Тема: Организация обмена между датчиками, УВМ и исполнительными устройствами

Цель: изучение принципов организации инфообмена между ядром СРВ и периферийными устройствами; приобретение практических навыков разработки функциональных схем, алгоритмов и управляющих программ для управления технологическими процессами с использованием измерительных преобразователей и исполнительных устройств.

**Задание:**

Задан технологический процесс или функционирование объекта управления, контроль за состоянием которого осуществляется с помощью аналоговых и дискретных датчиков. Для аналоговых датчиков определены допустимый диапазон измерений, диапазон выходного сигнала, инерционность и интервал или режим опроса. Известен АЦП, преобразующий входную аналоговую информацию в двоичный код. В соответствии с измеряемыми значениями необходимо выдать аналоговое управляющее воздействие на исполнительные устройства, регулирующие протекание технологического процесса или состояние объекта управления. Для вывода аналоговой информации используется ЦАП. Разрядности АЦП и ЦАП, диапазоны аналоговых сигналов заданы в таблице.

**Управление** **вентилятором**. Имеется вентилятор, обеспечивающий проветривание комнаты. Мощность двигателя, а значит, и скорость вращения, определяется управляющим напряжением. Режим включения вентилятора должен соответствовать уровню температуры, которая измеряется четырьмя датчиками, и значению влажности, которая определяется датчиком, дополненным компаратором, выдающим 0, если уровень влажности соответствует норме или 1 в противном случае. Диапазон температуры, измеряемый датчиками, составляет –20°С ÷ 60°С, диапазон выходного сигнала –1,5 ÷ 4,5В. Частота опроса датчиков соответственно равна 20 мс, 80 мс, 60 мс и 120 мс. Пока среднее значение температуры в помещении ниже 20°С – скорость вращения лопастей минимальна. При достижении среднего значения температуры в помещении 20°С вентилятор переключается во второй режим, когда срабатывает датчик влажности – в третий режим. Напряжения, соответствующие режимам, равны –2 В, 2.8 В и 4 В.

|  |  |  |  |
| --- | --- | --- | --- |
| Разрядность АЦП | Разрядность ЦАП | Диапазон напряжения, В | |
| Входной сигнал | Выходное воздействие |
| 12 | 14 | -10…10 | -5…5 |

**Ход работы:**

На следующем рисунке представлена структурная схема работы вентилятора:

**Изображение выглядит как диаграмма, План, Технический чертеж, текст

Автоматически созданное описание**

Рисунок 1 – структурная схема

Градировочная характеристика датчика температуры:

U=

k=

U=0,075(20 - (-20)) - 1,5 = 1,5 B

**Изображение выглядит как линия, диаграмма, График, Параллельный

Автоматически созданное описание**

Рисунок 2 – Градировочная характеристика датчика температуры

Рассчитаем цифровой эквивалент на выходе АЦП уровня напряжения на выходе термодатчика, соответствующего температуре 20°С:

 **933h**.

Рассчитаем цифровой эквивалент на входе ЦАП, соответствующий уровню управляющего воздействия -2 В, 2,8 B, 4B:

 **1333h**.

 **31EBh**.

 **3EB8h**.

Входной порт 300h:

* + 0…11 разряды – Данные с АЦП
  + 14 разряд – Датчик влажности
  + 15 разряд – ГТ

Выходной порт 301h:

* + 0…13 разряды – Данные ЦАП
  + 14 и 15 разряды – Коммутатор
  + 16 разряд – Запуск ЦАП
  + 17 разряд – Запуск АЦП

Параметры датчиков температуры:

* + диапазон входного сигнала -20…60 °С;
  + диапазон выходного сигнала -1,5…4,5 В.

Параметры АЦП:

* + разрядность – 12;
  + диапазон входного сигнала -10…10 В.

Параметры ЦАП:

* + разрядность – 14;
  + диапазон выходного сигнала –5…5 В.

Представление датчиков температуры в коммутаторе:

* + 1 датчик – 00;
  + 2 датчик – 01;
  + 3 датчик – 10;
  + 4 датчик – 11;

Зная, что 1 цикл – 240 мс, рассчитаем кол-во опросов каждого датчика:

Изображение выглядит как линия, График, текст, снимок экрана

Автоматически созданное описание



Рисунок 3 – Цикл опроса датчиков

Получим, что 1 датчик будет опрошен 12 раз, 2 датчик – 3 раза, 3 датчик – 4 раза, 4 датчик – 2 раза. В сумме 21.

Время коммутации сигнала – 10 мс.

Время опроса датчиков в цикле – 20 мс.

Для предоставления процедуры расчета времени задержки при составлении подпрограммы DELAY использовалась следующая информация: при тактовой частоте процессора 3,4 ГГц длительность одного такта процессора приблизительно составляет 0,294 нс; формула поиска числа тактов, за которое выполняется подпрограмма:

Т= TCALL+ТMOV+It\_vn\*(ТMOV+250⋅6\*ТNOP+249⋅TLOOP\_with\_jump+TLOOP\_without\_jump+

+TDEC+TJNZ\_with\_jump) - TJNZ\_with\_jump+TJNZ\_without\_jump+TRET.

В таком случае, для осуществления задержки в 20 мс (20 000 000 нс) необходимо, чтобы процессор сделал . Подставим в формулу след значения: TCALL = 29 , ТMOV = 4 , ТNOP = 3 , TLOOP\_with\_jump = 17, TLOOP\_without\_jump = 5, TDEC = 3, TJNZ\_with\_jump = 16,TJNZ\_without\_jump= 4,TRET = 20, T= , x – количество итераций внутреннего цикла, It\_vn – внешнего цикла. Получим:

29 + 4 + It\_vn \* (4\*3 + 6\*3\*x + (x - 1)\*17 + 5 + 3 +16) – 16 + 4 + 20 =

Определив методом подбора It\_vn = 100, получим x 19436. Подставим найденные значения:

41 + 100 \* (4\*3 + 6\*3\*19436 + (19436 - 1)\*17 + 5 + 3 +16) = 68027941.

Переведем полученное значение в мс: = 20,000214654 мс.

По аналогии сделаем для 10 мс:

Определив методом подбора It\_vn = 50, получим x 19436. Подставим найденные значения:

41 + 50 \* (4\*3 + 6\*3\*19436 + (19436 - 1)\*17 + 5 + 3 +16) = 34 013 991.

Переведем полученное значение в мс: = 10,000113354 мс.

Реализация задержки времени 20 мс осуществлялась при помощи подпрограммы DELAY20, для 10 мс – DELAY10.

Рассмотрим алгоритм функционирования объекта. В самом начале происходит полный цикл опроса датчиков, согласно Рисунку 3. При опросе каждого датчика проверяются значения влажности. В случае, если хотя бы у одного оно будет равно 1, то цикл прекратится и вентилятор перейдет в 3 режим. В противном случае, вычислится среднее значение температур, которое впоследствии будет сравниваться с контрольным значением. Исходя из этого будет выбираться режим вентилятора (либо 1, либо 2).

Исходя из того, что за цикл в сумме необходимо опросить 21 раз датчики, были введены переменные количества и суммы, в которую в последствии будет записываться среднее значение температуры. Рассмотрим метки программы, реализующие данный алгоритм:

.Start – Задает количество итераций в цикле. В данном случае оно равно 12.

Далее описываются метки для работы датчиков, опрос которых не пересекается с другими (кроме 1). Работа в них ведется аналогично: запуск АЦП и установка соответствующего значения коммутатора, ожидание коммутации, опрос датчиков до тех пор, пока ГТ не станет равен 1, проверка датчика влажности, выделение данных с АЦП, добавление к переменной суммы, очистка ax для след. Датчика, задержка 20мс и переход на след. Итерацию.

.Met1\_o, .Met1\_i – для 1 датчика

.Met2\_o, .Met2\_i – для 2 датчика

.Met3\_o, .Met3\_i – для 3 датчика

Главное отличие заключается в том, что 1 датчик опрашивается на каждой итерации цикла, 2 – на четвертой, 3 – на третьей и девятой.

Рассмотрим метки для нескольких датчиков, опрашиваемых на одной итерации. Их работа аналогична меткам Met..\_o и Met..\_i, за исключением перехода к след.метке:

.T1\_T2\_T3\_T4\_2\_i,.T1\_T2\_T3\_T4\_2\_o,.T1\_T2\_T3\_T4\_3\_i, .T1\_T2\_T3\_T4\_3\_o,.T1\_T2\_T3\_T4\_4\_i, .T1\_T2\_T3\_T4\_4\_o – опрос всех датчиков на 12 итерации

T1\_T3\_T4\_3\_o, T1\_T3\_T4\_3\_i, T1\_T3\_T4\_4\_o, T1\_T3\_T4\_4\_i – опрос 1, 3, 4 датчиков на 6 итерации.

Каждая из вышеописанных меток переходит в DELAY20, служащей задержкой на 20 мс. После нее запускается метка Loop\_next.

. Loop\_next – следит за изменением итераций. В случае, когда цикл закончится, происходит поиск среднего значения температуры и переход на main\_cap\_1.

. main\_cap\_1 – сравниает контрольное значение для температуры равной 20 с полученным значением. В случае, если >=20 осуществляется переход на main\_cap\_2, иначе присваивается значение, соответствующее 1 режиму, включается ЦАП и происходит вывод управляющих воздействий. Цикл запускается сначала.

. main\_cap\_2 – присваивается значение, соответствующее 2 режиму, включается ЦАП и происходит вывод управляющих воздействий. Цикл запускается сначала.

.main\_cap\_3 – присваивается значение соответствующее 3 режиму, включается ЦАП и происходит вывод управляющих воздействий. Цикл запускается сначала.

Код программы:

.686

include \masm32\include\io.asm

.data

kol db 21 ; Количество элементов

sum dw 0 ; Сумма

.code

; Инициализация на начальном этапе

start:

mov bx, 12

;Датчик температуры 1

Met1\_o:

mov eax, 0

mov dx, 301h

out dx, eax

or eax, 20000h

out dx, eax

mov dx, 300h

Met1\_i:

in ax, dx

test ax, 8000h ; Проверка ГТ

jz Met1\_i

test ax, 1000000000000000b ; Проверка влажности14 разряда

jz main\_cap\_3 ; Если 1 переход

and ax, 0FFFh

ADD sum, ax ; Вычисляем сумму

mov ax, 0

start\_1: ; Проверка итераций в цикле

cmp cx, 12

je T1\_T2\_T3\_T4\_2\_o

cmp cx, 9

je Met3\_o

cmp cx, 8

je Met2\_o

cmp cx, 6

je T1\_T3\_T4\_3\_o

cmp cx, 4

je Met2\_o

cmp cx, 3

je Met3\_o

jmp DELAY20 ; ; Задержка 20 мс + след итерация

; Датчик температуры 2

Met2\_o:

or eax, 4000h ; + 01 коммутации

mov dx, 301h

out dx, eax

CALL DELAY10; ожидание коммутации 10мс

or eax, 20000h ; Запуск АЦП

mov dx, 300h

Met2\_i:

in ax, dx

test ax, 8000h ; Проверка 15р ГТ

jz Met2\_i

test ax, 1000000000000000b ; Проверка влажности

jz main\_cap\_3 ; Если 1 переход

and ax, 0FFFh

ADD sum, ax ; Вычисляем сумму

mov ax, 0

jmp DELAY20 ; Задержка 20 мс + след итерация

; Датчик температуры 3

Met3\_o:

or eax, 8000h ; + 10 коммутации

mov dx, 301h

out dx, eax

CALL DELAY10; ожидание коммутации 10мс

or eax, 20000h ; Запуск АЦП

mov dx, 300h

Met3\_i:

in ax, dx

test ax, 8000h ; Проверка 15р ГТ

jz Met2\_i

test ax, 1000000000000000b ; Проверка влажности

jz main\_cap\_3 ; Если 1 переход

and ax, 0FFFh

ADD sum, ax

mov ax, 0

jmp DELAY20 ; Задержка 20 мс + след итерация

T1\_T2\_T3\_T4\_2\_o:

or eax, 4000h ; 01 коммутатор

mov dx, 301h

out dx, eax

CALL DELAY10;

or eax, 20000h ; Запуск АЦП

mov dx, 300h

T1\_T2\_T3\_T4\_2\_i:

in ax, dx

test ax, 8000h ;

jz T1\_T2\_T3\_T4\_2\_i

test ax, 1000000000000000b

jz main\_cap\_3

and ax, 0FFFh

ADD sum, ax

mov ax, 0

; Датчик температуры 3

T1\_T2\_T3\_T4\_3\_o:

or eax, 8000h ; 10 коммутатор

mov dx, 301h

out dx, eax

CALL DELAY10;

or eax, 20000h ; запуск АЦП

mov dx, 300h

T1\_T2\_T3\_T4\_3\_i:

in ax, dx

test ax, 8000h ;

jz T1\_T2\_T3\_T4\_3\_i

test ax, 1000000000000000b ;

jz main\_cap\_3 ;

and ax, 0FFFh

ADD sum, ax ;

mov ax, 0

; Датчик температуры 4

T1\_T2\_T3\_T4\_4\_o:

or eax, C000h ; 11 коммутатор

mov dx, 301h

out dx, eax

CALL DELAY10

or eax, 20000h ; Запуск АЦП

mov dx, 300h

T1\_T2\_T3\_T4\_4\_i:

in ax, dx

test ax, 4000h

jz T1\_T2\_T3\_T4\_4\_i

test ax, 1000000000000000b

jz main\_cap\_3

and ax, 0FFFh

ADD sum, ax

mov ax, 0

jmp DELAY20

; Датчик температуры 3

T1\_T3\_T4\_3\_o:

or eax, 8000h ; 10 коммутатор

mov dx, 301h

out dx, eax

CALL DELAY10

or eax, 20000h ; Запуск АЦП

mov dx, 300h

T1\_T3\_T4\_3\_i:

in ax, dx

test ax, 8000h

jz T1\_T3\_T4\_3\_i

test ax, 1000000000000000b

jz main\_cap\_3

and ax, 0FFFh

ADD sum, ax

mov ax, 0

; Датчик температуры 4

T1\_T3\_T4\_4\_o:

or eax, C000h ; 11 коммутатор

mov dx, 301h

out dx, eax

CALL DELAY10

or eax, 20000h ; Запуск АЦП

mov dx, 300h

T1\_T3\_T4\_4\_i:

in ax, dx

test ax, 8000h

jz T1\_T3\_T4\_4\_i

test ax, 1000000000000000b

jz main\_cap\_3

and ax, 0FFFh

ADD sum, ax

mov ax, 0

jmp DELAY20

; Конец цикла + поиск среднего

Loop\_next:

dec bx

jnz Met1\_o

MOV AX, sum

DIV kol

MOV sum, AX

jmp main\_cap\_1

;Третий режим работы

main\_cap\_3:

mov dx,301h

mov eax, 3EB8h

or eax,10000h ; Запуск ЦАП

out dx, eax

jmp start

;Первый режим работы

main\_cap\_1:

mov edx,301h

cmp eax, 933h ;Сравнение с контрольным значением

jge main\_cap\_2 ; если больше или равно 20

mov eax,1333h

or eax,10000h ; Запуск ЦАП

out dx, eax

jmp start

;Второй режим работы

main\_cap\_2:

mov eax, 31EBh

or eax,10000h ; Запуск ЦАП

out dx, eax

jmp start

; Задержка 10мс

DELAY10:

MOV SI, 50

L1: MOV CX,19436

MOV SI, SI

NOP

NOP

NOP

NOP

L2:

NOP

NOP

NOP

NOP

NOP

NOP

LOOP L2

DEC SI

JNZ L1

RET

; Задержка 20мс

DELAY20:

MOV SI, 100

L20\_1: MOV CX,19436

MOV SI, SI

NOP

NOP

NOP

NOP

L20\_2:

NOP

NOP

NOP

NOP

NOP

NOP

LOOP L20\_2

DEC SI

JNZ L20\_1

jmp Loop\_next ; переход на след итерацию

end start

**Вывод:** изучили принципы организации инфообмена между ядром СРВ и периферийными устройствами; приобрели практические навыки разработки функциональных схем, алгоритмов и управляющих программ для управления технологическими процессами с использованием измерительных преобразователей и исполнительных устройств.