

Отправка данных целевой функции на МСЦ с Nextion.

Страница с настройкой программы: этапы участка коза, выполняющие;

1. При запуске переходе на стр. points... требуется загрузка данных программы с МСЦ.

Для этого отправляем управляющую команду на МСЦ в формате (ОУКнаМСЦФ)

номер_программы

операция_установки_точек_программы_на_экране
Nextion_на_странице_points... - ~~0~~ ~~1~~ (READ_POINTS)
#0D; (1)

N - номер программы D - данные, представляющие собой:

$D = N, P$; - разделитель

N - номер программы (1...10) P - номер страницы: (points_1_5 или points_5_10) (1)

Страница set_points:

Операция 1. При запуске переходе на неё произвести загрузку данных конкретной точки, конкретной программы. Для этого отправить упр. команду на МСЦ в ф-те:

#0D;

O - операция установки данных одной точки программы на странице set_points (2)

$D = N, P$

N - номер программы P - номер точки (1...10) (1)

2. ~~Отправка~~ Изменение данных конкретной точки и отправка новых данных на экран.

Эта операция производится при нажатии на

1) Сохранить параметры

Формат отправки:

#0D; O - операция записи на флэш-носитель данных точки (3)

$D = N, P, H, M, T$

↓ ↓ H - часы M - минуты T - температура

II Изобретение в МСЦ:

Операция (3):

Запись производится в ^{файл} папку на SD микро-SD

Папка: с названием: ~~программ.1.txt~~ ^{1-номер программы} ~~1.txt~~ ^{папка папки файлы}

Папка с программами: программ, в ней ^{файлы} с конкретными программами, с именами "1, 2, ..., 10". В ^{файлах} ~~этих~~ ^{папках} есть ² ~~файла~~ ¹ ~~назва~~ ^а ~~minute~~ ^а ~~temp~~, в которых хранится информация в столбцах о часах, минутах и температуре необходимой программы. Строк 10 штук, т.к. 10 точек.

Операция (2)

Производится чтение с SD карты ~~из~~ по пути:

программ/N-пр./~~на~~ ^{номер программы} ~~цель-файла~~ ^{номер точки} ^{1, 2 или 3, потеряв} ^{исёт} ^{запись} ^в ^{ОЗУ памяти МСЦ}

Инициализация SD карты при включении МСЧ

✓ 1. Создать все папки, если их нет.

✓ 2. Создать необходимые файлы и проинициализировать их любыми значениями, если ~~их~~ их размер не соответствует (слишком мал)

✓ 3. Создание объекта ~~file~~ ~~File~~ file_obj

Функция ~~SD~~ init_SD();

Функция set_point_data(~~1~~ last)

Разбирает строку last и помещает в нее

D = ~~1~~ P, H, M, T

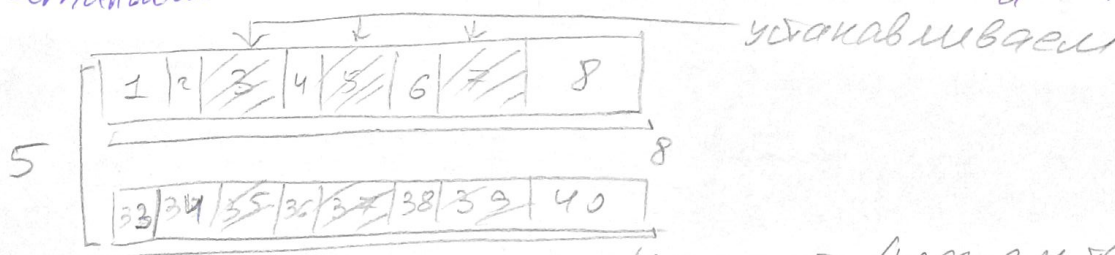
устанавливает значения полей в массиве.

строка

~~1~~ P, H, M, T
2 4 2 4
3-6 8-9 11-14

Функция set_point_data2(str, int s)

Устанавливает параметры программы на стр. points_1-5 и points_5-10



= 40 полей (исп-ем только 15)

str - строка s - номер ~~строки~~ (1-5)
точка 10
строки (1-5)

Файл с параметрами программы (структура)

номер

строки ~~12+2+14~~

1 $P, M, M, T = D$

2

$i, i, M_i, M_i, T_i, DA = D_i$

10 $i_0, M_{i_0}, M_{i_0}, T_{i_0}, DA$

Всего кол-во байт
(размер файла)

~~$10 \times 14 = 140$ байт~~

либо ~~138~~ без

$17 \times 10 = 170$ байт

либо 168 байт без (CR+LF) в последней строке

Всего объем всех программ = $168 \times 10 = 1680$ байт > 1 КБ

$1 \text{ КБ} = 2^{10} \text{ байт} = 1024$

• номера первых байт каждой строки:

1	0-16	$17 \times 0 = 0$
2	17-33	$17 \times 1 = 17$
3	34-50	$17 \times 2 = 34$
4		
5		
6		
7		
3	34-50	
i	$i \times 17 - (i \times 17 - 1)$	

D - имеет фиксированный размер

P - 2 символа H - 4 символа M - 2 символа

T - 4 символа. Итого 12 символа

но к концу строки добавляется

символ переноса строки и

каждый + 2 символа

итого 14 символов на строку

(байт)

0D0A = DA - условия

перенос перенос

каждый на след. строку

(CR+LF)

+ 3 символа заголовка

Итого 17 символов на строку

Операция 1 (UMF · READ_POINTS)

номер: 005 "A" (ASCII)

Установка точек программы на экране Nextion на страницах points...

НМІ подает #00; $D = N, P$ N - номер программы
P - номер страницы

Переменные: $last = D$

int N

int P

(0
точки 1-5 ' 4
точки 6-10)

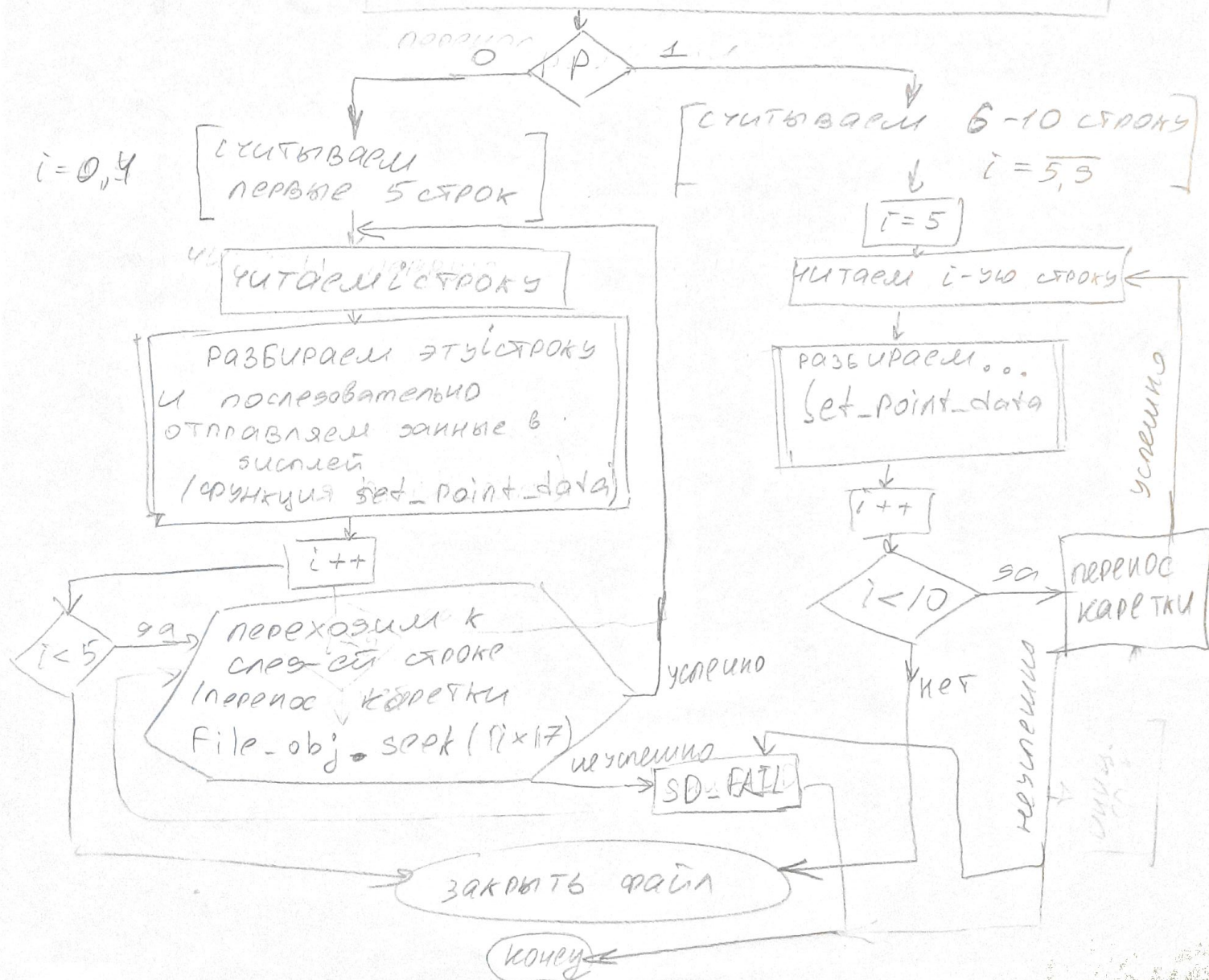
Алгоритм в MCLR

инициализация SD-карты в setup

ПАРСИНГ СТРОКИ И НАХОЖДЕНИЕ N И P

Формирование адреса файла
 $address = "program/N.txt"$

ОТКРЫТИЕ ФАЙЛА С ЭТИМ АДРЕСОМ



Блок-схема операции (3) - с номером SD (ASCII)

Необходимо записать в файл данные, пришедшие с датчика (HMI)

Имя: WRITE_POINT

номер: 62 (ASCII)

Но входе: переменная $last = D = \overset{2}{N}, \overset{3}{P}, H, M, T$ - строка

Переменные:

15 симв

int program_number = 0 - номер программы

String program_number = last.substring(0, 2) - символы номера

String address = "";

File file_obj; - объект класса файл, вспомогательный

String point_number = last.substring(3, 2);

Алгоритм

Инициализация SD-карты в setup

Обрезаем строку last, чтобы
 $last = P, H, M, T$

Формируем имя файла для записи
(адрес)
 $address = "program/No.txt"$

Открытие файла (должен быть создан ранее)
 $file_obj = SD.open(address, FILE_WRITE)$

Перенос каретки на начало нужной строки

$file_obj.seek((point_number - 1) * 17 / 1)$

Возвр. значение true?
(позиция кайзера)
успешно?

Значит файл
не проинициализирован
(не имеет нужного
размера)

нужно инициализировать
SD-карту заново и
потом снова повторить
(функция initSDCard())

Запись:
 $file_obj.write(last)$

Закрытие
файла

конец

SD_CARD_FAILED

Операция 2 Алгоритм имя: READ_POINT

На входе: переменные: $\&last = D = \frac{2 \cdot 10^2}{5} \cdot P$

Производит чтение с SD-карты по пути "програм/№.txt" и запись в НМІ в нужное поле.

Переменные: String address
int P

Алгоритм ВМСЧ

Инициализация SD-карты в setup

Формируем адрес файла (путь)
address = "програм/№.txt"

Находим P в строке last и переводим в число и записываем в P

буфер кол-во символов

открываем файл с address

чтение в пер-ую last
file_obj.read(last, 15)

успешно

переносим каретку на нужную строку
file_obj.seek((P-1)*17)

закрывать файл

неуспешно

"SD-CARD-FAILED"

разбор строки
last и последовательная установка значений полей в массив

ар-ум set_point_data

конец

Алгоритм в НМІ (Формирование запроса #0D)

Нужно сформировать $D = \frac{2 \cdot 10^2}{5} \cdot P$. Переводим N в строковый массив длиной N + 1 в 2 символа

нмало

Print "#149"

points = 5-10. програм.var

конвертируем номер программы в +xx и записываем в переменную N

Print N

P

Print "P" Print "g" → конец

10

Операция WRITE_POINT

Алгоритм формирования запроса в HMI для MCV

$D = N, P, M, M, T$
15

Нумеро из

	Значения
1 → 0001 = 3 ₁₆	1
20 → 0010 = 2 ₁₆	2
0100 = 4 ₁₆	3
1000 = 8 ₁₆	4

отправка кода операции #d62

~~перевос вращ и температура в txt~~
~~часов в txt~~

перевос N в 2-симв. строку
и отправка +;

перевос P в 2-симв. строку +;

перевос H в 4-симв. строку
и отправка +;

перевос M в 4-симв. строку
и отправка +;

перевос T в 4-симв. строку
и отправка

отправка конца
соединения ;