ORG 0x50

ADDR1: WORD $KG

ADDR2: WORD $GR

KG: WORD 0xC7CB

WORD 0x0000

GR: WORD 0x00C7

A: word 0x0000

B: word 0x3E8

C: word 0x7FFF

GR1: word 0x300

RES\_KG: word 0x0000

RES\_GR: word 0x0010

START: CLA

OUT 0x0C

INPUT1: iN 7 ; Ввод старшей части числа

AND #0x40

BEQ INPUT1

IN 6

SWAB

ST A

INPUT2: IN 7 ; Ввод младшей части числа

AND #0x40

BEQ INPUT2

CLA

IN 6

ADD A

ST A

BPL HANDLE\_7FFF

LD #0x20 ; Обработка 8000

ST RES\_KG

HANDLE\_7FFF: LD A ;Обработка 7FFF

AND C

SPIN2: CMP B ;Цикл подсчёта сколько килограмм осталось в числе

BMI OUTER

SUB B

SWAM RES\_KG

INC ;прибавление по 1 кг

SWAM RES\_KG

JUMP SPIN2

;Цикл вывода данных на ву

OUTER: ST RES\_GR ;Проверка если число >= 8000 добавить надо остаток

LD A

BPL OUT\_KG ;Проверка если число < 8000 надо переходить на вывод данных иначе доп обработка

LD GR1 ; Повторение цикла с колличесвтом кг но нужно добавить остаток от 8000

ADD RES\_GR

CMP B

BMI OUTER

SUB B

SWAM RES\_KG

INC

SWAM RES\_KG

ST RES\_GR

OUT\_KG: LD RES\_KG ; Цикл перевода кг в 10сс

BEQ OUT\_GR ; если 0 то пропускаем

PUSH

CALL $CONVERTOR ; Данный метод вызывает как вывод данных так и их перевод

POP

OUTPUT\_KG: LD (ADDR1)+ ; Вывод тектовой строки единиц измерения

BEQ OUT\_GR

OUT 0x0C

SWAB

SXTB

BEQ OUT\_GR

OUT 0x0C

JUMP OUTPUT\_KG

OUT\_GR: LD #0x20

OUT 0x0C

LD RES\_GR ; Цикл перевода г в 10сс

PUSH

CALL $CONVERTOR

POP

OUTPUT\_GR: LD (ADDR2)+ ; Вывод тектовой строки единиц измерения

BEQ STOP\_POINT

OUT 0x0C

SWAB

SXTB

BEQ STOP\_POINT

OUT 0x0C

JUMP OUTPUT\_GR

STOP\_POINT: HLT

ORG 0x200

ASD1: word 0x000A

ASD2: word 0x00A0

ASD3: word 0x0060

QWE: word 0x000F

CONVERTOR: CLA

LD &1

STAGE1: CMP ASD1 ; Цикл для перевода в 10сс заключается в поочерёдном вычитании A и добавлении 6 к 2 разным числам

BMI STAGE2

SUB ASD1

SWAM &1

ADD #0x06

SWAM &1

JUMP STAGE1

STAGE2: LD &1 ;обновление всех данных

STAGE3: CMP ASD2 ; 2 часть цикла заключается в поочерёдном вычитании A0 и добавлении 60 к 2 разным числам

BMI STOP

SUB ASD2

SWAM &1

ADD ASD3

SWAM &1

JUMP STAGE3

STOP: LD &1 ; начало вывода данных на консоль

SWAB

CALL ITERARION\_1

LD &1

CALL ITERARION\_2

LD &1

CALL ITERARION\_1

RET

ITERARION\_1: AND QWE

ADD #0x30

OUT 0x0C

RET

ITERARION\_2: ASR

ASR

ASR

ASR

AND QWE

ADD #0x30

OUT 0x0C

RET

ORG 0x50

ADDR1: WORD $KG

ADDR2: WORD $GR

KG: WORD 0xC7CB

WORD 0x0000

GR: WORD 0x00C7

A: word 0x0000

B: word 0x3E8

C: word 0x7FFF

GR1: word 0x300

RES\_KG: word 0x0000

RES\_GR: word 0x0010

START: CLA

OUT 0x0C

INPUT1: iN 7 ; Ввод старшей части числа

AND #0x40

BEQ INPUT1

IN 6

SWAB

ST A

INPUT2: IN 7 ; Ввод младшей части числа

AND #0x40

BEQ INPUT2

CLA

IN 6

ADD A

ST A

BPL HANDLE\_7FFF

LD #0x20 ; Обработка 8000

ST RES\_KG

HANDLE\_7FFF: LD A ;Обработка 7FFF

AND C

SPIN2: CMP B ;Цикл подсчёта сколько килограмм осталось в числе

BMI OUTER

SUB B

SWAM RES\_KG

INC ;прибавление по 1 кг

SWAM RES\_KG

JUMP SPIN2

;Цикл вывода данных на ву

OUTER: ST RES\_GR ;Проверка если число >= 8000 добавить надо остаток

LD A

BPL OUT\_KG ;Проверка если число < 8000 надо переходить на вывод данных иначе доп обработка

LD GR1 ; Повторение цикла с колличесвтом кг но нужно добавить остаток от 8000

ADD RES\_GR

CMP B

BMI OUTER

SUB B

SWAM RES\_KG

INC

SWAM RES\_KG

ST RES\_GR

OUT\_KG: LD RES\_KG ; Цикл перевода кг в 10сс

BEQ OUT\_GR ; если 0 то пропускаем

PUSH

CALL $CONVERTOR ; Данный метод вызывает как вывод данных так и их перевод

POP

OUTPUT\_KG: LD (ADDR1)+ ; Вывод тектовой строки единиц измерения

BEQ OUT\_GR

PUSH

IT1: IN 0x0D

AND #0x40

BEQ IT1

POP

OUT 0x0C

SWAB

SXTB

BEQ OUT\_GR

PUSH

IT2: IN 0x0D

AND #0x40

BEQ IT2

POP

OUT 0x0C

JUMP OUTPUT\_KG

OUT\_GR: LD #0x20

PUSH

IT3: IN 0x0D

AND #0x40

BEQ IT3

POP

OUT 0x0C

LD RES\_GR ; Цикл перевода г в 10сс

PUSH

CALL $CONVERTOR

POP

OUTPUT\_GR: LD (ADDR2)+ ; Вывод тектовой строки единиц измерения

BEQ STOP\_POINT

PUSH

IT4: IN 0x0D

AND #0x40

BEQ IT4

POP

OUT 0x0C

SWAB

SXTB

BEQ STOP\_POINT

PUSH

IT5: IN 0x0D

AND #0x40

BEQ IT5

POP

OUT 0x0C

JUMP OUTPUT\_GR

STOP\_POINT: HLT

ORG 0x200

ASD1: word 0x000A

ASD2: word 0x00A0

ASD3: word 0x0060

QWE: word 0x000F

CONVERTOR: CLA

LD &1

STAGE1: CMP ASD1 ; Цикл для перевода в 10сс заключается в поочерёдном вычитании A и добавлении 6 к 2 разным числам

BMI STAGE2

SUB ASD1

SWAM &1

ADD #0x06

SWAM &1

JUMP STAGE1

STAGE2: LD &1 ;обновление всех данных

STAGE3: CMP ASD2 ; 2 часть цикла заключается в поочерёдном вычитании A0 и добавлении 60 к 2 разным числам

BMI STOP

SUB ASD2

SWAM &1

ADD ASD3

SWAM &1

JUMP STAGE3

STOP: LD &1 ; начало вывода данных на консоль

SWAB

CALL ITERARION\_1

LD &1

CALL ITERARION\_2

LD &1

CALL ITERARION\_1

RET

ITERARION\_1: AND QWE

ADD #0x30

PUSH

IT6: IN 0x0D

AND #0x40

BEQ IT6

POP

OUT 0x0C

RET

ITERARION\_2: ASR

ASR

ASR

ASR

AND QWE

ADD #0x30

PUSH

IT7: IN 0x0D

AND #0x40

BEQ IT7

POP

OUT 0x0C

RET