

Министерство образования Республики Беларусь
Учреждение Образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра электронных вычислительных средств

Лабораторные работы № 5, 6
«Программирование и проверка базовой операции алгоритмов быстрого
преобразования Фурье»
Вариант 12

Проверил:
Шемаров А.И.

Выполнил:
ст. гр. 850701
Филипцов Д. А.

1 Цель работы

Разработка программы БПФ для алгоритма с замещением данных и прореживанием по времени на ассемблере процессора TMS320VC5402 и её отладка на лабораторном макете TMS320VC5402 DSP Starter Kit (DSK).

2 Задание

Написать программу БПФ и проверить её на трёх тестовых сигналах: постоянная, дельта-импульс, синус заданной гармоника.

Таблица 2.1 – Условие выполнения задания

Вариант	Тип и параметры БПФ			№ гарм.
	N	Прореживание по	Порядок вх. данных	10
12	256	частоте	двоично-инверсный	

3 Ход работы

Assembler-код:

```
.mmregs
.def _c_int00
.text

_c_int00:
    LD #Cn, DP
    STM MemSin, AR1
    STM MemCos, AR4
    STM #N, AR2
    STM k, AR3
    SSBX SXM
    SSBX OVM
    SSBX frct
    NOP

func01:
    XOR A, A
    XOR B, B
    CALL func1
    NOP
    XOR A, A
    XOR B, B
    ST #0, *AR1+
    ST 7FFFh, *AR4+
    NOP
```

```

        CALL func2
        NOP
        NOP
        NOP

INIT_0:
        SSBX sxm ; расширение знака
        SSBX frct ; умножение дробных чисел
        SSBX OVM
        STM PQR, AR1
        STM MemSin, AR2
        STM N, AR4

func3:
        NOP
        LD *AR2+, A
        STL A, *AR1+
        NOP
        BANZ func3, *AR4-
        XOR A, A
        ST 0, Sk
        ST #32767, Ck
        ST 0, Sn
        ST #32767, Cn
        STM #N, AR2
        STM MemSin, AR1
        STM MemCos, AR4
        ST 0, *AR1+
        ST 7FFFh, *AR4+
        STM 0, AR3
        CALL func1
        NOP
        CALL func2
        NOP
        RSBX OVM
        NOP

cost:
        ST 1, IE
        ST 64, N2

loop_Hell:
        XOR A, A
        XOR B, B
        STL A, K      ; k = 0
        STL A, J      ; j = 0
        NOP

loop_Hell2:
        LD J, A
        STL A, I      ; I = J
        NOP

```

```

STM #MemSin, AR2
STM #MemCos, AR3
STM #PQR, AR1
STM #PQI, AR5
NOP
LDM AR2, A
NOP
ADD K, A
STLM A,AR2 ; указатель в массиве синусов на K
NOP
LDM AR3, A
NOP
ADD K, A
STLM A,AR3 ; указатель в массиве косинусов на K
NOP
LDM AR1, A
NOP
ADD I, A
STLM A,AR1 ; указатель в массиве PQR на I
NOP
LDM AR5, A
NOP
ADD I, A
STLM A,AR5 ; указатель в массиве PQI на I
NOP
NOP
NOP

```

BTRFLY:

```

NOP
MVDM N2, AR0
NOP
LD *AR1+0, 15, A ; A = 1/2 PR
NOP
ADD *AR1-0, 15, A ; A = 1/2 (PR+QR)
NOP
STH A, *AR1+0 ; PR = 1/2 new PR
NOP
SUB *AR1, 16, a ; A = XR = 1/2 (PR-QR)
STH a, *AR1 ; QR = a = XR
MAR *AR5+0
LD *AR5-0, 15, a ; A = 1/2 QI
ADD *AR5, 15, a ; A = 1/2 (PI+QI)
STH a, *AR5+0 ; PI = 1/2 new PI
SUB *AR5, 16, a ; A = XI = 1/2 (PI-QI)
STH a, *AR5 ; QI = a = XI
LD *AR1, T ; T = QR = 1/2 (PR-QR)
MPY *AR3, a ; A = 1/2 (PR-QR)*cos
LD *AR5, T ; T = QI = 1/2 (PI-QI)
MAC *AR2, a ; A = 1/2 [(PR-QR)cos+(PI-QI)sin]
MPY *AR3, b ; B = 1/2 (PI-QI)*cos
LD *AR1, T ; T = QR = 1/2 (PR-QR)

```

```

    STH a, *AR1 ; QR = 1/2 new QR
    MAS *AR2, b ; B = 1/2 [ (PI-QI) cos - (PR-QR) sin ]
    STH b, *AR5 ; QI = 1/2 new QI
    NOP
    NOP
    NOP

INIT:
    NOP
    LD N2, A
    SFTA A, 1
    ADD I, A ; I = I + 2 * N2
    STL A, I
    NOP
    STM PQR, AR1
    NOP
    STM PQI, AR5
    NOP
    LDM AR1, A
    NOP
    ADD I, A
    NOP
    STLM A, AR1 ; указатель в массиве PQR на I
    NOP
    LDM AR5, A
    NOP
    ADD I, A
    NOP
    STLM A, AR5 ; указатель в массиве PQI на I
    NOP
    MVDM I, AR0
    NOP
    STM N, AR4
    NOP
    CMPR GT, AR4 ; I < N? (yes: exit)
    NOP
    BC BTRFLY, TC
    NOP

loop_hell3:
    LD IE, A
    ADD K, A
    STL A, K ; K = K + IE
    NOP
    ADDM #1, J ; J = J + 1
    NOP
    MVDM J, AR0
    NOP
    MVDM N2, AR4
    NOP
    NOP
    CMPR GT, AR4 ; J < N2? (yes: exit)

```

```

NOP
BC loop_Hell2, TC
NOP

loop_Hell4:
    LD IE,1, A
    STL A, IE    ; IE = IE*2
    NOP
    LD N2,-1, A
    STL A, N2    ; N2 = N2/2
    NOP
    STM #N, AR4
    NOP
    MVDM IE, AR0
    NOP
    CMPR GT, AR4    ; IE < N (yes: exit)
    NOP
    BC loop_Hell,TC
    NOP
    NOP
    B cost
    NOP

func1:
    LD Sk, T
    LD C1,16, A
    MPYA B
    LD Ck, T
    MAC S1, B
    LD Ck, T
    LD C1,16, A
    MPYA A
    LD Sk, T
    STH B, Sk
    MAS S1, A
    STH A, Ck
    NOP
    BANZ func1, *AR3-
    NOP
    RET

func2:
    LD Sn, T
    LD Ck, 16, A
    MPYA B
    LD Cn, T
    MAC Sk, B
    STH B, *AR1+
    LD Cn, T
    LD Ck, 16, A
    MPYA A
    LD Sn, T

```

```

        STH B, Sn
        MAS Sk, A
        STH A, Cn
        STH A, *AR4+
        NOP
        BANZ func2, *AR2-
        NOP
        RET

        .align
k        .set 10-1
MemSin    .set 1000h
MemCos    .set 1500h
N        .set 256-1
PQR      .set 2000h
PQI      .set 2100h

        .data
Cn        .word 7FFFh
Sn        .word 0
Ck        .word 7FFFh
Sk        .word 0
C1        .word 32728
S1        .word 1608
IE        .word 1
N2        .word 64
K         .word 0
J         .word 0
I         .word 0

```

Выполнение:

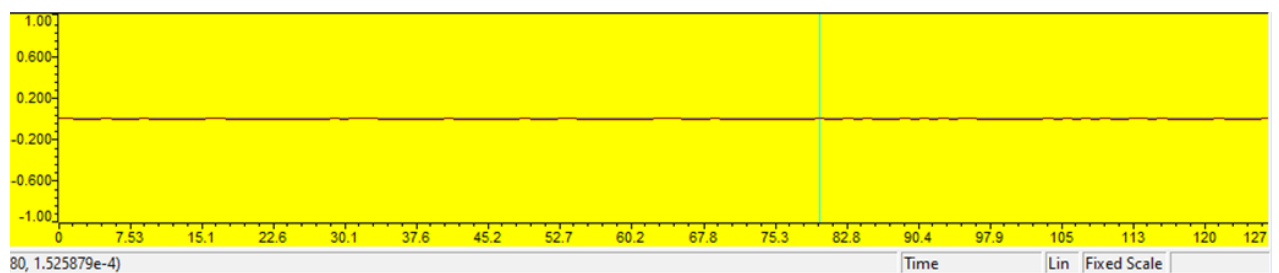


Рисунок 3.1 – Действительная часть синуса 10-ой гармоники

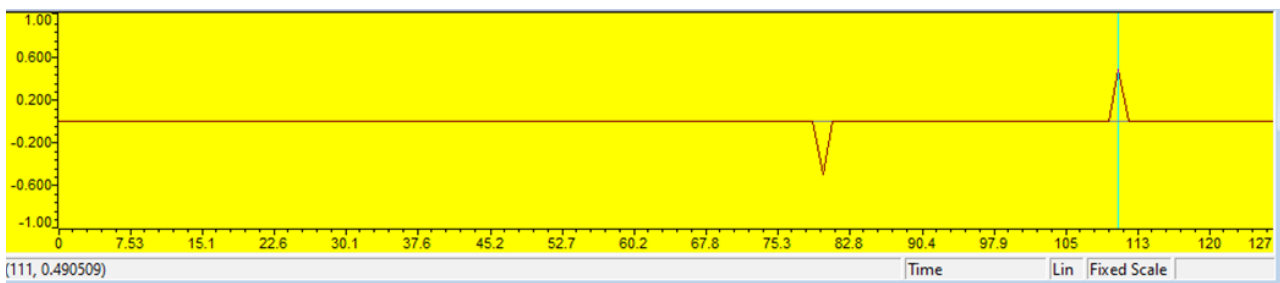


Рисунок 3.2 – Мнимая часть синуса 10-ой гармоники

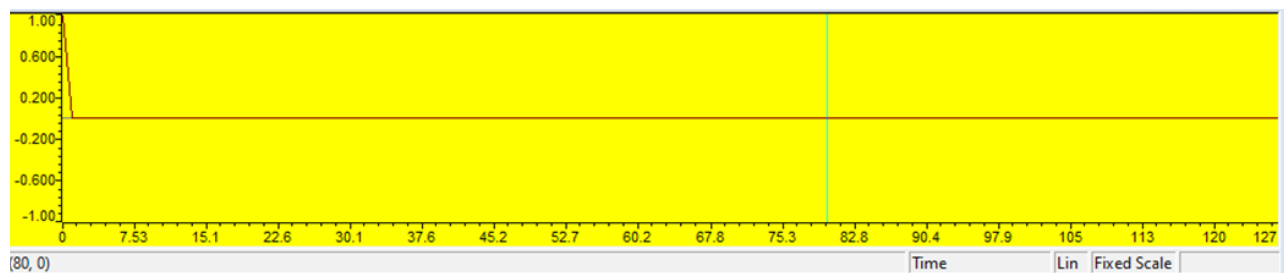


Рисунок 3.3 – Действительная часть постоянного сигнала

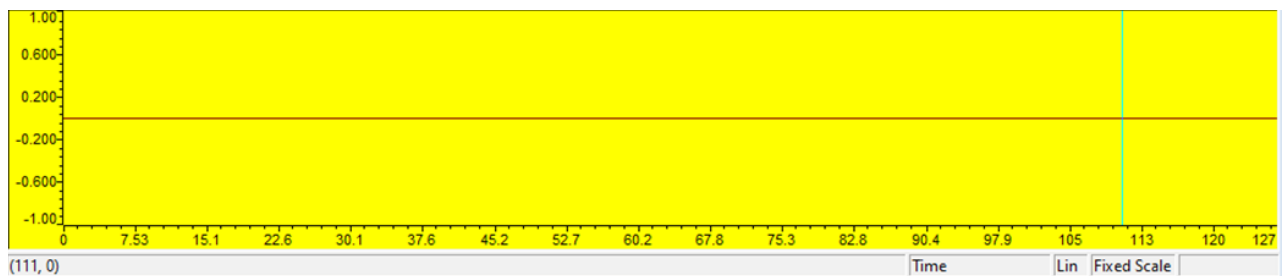


Рисунок 3.4 – Мнимая часть постоянного сигнала

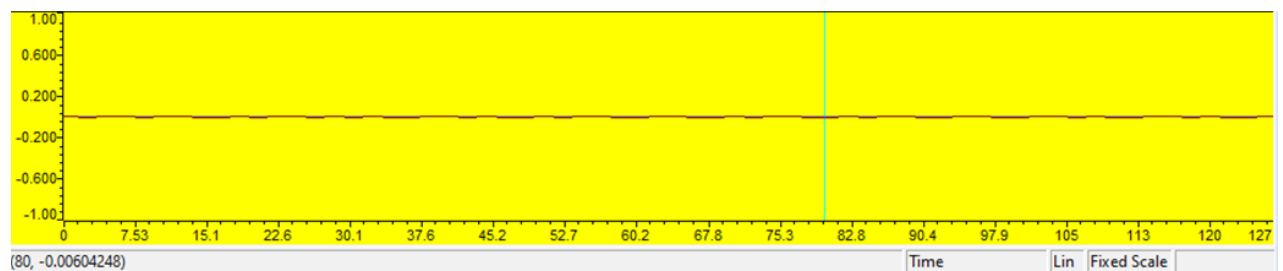


Рисунок 3.5 – Действительная часть единичного импульса

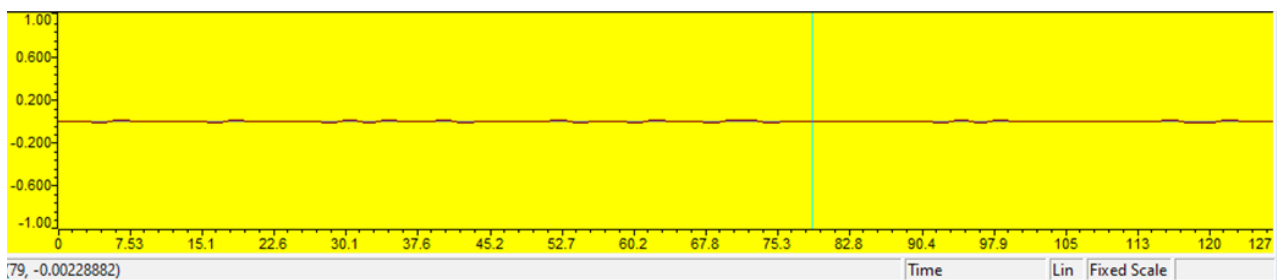


Рисунок 3.6 – Мнимая часть единичного импульса

4 Вывод

В данной лабораторной работе была написана и проверена (на трех разных сигналах) программа БПФ на ассемблере процессора TMS320VC5402.