

Для конечной точки 1 будем использовать режим вывода по прерыванию. В техническом задании оговорена частота выходного сигнала 0 – 300 Гц. Так как максимальная частота опроса по прерыванию равно 1 кГц (время опроса 1 мс), то этого достаточно для условий технического задания. Максимальный размер пакета для 1 конечной точки составляет 16 байт.

Для конечной точки 2 будем использовать режим ввода *bulk*. Пересылка данных от объекта управления в микроконтроллер осуществляется последовательно, и рассматривалась выше.

Сброс устройства реализован аппаратно таким образом, что все устройства будут сброшены в первоначальное состояние при отключении питания, поэтому по начальному адресу в микроконтроллере (с которого начинает выполняться программа) должна храниться процедура инициализации, далее микроконтроллер должен находиться в бесконечном цикле и выходить из него только по прерыванию.

4.2 Блок схемы алгоритмов программы функционирования модуля

На рисунке 4.1 представлена блок-схема макроуровня программы функционирования модуля. Блок «возможные прерывания» отмечен курсивом, так как прерывания возникают лишь в определенные промежутки времени. Фактически тело программы представляет собой бесконечный цикл без каких либо действий (ожидание возникновения прерывания). При этом ввод и вывод данных в хост осуществляется по запросу хоста, который вызывает прерывание *PDIUSB*D12. В регистре прерывания находится код транзакции (*IN*, *OUT*) и номер конечной точки.



Рисунок 4.1 – Макроуровень программы функционирования модуля

В качестве обработчика событий прерывания процесс чтения цифрового канала, блок-схема алгоритма представлена на Рисунке 4.2.

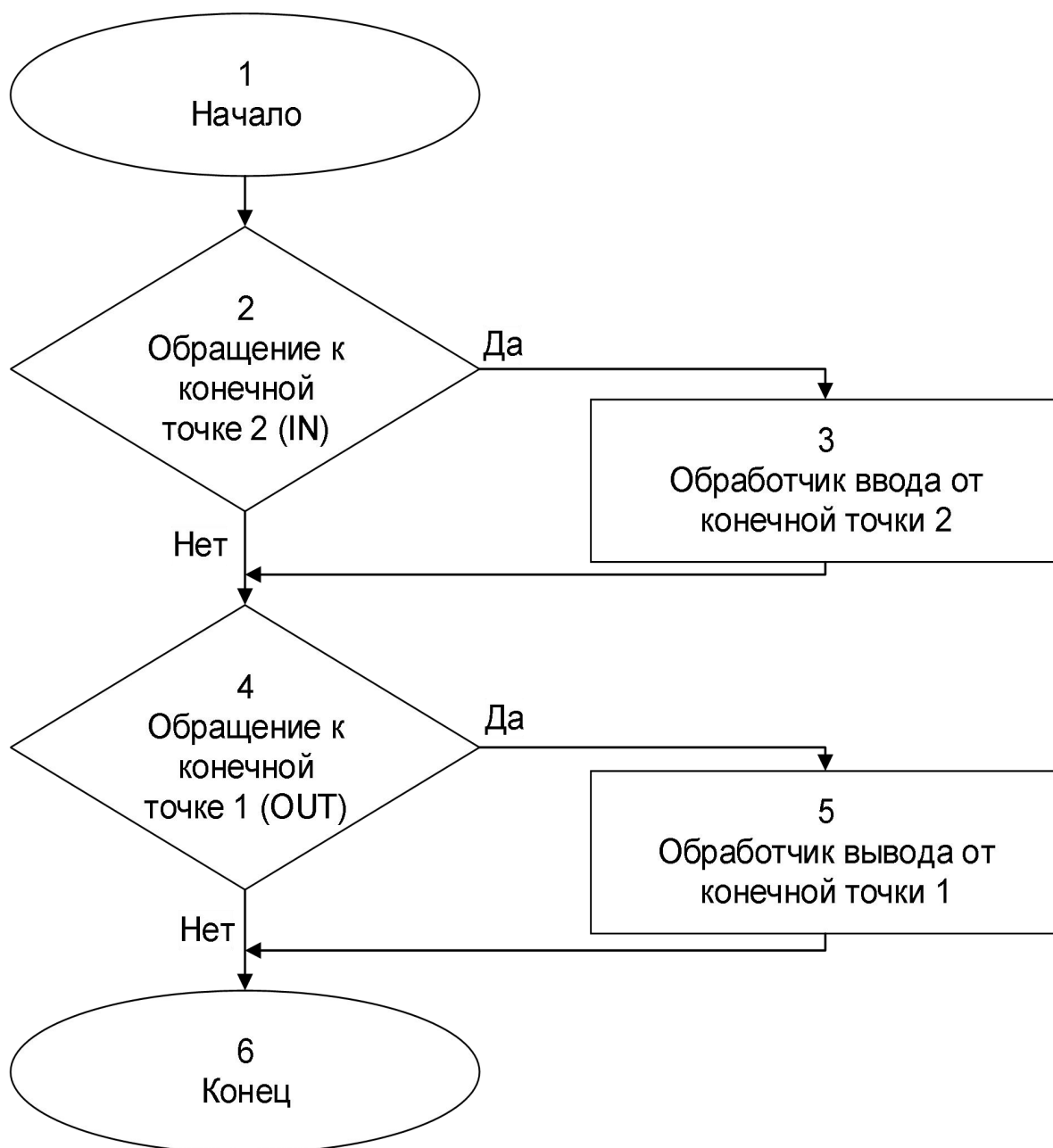


Рисунок 4.2 – Блок-схема обработчика прерываний.

Наличие обращения к той или иной конечной точке устанавливается путём анализа регистра прерываний микросхемы *PDIUSB12*. Обработчиков прерываний на ввод конечной точки 1 и на вывод конечной точки 2 не предусмотрено, так как для каждой из конечных точек (кроме нулевой) определён только один режим при конфигурации конечной точки

Рассмотрим обработку ввода информации подробнее (рисунок 4.3)



Рисунок 4.3 – Блок-схема ввода цифровой информации с ОУ

Условие «конечная точка доступна» и «буфер ввода пуст» проверяется путём анализа соответствующих битов, после выполнения команды «чтение состояния конечной точки». «Запись любого байта» – формальная операция, так как байт в буфере по нулевому смещению зарезервирован и может принимать любое значение [2].

Рассмотрим обработку вывода информации подробнее (рисунок 4.4)

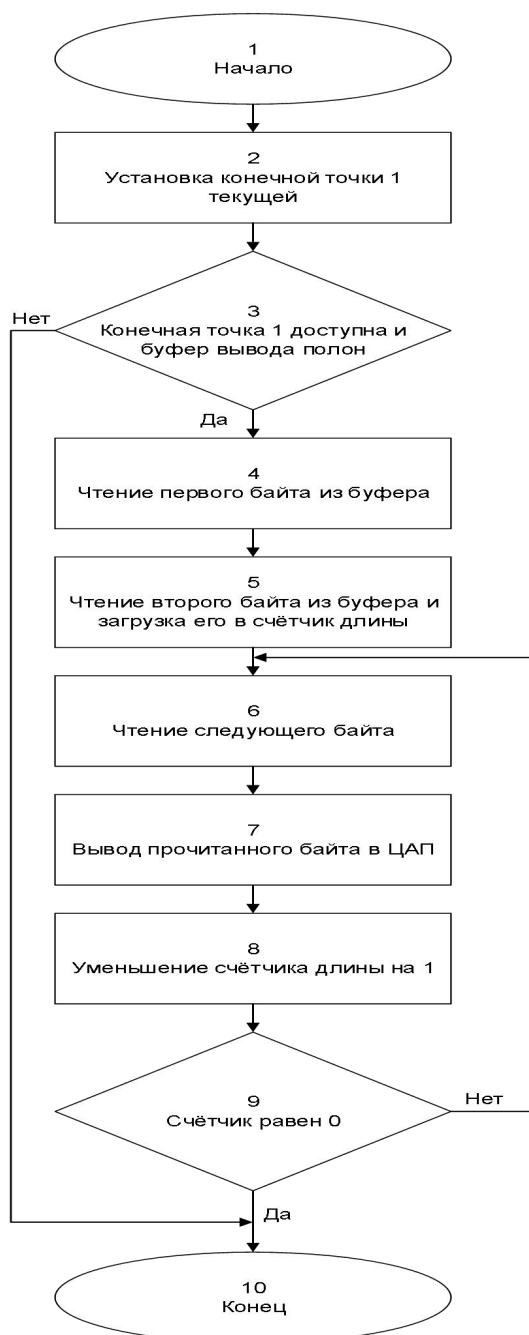


Рисунок 4.3 – Блок-схема вывода цифровой информации от хоста