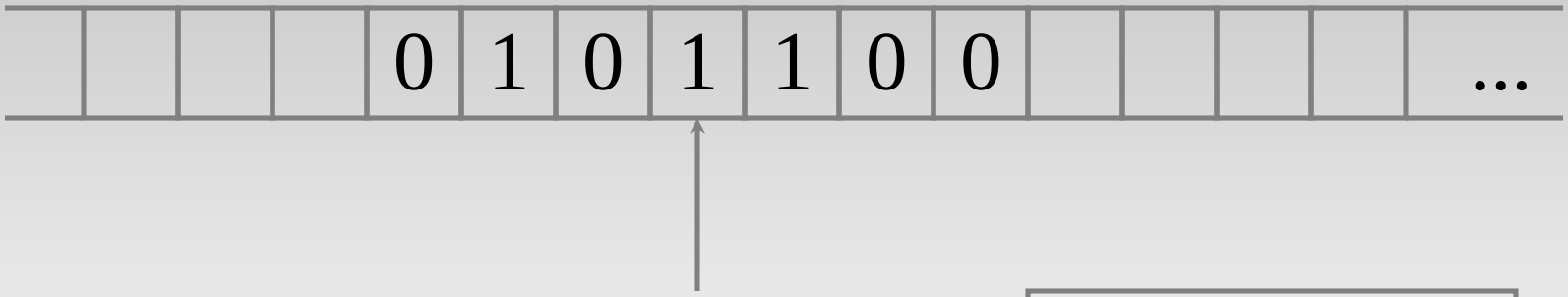
Budowa i zasady działania komputera (1)

Ojcowie informatyki

MASZYNA TURINGA-POSTA



Język Turinga-Posta

000	Drukuj 0
001	Drukuj 1
010	Idź w lewo
011	Idź w prawo
1010...01 (i zer)	Idź do kroku i jeśli przeczytałeś 0
1101...10 (i jedynek)	Idź do kroku i jeśli przeczytałeś 1
100	Zatrzymaj się

1 ← przed programem 111 ← po programie

Program podwajania

1. DRUKUJ 0
2. IDŹ W LEWO
3. IDŹ DO KROKU 2 JEŚLI PRZECZYTAŁEŚ 1
4. DRUKUJ 1
5. IDŹ W PRAWO
6. IDŹ DO KROKU 5 JESLI PRZECZYTAŁEŚ 1
7. DRUKUJ 1
8. IDŹ W PRAWO
9. IDŹ DO KROKU 1 JEŚLI PRZECZYTAŁEŚ 1
10. ZATRZYMAJ SIĘ

Jak zadziała program dla poniższej konfiguracji?

... 0 0 1 1 0 0 ...

↑

Program, który może się nie zatrzymać:

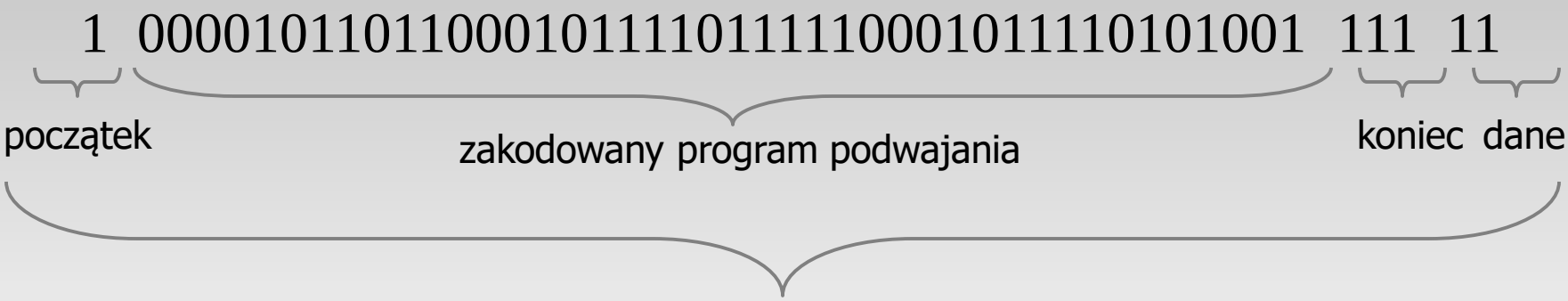
1. Idź w prawo
2. Idź do kroku 1 jeśli przeczytałeś 0
3. Zatrzymaj się

PROBLEM STOPU jest nierozstrzygalny

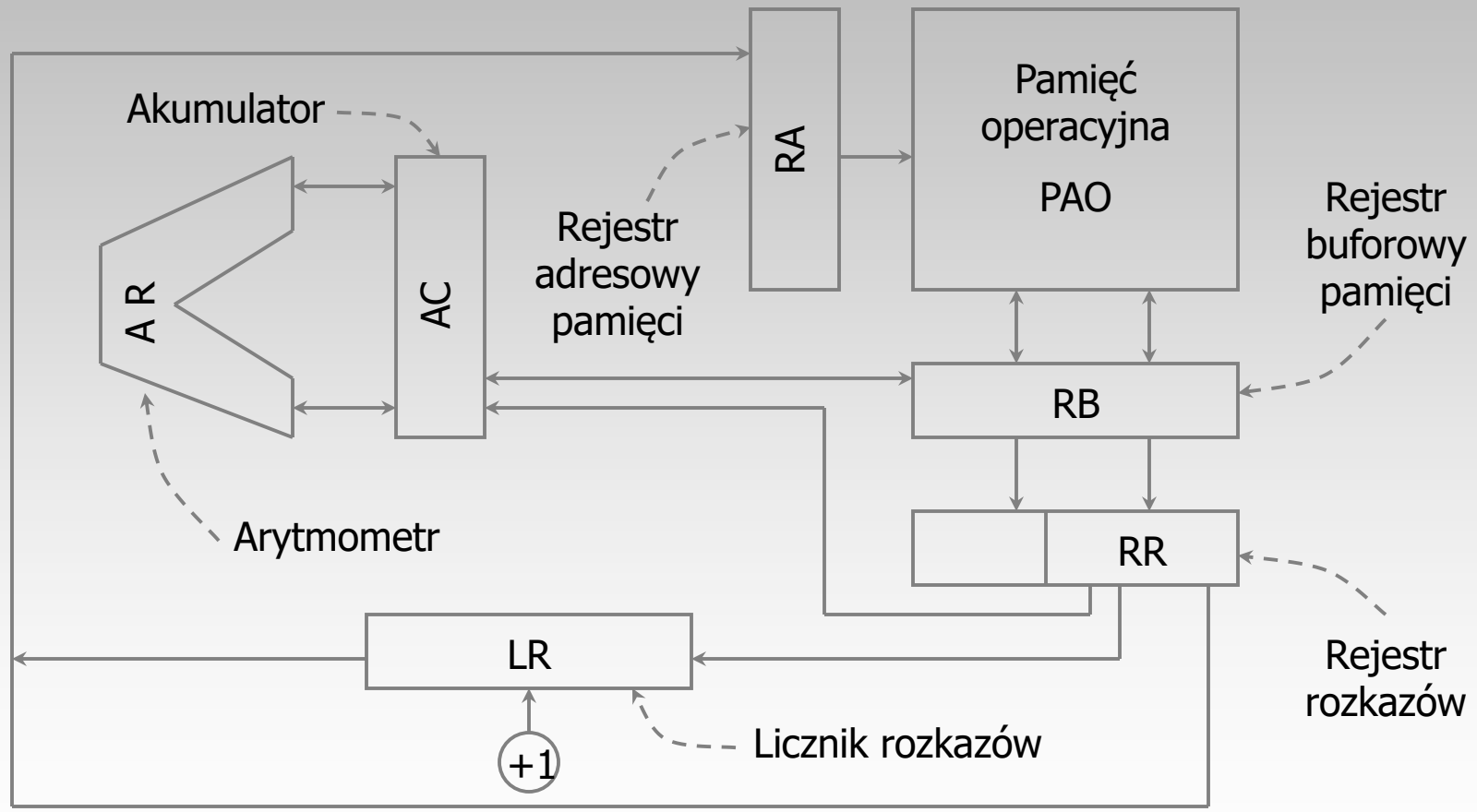
W ogólnym przypadku nie istnieje procedura rozstrzygająca, czy program zawsze się zatrzyma.

UNIWERSALNY PROGRAM TURINGA-POSTA

to program w języku Turinga-Posta, który potrafi wykonać dowolny program w tym języku dla dowolnych danych. Traktuje on zarówno program do wykonywania, jak i jego dane, jako swoje dane.



KLASYCZNY SCHEMAT KOMPUTERA



Budowa i zasady działania komputera (1)

* Pamięć operacyjna (PAO) składa się ze słów o stałej długości ponumerowanych od 0 wzwyż. Słowa są ciągami bitów (komórek, w których można zapisać 0 lub 1).

Adres w PAO = Nr słowa

* PAO pracuje w trybie zapisu lub odczytu.

Zapis oznacza umieszczenie w PAO zawartości RB pod adresem znajdującym się w RA z wymazaniem poprzedniej wartości.

Odczyt oznacza umieszczenie w RB zawartości PAO znajdującej się pod adresem umieszczonym w RA.

Budowa i zasady działania komputera (1)

* Rozkaz ma strukturę:

<kod operacji> <sposób adresacji> <argument>

Sposoby adresacji:

a) natychmiastowy (n) – argument jest liczbą, która weźmie udział bezpośrednio w operacji,

b) bezpośredni (b) – argument jest adresem w PAO liczby która weźmie udział w operacji

c) pośredni (p) – argument jest adresem w PAO liczby, która jest adresem liczby, która weźmie udział w operacji

* Wszelkie operacje arytmometru (AR) są wykonywane w akumulatorze (AC) i ich wynik też tam pozostaje.

Budowa i zasady działania komputera (1)

* Komputer wykonuje kolejno rozkazy znajdujące się w słowie o adresie znajdującym się w liczniku rozkazów LR

* Wykonanie rozkazu przebiega w cyklu:

  a) Prześlij zawartość LR do RA.

  b) Dokonaj odczytu odpowiedniego słowa pamięci.

  c) Umieść zawartość RB w RR.

  d) Dokonaj dekodowania zawartości RR.

  e) Jeśli rozkaz jest rozkazem sterującym, to zmień zawartość LR i rozpocznij nowy cykl od punktu a), jeśli innym, to go wykonaj.

  f) Zwiększ licznik rozkazów LR o 1.

* Przykładowa lista rozkazów

load, <tryb adr.>, <arg>	-umieść liczbę w akumulatorze
store, <tryb adr.>, <arg>	-umieść liczbę z akumulatora w odpowiednim słowie pamięci
add, <tryb adr.>, <arg>	-dodaj liczbę do zawartości akumulatora
sub, <tryb adr.>, <arg>	- odejmij liczbę od zawartości akumulatora

Przykładowa lista rozkazów c.d.

jump, <tryb adr.>, <arg>	- skocz do kroku wyznaczonego przez argument
jumppos, <tryb adr.>, <arg>	- skocz do kroku wyznaczonego przez argument, gdy w akumulatorze jest liczba dodatnia
jumpzero, <tryb adr.>, <arg>	- skocz do kroku wyznaczonego przez argument, gdy w akumulatorze jest zero
stop	- zatrzymaj działanie komputera

* Przykładowy program dodający dwie liczby znajdujące się w czwartym i piątym słowie PAO i umieszczający ich sumę w słowie szóstym.

Obraz pamięci:

0: load, b, 4

1: add, b, 5

2: store, n, 6

3: stop

4: ... tu pierwszy składnik ...

5: ... tu drugi składnik ...

6: ... tu suma ...