|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Описание программного обеспечения**  **№ 05 Ф 055/02/1\_1** | | | | |
| Наименование изделия | Дозатор шприцевый портативный лекарственных средств | | | |
| Модель | ДШП 5-20-ШМЕЛЬ | | | |
| Класс программного обеспечения | А | | | |
|  | Должность | Ф.И.О. | Подпись | Дата |
| Разработан | Инженер-программист | Николаев А.С. |  | 29.03.2018 |
| Утвержден | И.о. начальника КТО | Тюрин И.Р. |  | 29.03.2018 |

**История внесения изменений**

| **Номер версии** | **Дата** | **Описание изменения** |
| --- | --- | --- |
| 01 | 29.03.2018 | Начальная версия |
|  |  |  |
|  |  |  |
|  |  |  |

1. **Введение**

Дозатор шприцевый портативный лекарственных средств «ДШП 5-20-ШМЕЛЬ» (далее – дозатор) построен на основе микроконтроллера ATmega164PA-AU 8-разрядного семейства AVR фирмы Microchip Technology и предназначен для длительной инфузии препаратов пациенту с заданной скоростью.

1. **Общие сведения**

Аппаратная платформа построена на 8-разрядном микроконтроллере семейства AVR фирмы Microchip Technology ATmega164PA-AU c FLASH-памятью программ объемом 16Кбайт, ОЗУ объемом 1Кбайт и EEPROM-памятью данных объемом 512 байт.

Программа написана на языке С. Компиляция программы выполняется в наборе пакетов программ AVR GCC Toolchain.

Имя проектов программы: Dozator\_1\_0.eww.

Имя исполняемого кода: Dozator\_1\_0.hex.

Запись исполняемого кода во внутреннюю FLASH-память микроконтроллера производиться через USB порт с помощью программатора AVRISP-MkII.

Минимальные требования к персональному компьютеру для компиляции проекта и загрузки исполняемого кода в память микроконтроллера:

* IBM-PC-совместимый компьютер;
* MS Windows XP;
* 512 Мб оперативной памяти;
* Наличие 1 Гб свободного места на жестком диске;
* Наличие USB порта.

1. **Функциональное назначение**

Встроенное программное обеспечение (далее по тексту - ВПО) предназначено для реализации следующих функций:

* начальная инициализация аппаратной части микроконтроллера;
* постоянный самоконтроль работоспособности микроконтроллера и корректности работы;
* восстановление работоспособности системы после системного сброса по сторожевому таймеру (в случае зависания программного обеспечения);
* восстановление работоспособности системы после системного сброса по питанию;
* формирование блокирующих воздействий на управление прибора при неправильном обращении с ним;
* измерение диаметра шприца;
* изменение порогов срабатывания аварийной сигнализации в соответствии с установленными параметрами окклюзии;
* установка целевых значений скорости инфузии.

1. **Описание логической структуры**

ВПО построено на основе обработки событий, возникающих от внутренних систем микроконтроллера, от устройств дозатора и от действий оператора.

В *Таблице 1* приведен перечень файлов, входящих в проект, и их назначение.

*Таблица 1*

|  |  |
| --- | --- |
| **Наименование файла** | **Назначение файла** |
| 1 | 2 |
| Файлы текста программ на языке С. Проект Dozator\_1\_0.eww | |
| doz.c | Основной файл проекта, в котором реализована функция main () |
| doz\_a.c | Подпрограммы драйверы аппаратной части |
| doz\_ak.c | Подпрограммы обслуживания аккумулятора |
| doz\_k.c | Подпрограммы обслуживания клавиатуры прибора |
| doz\_r.c | Подпрограммы расчётов |
| doz\_bol.c | Подпрограммы реализации режима «Болюс» |
| doz\_inf.c | Подпрограммы реализации режима «Инфузия» |
| doz\_slr.c | Подпрограммы реализации режима «СЛР» |

В *Таблице 2* приведен перечень основных глобальных переменных и их назначение.

*Таблица 2*

|  |  |
| --- | --- |
| **Наименование** | **Назначение** |
| 1 | 2 |
| Глобальные переменные проекта Dozator\_1\_0.eww | |
| flagtakt | Флаг системного такта 2 мсек |
| flagsek | Флаг события секунда |
| codebutton | Код нажатой кнопки на клавиатуре |
| mode | Режим работы, выбранный оператором |
| diameter | Измеренный диаметр шприца |
| tip | Тип установленного шприца: 5, 10, 20 мл |
| Spreset | Расход, установленный оператором |
| Vsum | Суммарный объём, накопленный в процессе сеанса |
| stinf | Состояние процесса инфузии |
| stslr | Состояние процесса СЛР |
| stbol | Состояние процесса болюс |

Блок-схема ВПО, описывающая алгоритм работы программы, приведена в ниже (см. п. 7).

1. **Входные данные**

Входных данных нет.

1. **Выходные данные**

Выходных данных нет.

1. **Алгоритм работы программного кода**

Алгоритм работы программного кода представлен в виде блок-схем:

* Начало выполнения программы после включения питания.

И фоновый цикл.

Обработка

системного

такта

Флаг секунды ?

2??1

Обработка

секунды

Обработка

нажатой

кнопки

Кнопка ?

Инициализация

Начало

Флаг такта ?

* Обработка системного такта

вход

Режим ?

Процесс

Инфузии

такт

Процесс

СЛР

такт

Процесс

Болюс

такт

выход

* Обработка события секунда

вход

Режим ?

Обработка

Инфузии

секунда

Обработка

СЛР

секунда

Обработка

Болюса

секунда

выход

* Обработка нажатой кнопки

Выбор

режима

Ввод

параметров.

Измерение диаметра

шприца

Пуск-остановка

выбранного

процесса

вход

Анализ

кнопки

выход

Процесс

Инфузии

такт

выход

вход

Управление двигателем

Анализ нештатных ситуаций

Подсчёт суммарного объёма и вывод на индикаторы

вход

Обработка

Инфузии

секунда

выход

Проверка положения шприца

Индикация работы двигателя

вход

Процесс

СЛР

такт

выход

Управление двигателем

Анализ нештатных ситуаций

Подсчёт суммарного объёма и вывод на индикаторы

вход

Обработка

СЛР

секунда

выход

Проверка положения шприца

Индикация работы двигателя

вход

Процесс

Болюс

такт

выход

Управление двигателем

Анализ нештатных ситуаций

Подсчёт суммарного объёма и вывод на индикаторы

вход

Обработка

Болюса

секунда

выход

Проверка положения шприца

Индикация работы двигателя