COMP10020
Introduction to Programming II
Algorithms 1

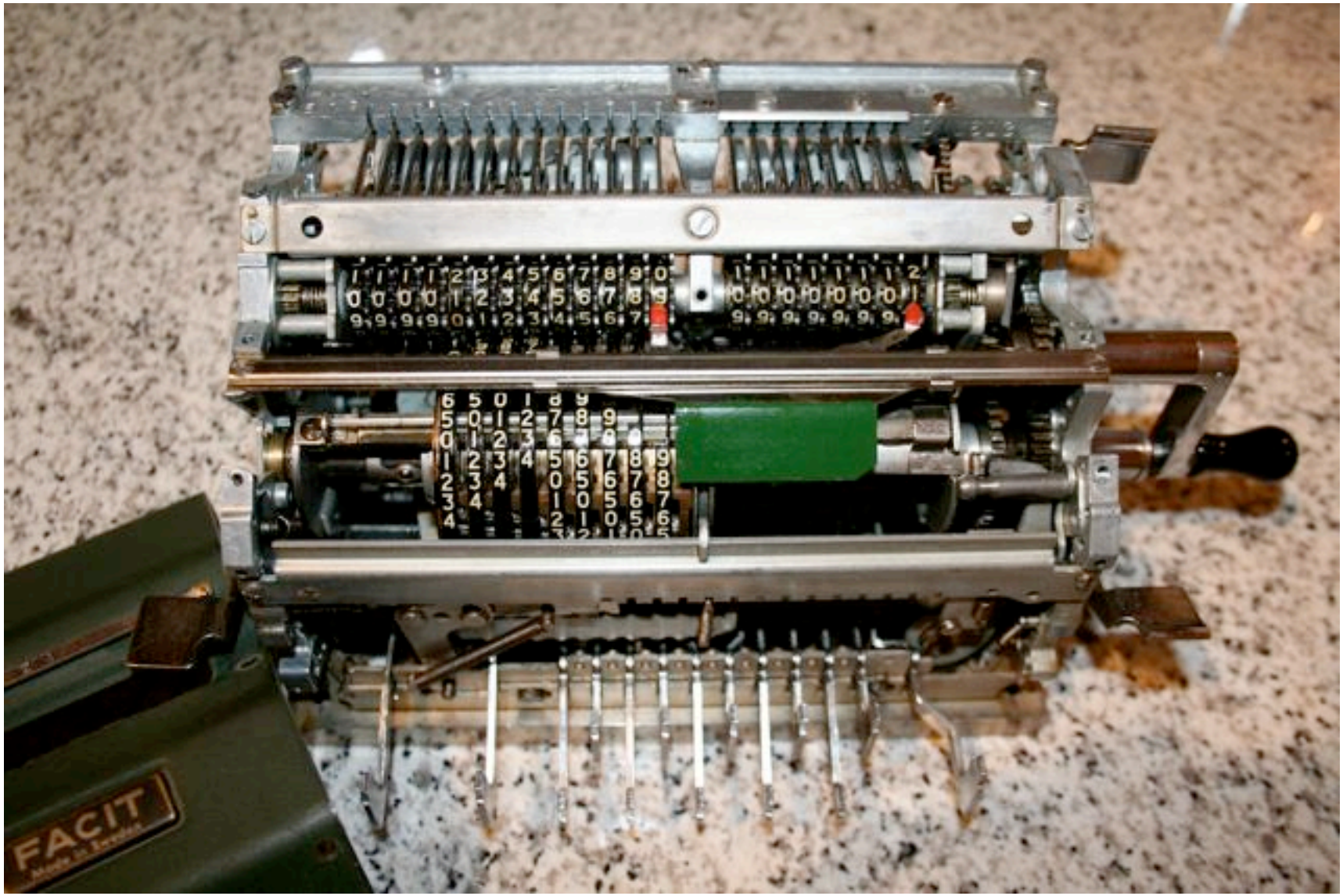
Dr. Brian Mac Namee
brian.macnamee@ucd.ie
School of Computer Science
University College Dublin

EARLY COMPUTING

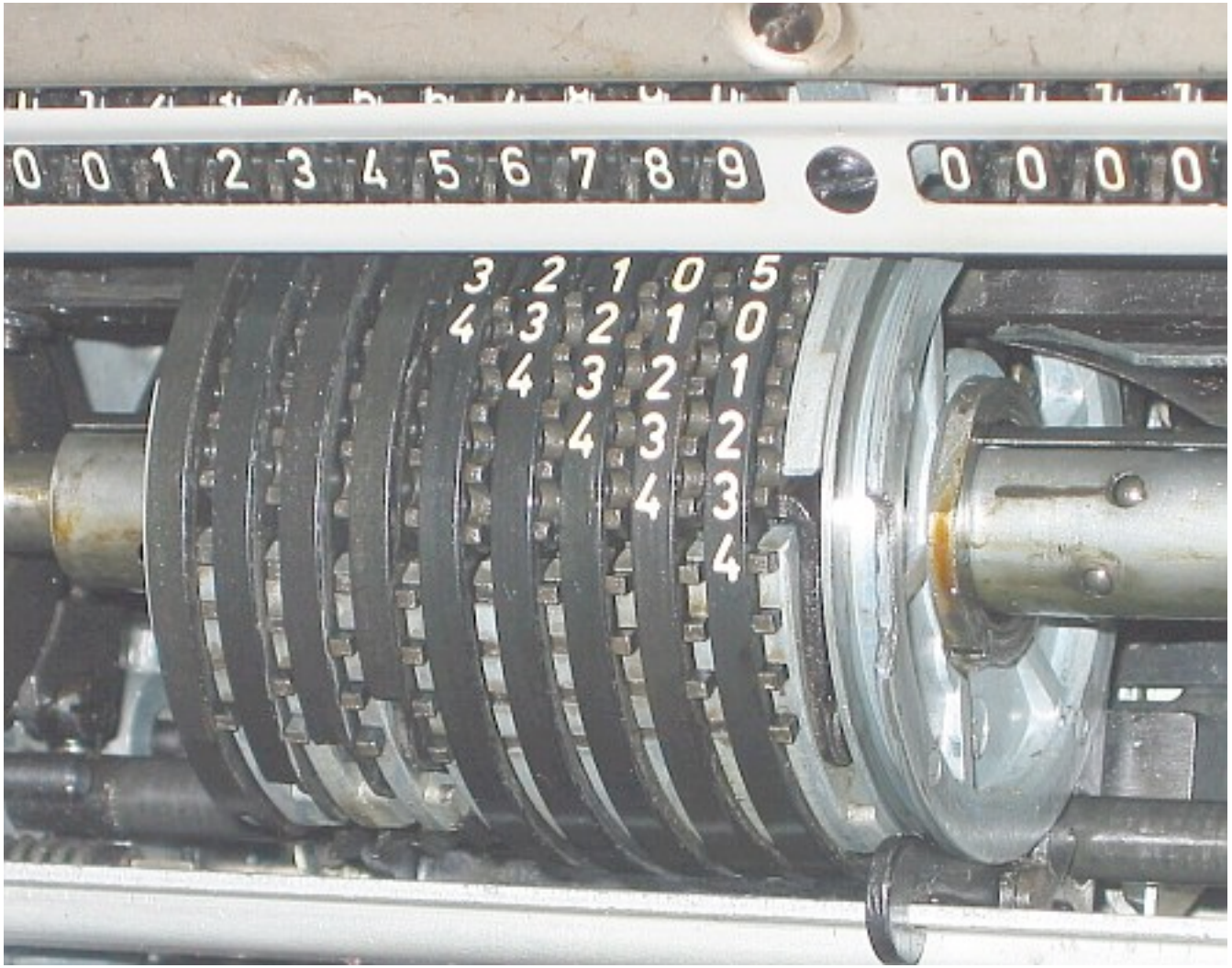
FACIT C1-13 Mechanical Computer circa 1960

Prof. Dr.-Ing. Christian-M. Hamann Online Museum http://public.beuth-hochschule.de/hamann/calc/index.html


FACIT
INCORPORATED
NEW YORK • CHICAGO
SAN FRANCISCO
FACIT-ODHNER
CALCULATORS ADDING MACHINES
FACIT
TYPEWRITERS
KAYE OUTDOOR ADV. CO. INC.



Prof. Dr.-Ing. Christian-M. Hamann Online Museum http://public.beuth-hochschule.de/hamann/calc/index.html



Prof. Dr.-Ing. Christian-M. Hamann Online Museum http://public.beuth-hochschule.de/hamann/calc/index.html


Arithmetic unit
Input unit
Counter
FACIT


Arithmetic unit
Clear counter
Clear arithmetic unit
Input unit
Counter
FACIT
Addition/ subtraction crank
Shift input unit (left/right)
Clear input unit
Number input
Jump input unit

OPERATIONS

Examples

111 + 56 = 167

123 + 45 - 6 = 162

123 - 204 = -81

8 * 12 = 96

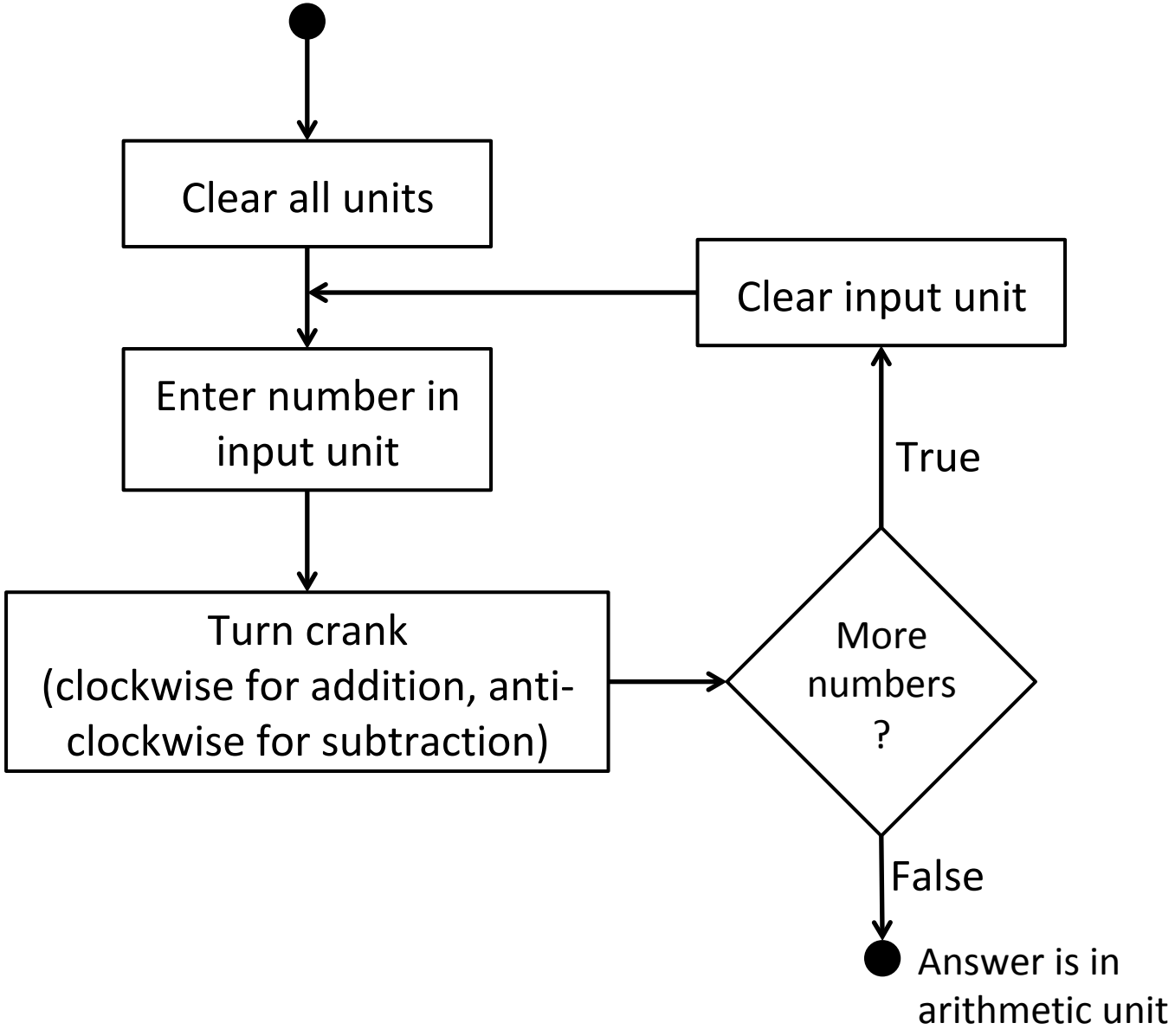
8 * 45 = 360

56 / 8 = 7

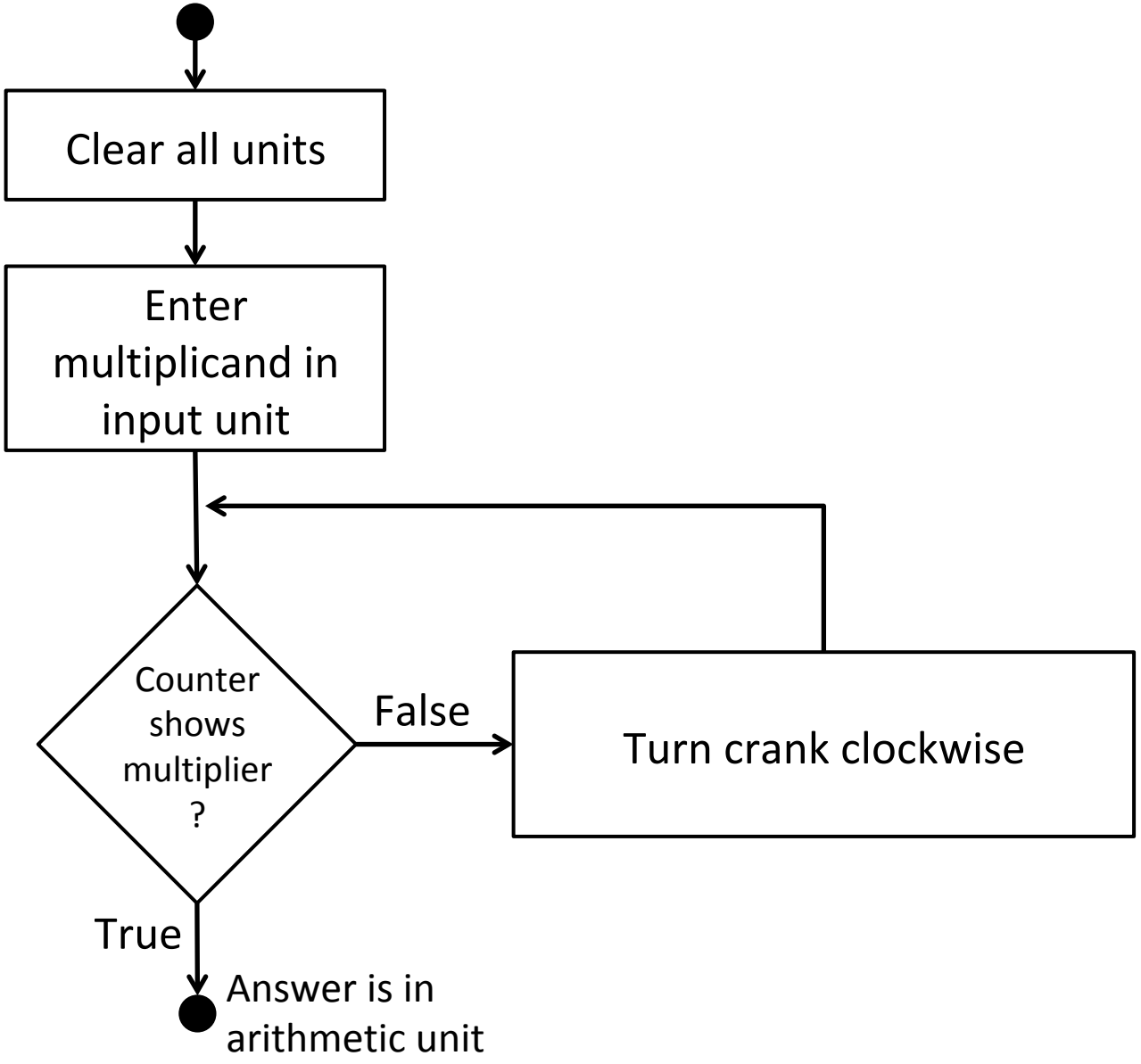
1 / 4 = 0.25

22 / 7 = 3.14

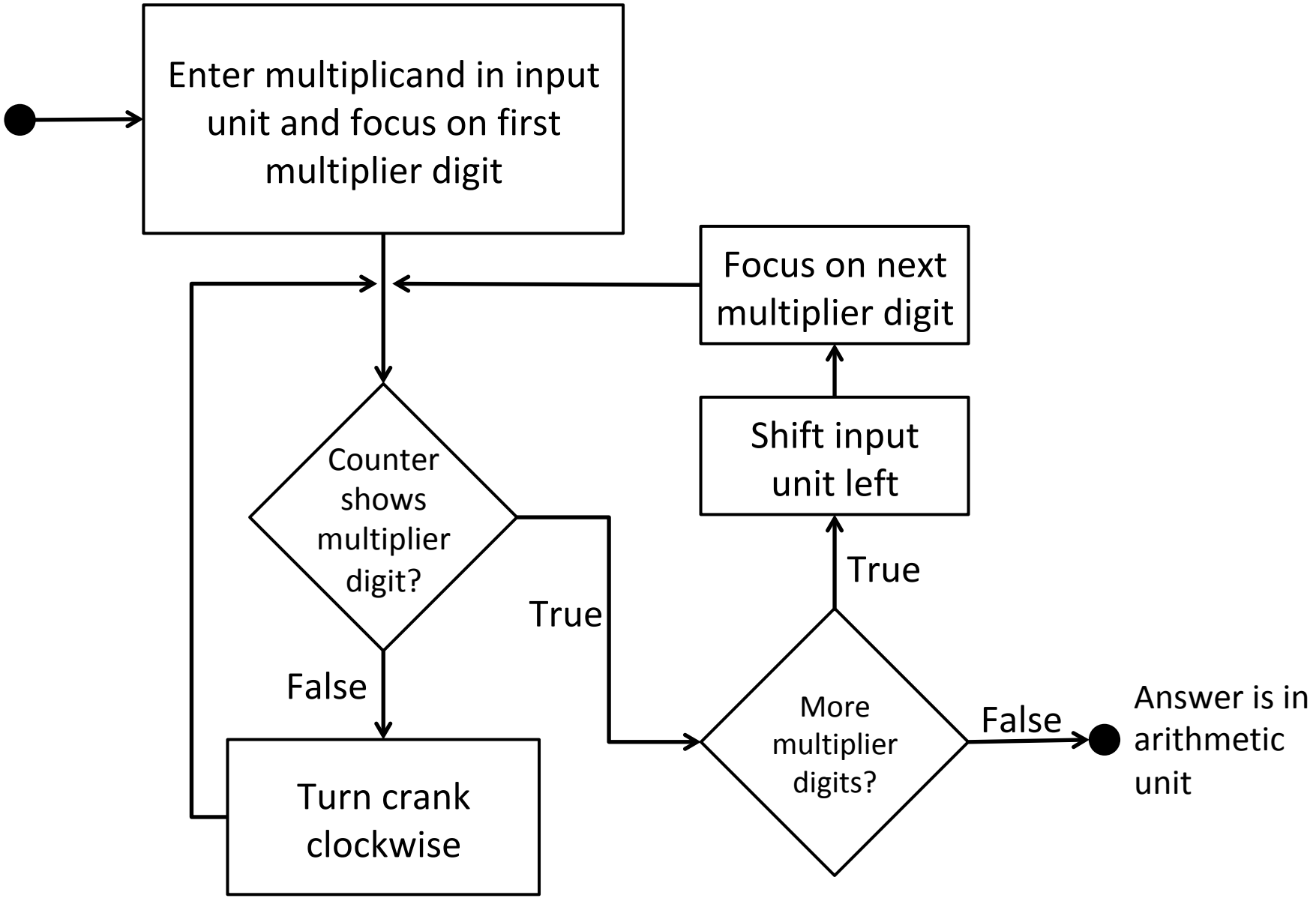
Addition/Subtraction



Multiplication

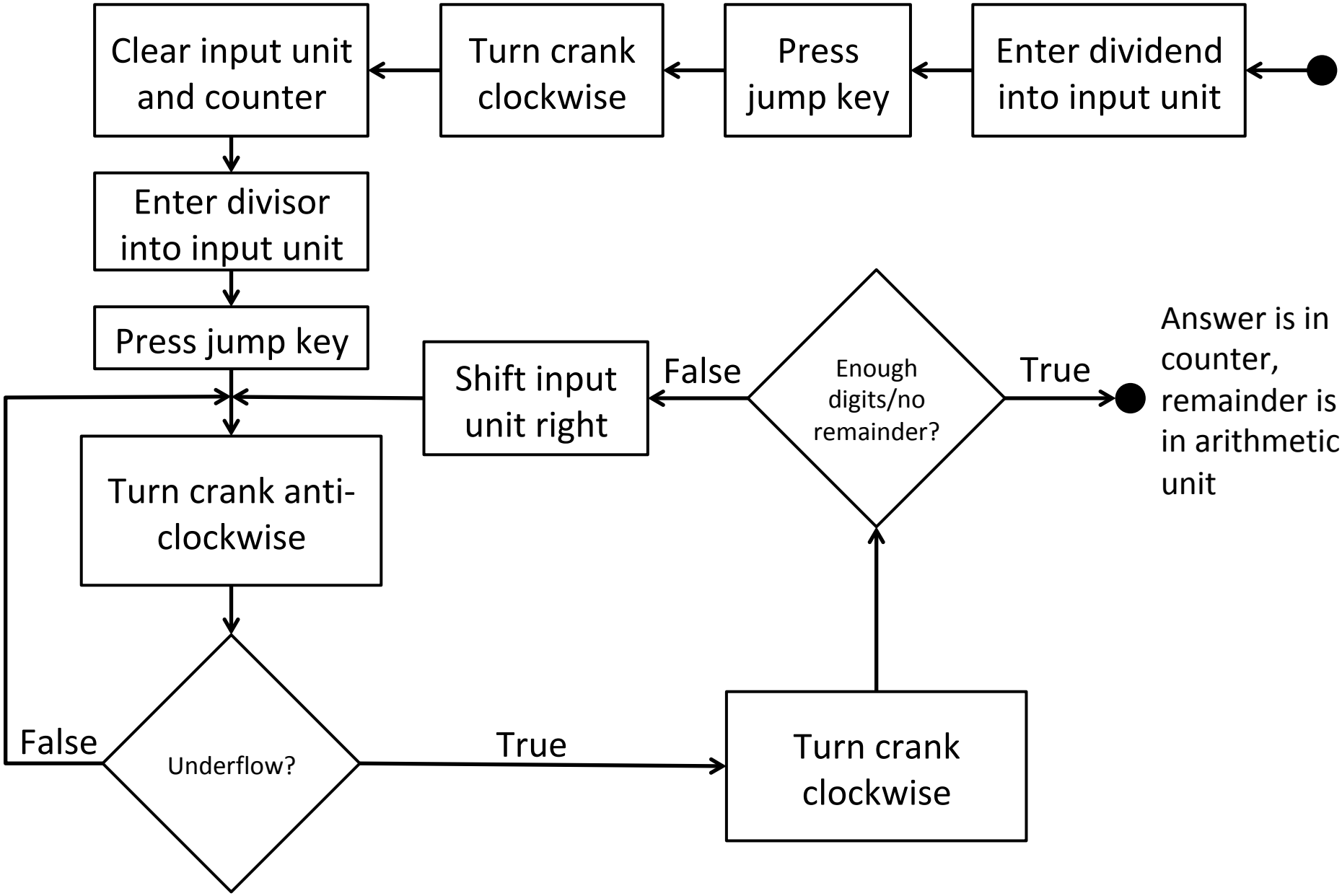


Multiplication (Faster!)



Prof. Dr.-Ing. Christian-M. Hamann Online Museum http://public.beuth-hochschule.de/hamann/calc/index.html

Division


Clear input unit and counter
Turn crank clockwise
Press jump key
Enter dividend into input unit
Enter divisor into input unit
Press jump key
Shift input unit right
False
Enough digits/no remainder?
True
Answer is in counter, remainder is in arithmetic unit
Turn crank anti-clockwise
False
Underflow?
True
Turn crank clockwise

Division

(A) To Set the Dividend into Arithmetic Unit:

Enter the dividend into input unit. Press the JUMP [<<] key. Make a positive (clockwise) turn with the crank to transfer the input into arithmetic unit.

(B) To Set the Divisor into Input Unit:

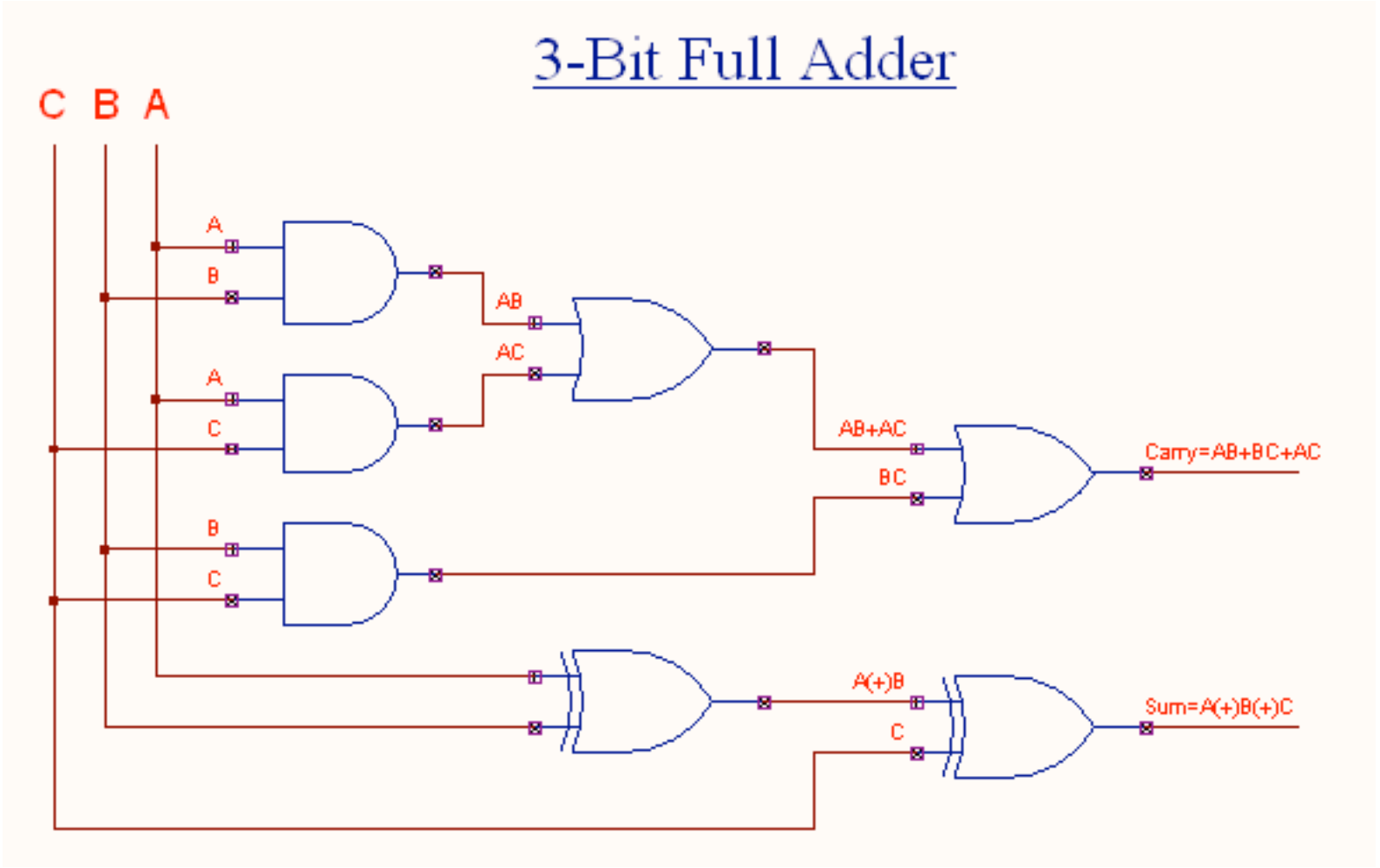
Clear input and counter unit with one grip. Enter the divisor into input unit. Press the JUMP [<<] key.

(C) To Divide:

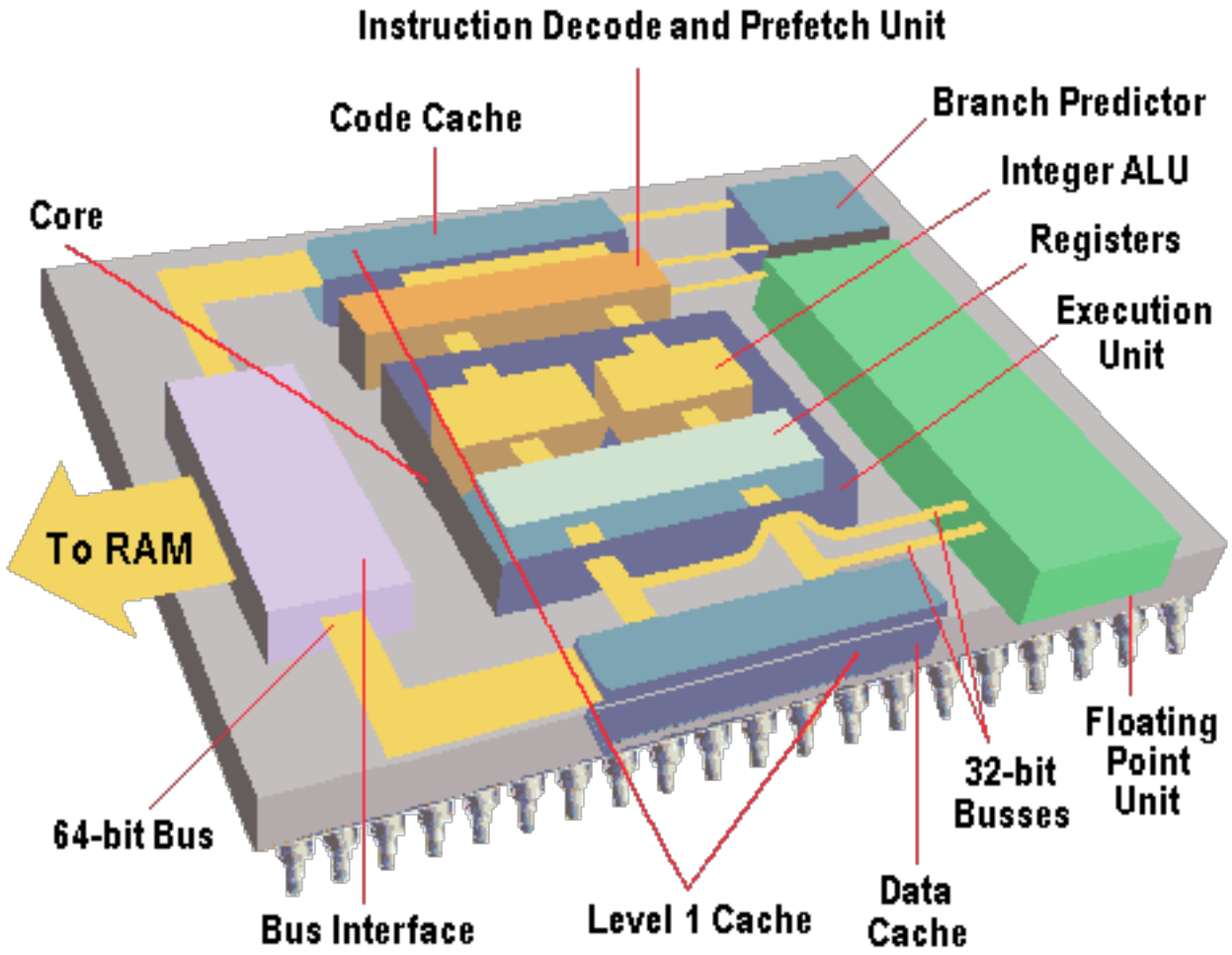
Make negative (counter-clockwise) turns with the crank (now the red indicator is on) until the arithmetic unit shows an "underflow". Make one positive (clockwise) turn with the crank. Move the input unit with the STEP [>] key one position to the right. Make negative turns again ... and repeat this procedure until the required number of decimals are calculated... The result is in the counter unit, and the remainder is in the arithmetic unit. The divisor stays in the input unit, therefore an additional decimal can be estimated...

DIFFERENT PROGRAMMING LANGUAGE LEVELS

Adder Circuit Diagram



Pentium Chip Structure Diagram



Assembly Language

```
MONITOR FOR 6802 1.4          9-14-80  TSC ASSEMBLER  PAGE    2


C000                    ORG    ROM+$0000 BEGIN MONITOR
C000 8E 00 70  START    LDS    #STACK

                ****************************************
                * FUNCTION: INITA - Initialize ACIA
                * INPUT: none
                * OUTPUT: none
                * CALLS: none
                * DESTROYS: acc A

0013            RESETA  EQU    %00010011
0011            CTLREG  EQU    %00010001

C003 86 13      INITA   LDA A  #RESETA   RESET ACIA
C005 B7 80 04           STA A  ACIA
C008 86 11              LDA A  #CTLREG   SET 8 BITS AND 2 STOP
C00A B7 80 04           STA A  ACIA

C00D 7E C0 F1           JMP    SIGNON    GO TO START OF MONITOR

                ****************************************
                * FUNCTION: INCH - Input character
                * INPUT: none
                * OUTPUT: char in acc A
                * DESTROYS: acc A
                * CALLS: none
                * DESCRIPTION: Gets 1 character from terminal

C010 B6 80 04   INCH    LDA A  ACIA      GET STATUS
```

Python

```
for line in open("file.txt"):
    for word in line.split():
        if word.endswith('ing'):
            print(word)
```

SUMMARY

Summary

Algorithm design is all about determining a way to perform complex operations using simple operations as efficiently as possible