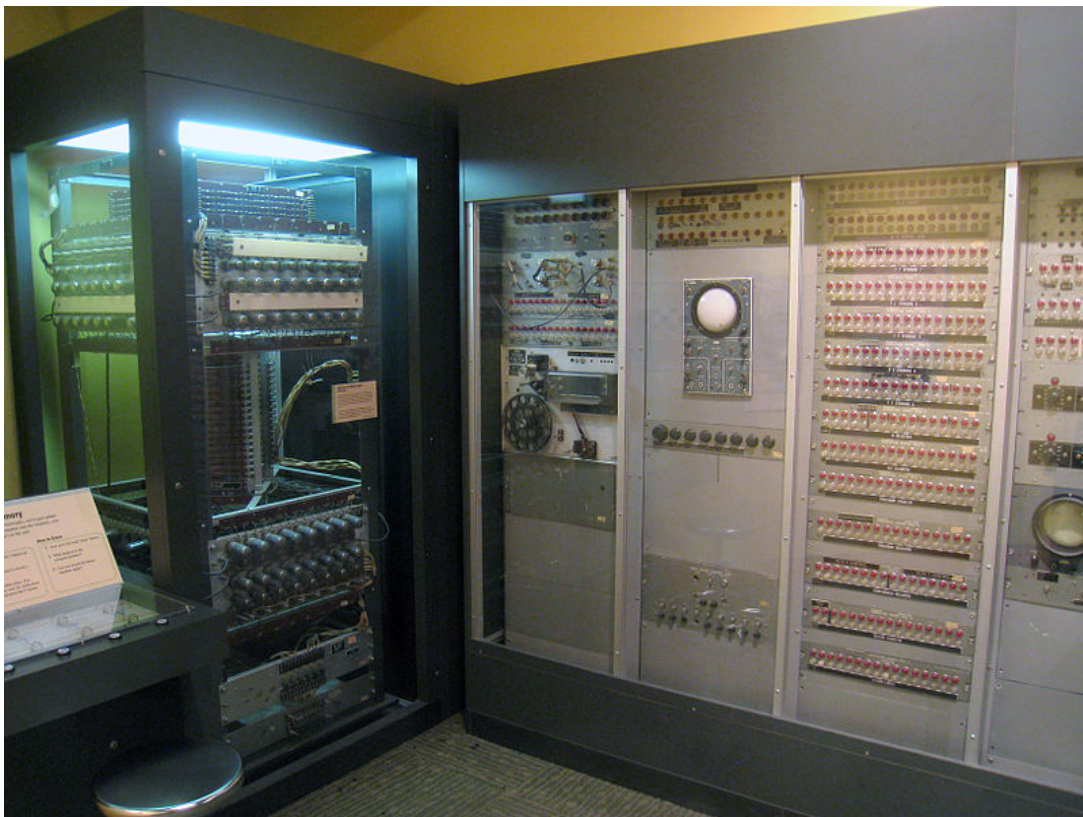
Niepewnym jest gdzie i kiedy narodził się termin *systemów czasu rzeczywistego*, jednak prawdopodobne jest, że pierwszy raz został on użyty w projekcie *Whirlwind* (pl. Trąba Powietrzna) w 1947 roku. Whirlwind to komputer wykorzystujący lampy elektronowe, który miał służyć do symulowania lotów dla załogi samolotów bombowych. Jest to jeden z pierwszych komputerów, który umożliwiał przetwarzanie równoległe w czasie rzeczywistym.

2.1.1 Pamięć ferrytowa

Whirlwind został opracowany w czasach zimnej wojny przez MIT dla marynarki wojennej Stanów Zjednoczonych. Innowacyjnym w projekcie było zastosowanie pamięci opartej o rdzeń ferrytowy, dzięki czemu dostęp do pamięci był praktycznie natychmiastowy (prekursor współczesnych pamięci RAM).

2.1.2 Whirlwind assembler

W trakcie tworzenia tego projektu utworzony został również specjalny język programowania *Whirlwind assembler*, który opisano jako *algebraiczny kompilator do uproszczenia kodowania* (ang. algebraic compiler to simplify coding), był to jeden z pierwszych języków wysokiego poziomu stworzony jeszcze na 10 lat przed Fortranem.



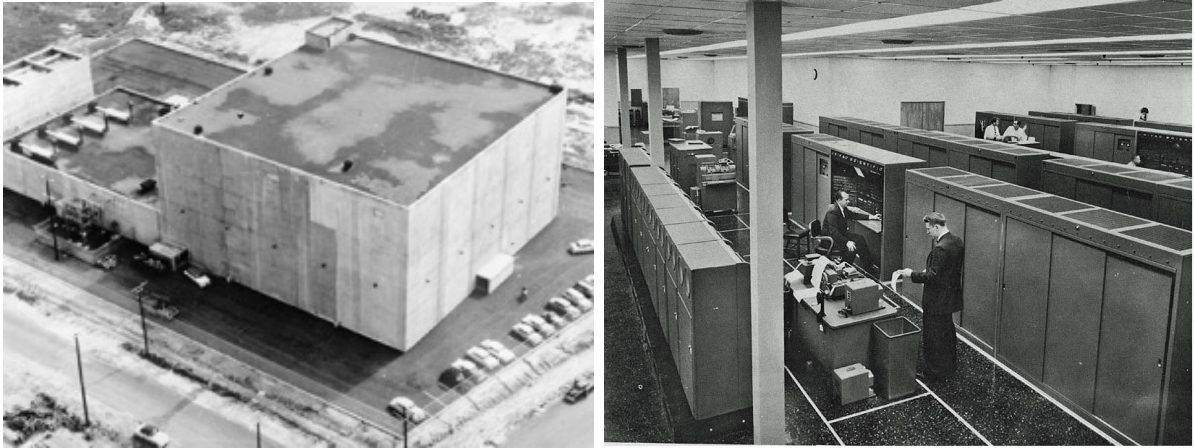
Rys. 1: Zdjęcie komputera Whirlwind I. Z lewej strony można zobaczyć pamięć ferrytową oraz stanowisko operatora. Źródło: https://en.wikipedia.org/wiki/Whirlwind_I

2.2 Projekt SAGE

W 1957 roku (10 lat po projekcie Whirlwind) swoje piętno w historii Sił Powietrznych Ameryki odcisnął projekt SAGE: Semi-Automatic Ground Environment. Inżynierowie IBM w współpracy z MIT stworzyli olbrzymi system komputerów, którego zadaniem było między innymi koordynowanie i przetwarzanie danych przyjmowanych z wielu systemów radarowych. System ten w rezultacie był w stanie przedstawić wizualizację ruchu powietrznego w badanych obszarach.

2.2.1 Univac 1103A

Zadania wykonywane przez system SAGE nie były by możliwe bez użycia nowoczesnych na tamte czasy komputerów IBM'u *Univac Scientific 1103A*, które obsługiwały przetwarzania asynchroniczne. Komputery potrzebne do pracy systemu zajmowały specjalnie postawiony budynek o powierzchni 1.4 hektara.



Rys. 2: Budynek w którym znajdował się SAGE oraz komputer Univac.

Źródła:

https://en.wikipedia.org/wiki/Semi-Automatic_Ground_Environment

https://en.wikipedia.org/wiki/UNIVAC_1103

3 Rozwój języków programowania

Rozwój systemów czasu rzeczywistego wymusił potrzebę stworzenia nowych, wysokopoziomowych języków oprogramowania, które ułatwiły by pracę inżynierom. U.S. Army w swoich systemach stosowało głównie język *Fortran*, U.S. Navy natomiast *CMS-2*, gdy w tym samym czasie Air Force wykorzystywało *JOVIAL*.

Tak duży podział wśród tych jednostek doprowadził do tego, że w latach siedemdziesiątych Departament Obrony Stanów Zjednoczonych zlecił opracowanie języka Ada, który miał być standardowo stosowany przez nowe systemy tworzone na zlecenie wyżej wymienionych jednostek.

3.1 Fortran

Fortran(Formula Translator) jest najstarszym z języków wysokiego poziomu stosowanym współcześnie w systemach czasu rzeczywistego. Stworzony przez IBM dominował na arenie programów służących do analizowania zmian pogodowych, analizach fizycznych i chemicznych oraz innych obszarach, gdzie priorytetowa była wydajność i responsywność.

3.1.1 Wady

Fortran został stworzony około roku 1955 i jego ówczesna wersja nie wspierała tak ważnych dla systemów czasu rzeczywistego funkcji jak *przerwanie* i *kolejkowanie zdarzeń*, dlatego musiał współpracować razem z dużymi porcjami kodu assemblera. Język ten nie posiadał również mechanizmów dynamicznego alokowania pamięci oraz rekurencji.

3.1.2 Zalety

Fortran pozwalał na pisanie konstrukcji typu if-then-else, co umożliwiało pisanie na znacznie wyższym poziomie abstrakcji niż dotychczas. Język ten powstawał w czasach małych i wolnych systemów wbudowanych, co przyczyniło się do zaprojektowania go w taki sposób, aby można w nim było pisać wydajne programy przy małej mocy obliczeniowej.

3.1.3 Przykładowy program

Przykładowy program, który losuje numer pomiędzy 0 a 1 i sprawdza czy mieści się w określonych przedziałach.

```
program xif
  implicit none
  real :: x
  real, parameter :: x1 = 0.3, x2 = 0.6
  call random_seed()
  call random_number(x)
  if (x < x1) then
    print*,x,"<",x1
  else if (x < x2) then
    print*,x,"<",x2
  else
    print*,x,">=",x2
  end if
end program xif
```

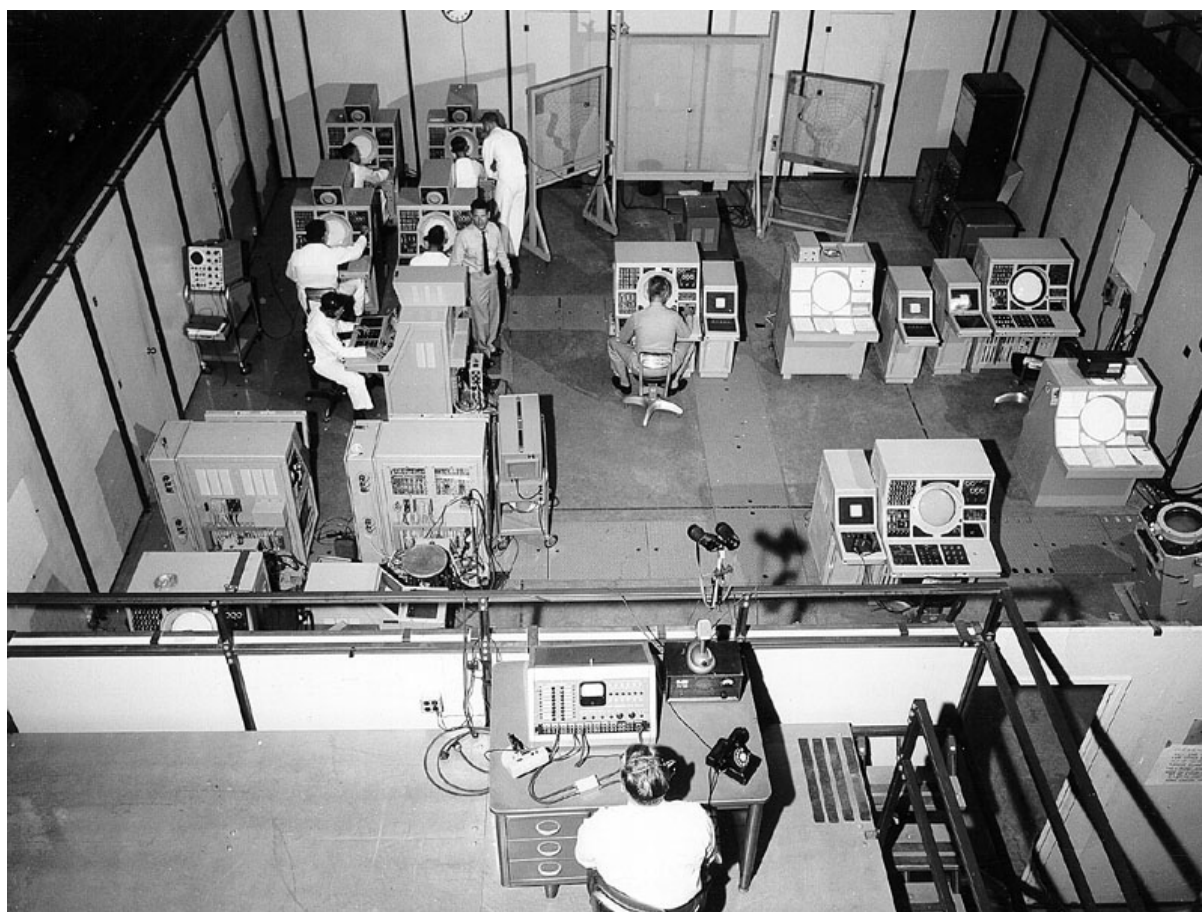
źródło: https://en.wikibooks.org/wiki/Fortran/Fortran_control

3.2 CMS-2

CMS-2 (Compiler Monitor System 2) był językiem stworzonym specjalnie na potrzeby U.S. Navy. Wydany około roku 1968 (ponad dekadę po Fortranie) stosowany jest w systemach wbudowanych.

3.2.1 Naval Tactical Data System

CMS-2 stosowany jest chociażby w systemie NTDS (Naval Tactical Data System), który wspiera misje bojowe do dnia dzisiejszego. System ten kolekcjonuje i procesuje ważne informacje aby przedstawić aktualną sytuację, dzięki czemu dowódca sił zbrojnych może szybko podejmować strategiczne decyzje. Okręty wojenne z systemem NTDS komunikują się wzajemnie, co pozwala na zbudowanie szczegółowej mapy taktycznej.



Rys. 3: Zdjęcie przedstawiające testy systemu NTDS w zainscenizowanym pokładowym pomieszczeniu operacyjnym CIC (Combat Information Center).

Źródło: https://en.wikipedia.org/wiki/Naval_Tactical_Data_System

Literatura

- [1] Phillip A. Laplante, *Real-time systems design and analysis*. IEEE PRESS, Third Edition, 2004.
- [2] Praca zbiorowa pod redakcją Leszka Trybusa i Sławomira Samoleja, *Projektowanie, analiza i implementacja systemów czasu rzeczywistego*. Wydawnictwo Komunikacji i Łączności, Warszawa, 2011.
- [3] Antoni M. Donigiewicz, Krzysztof Łuczak, Wojciech Sałata, Andrzej Stasiak, *Systemy czasu rzeczywistego '99 : Zastosowania Wojskowe*. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1999.
- [4] https://pl.wikipedia.org/wiki/System_czasu_rzeczywistego
- [5] https://en.wikipedia.org/wiki/Whirlwind_I
- [6] http://ethw.org/Magnetic-Core_Memory
- [7] https://en.wikipedia.org/wiki/Semi-Automatic_Ground_Environment
- [8] https://en.wikipedia.org/wiki/UNIVAC_1103
- [9] [https://en.wikipedia.org/wiki/CMS-2_\(programming_language\)](https://en.wikipedia.org/wiki/CMS-2_(programming_language))
- [10] <http://www.thefreedictionary.com/naval+tactical+data+system>
- [11] <http://mil-embedded.com/articles/ntds-navy-platforms-worldwide/>
- [12] https://en.wikipedia.org/wiki/Naval_Tactical_Data_System