

FPGA-конструктор советских микро-ЭВМ

ДВК-1

ДВК-2

ДВК-3

Электроника-60

Краткое руководство по созданию
собственных периферийных устройств.

Содержание

1. Введение.....3

1. Введение

Это проект разрабатывался в первую очередь как замена физически и морально устаревшего управляющего оборудования, все еще находящегося в эксплуатации на различных предприятиях. В 80-х — 90-х годах прошлого века контроллеры на основе PDP-11 — совместимого оборудования были весьма популярны и использовались в различных контрольно-измерительных приборах, системах ЧПУ, лабораторных стендах и много где еще. Сейчас это оборудование все еще частично эксплуатируется, и самым узким местом становится именно управляющий контроллер — древняя элементная база уже переработала все мыслимые сроки, и дальнейший ремонт становится весьма затруднительным.

В основе таких управляющих систем обычно находится какая-либо из плат микро-ЭВМ, чаще всего MC1201.02 или Электроника-60. К микро-ЭВМ через шину МПИ подключаются специализированные платы, стыкующие контроллер с управляемым оборудованием — аналоговыми и дискретными каналами ввода-вывода. В контроллер обычно или добавляется набор ПЗУ с управляющими программами, или программы грузятся с перфоленты или гибкого диска.

Предлагаемый, и уже опробованный мной подход предполагает не замену вышедших из строя плат в корзине МПИ, а замену всего контроллера целиком. Для этого придется разработать новую плату, уже на современных электронных компонентах, содержащую в своем составе все необходимые каналы для стыковки с оборудованием — ЦАП, АЦП, опторазвязки дискретных датчиков, реле, и все прочее. Плата эта подключается к портам FPGA, и вся обработка производится контроллером, схемотехнически реализованном уже внутри FPGA. Такой подход имеет массу достоинств. Можно выкинуть из электрошкафа громоздкую корзину МПИ и еще более громоздкий блок питания — новая схема будет компактной, и потреблять всего несколько ватт. Как следствие, контроллер будет совершенно холодным и не потребует систему охлаждающих вентиляторов, противно воющих в электрошкафе и собирающих из воздуха цеха грязь и пыль. Схема станет надежной — она построена на современной элементной базе, имеет минимум разъемных соединений.

В качестве платы FPGA можно использовать любую подходящую отладочную плату — их много продается и в Китае, и у локальных торгашей. Главное, чтобы стоящая на ней FPGA имела достаточно ресурсов для реализации схемы контроллера, и имела достаточное количество внешних портов.

Для реализации полноценного контроллера потребуется создать дополнительные модули, подключаемые к внутренней общей шине wishbone, и осуществляющие взаимодействие с оборудованием. Модули должны иметь тот же самый набор регистров и тот же протокол работы, что и оригинальные интерфейсные платы, работавшие ранее на шине МПИ в оригинальном контроллере. Этот документ дает краткое знакомство с процессом добавления модулей к проекту.

2. Постановка задачи

Процесс создания модуля будем разбирать на конкретном примере. Пример будет сильно упрощенным и избыточным по количеству регистров — он не предназначен для использования на практике, и нужен только как пример.

Предположим, что нам нужно сделать устройство двухканального дискретного ввода-вывода. Снаружи к модулю подключаются два исполнительных устройства, например два реле, и два дискретных датчика, например концевых выключателя. Назовем наш модуль просто и незатейливо — **dio**.

Модуль имеет на общей шине 2 регистра — регистр управления CSR и регистр данных DR. Биты регистра данных отражают состояние дискретных датчиков, а также управляют состоянием исполнительных устройств.

В случае появления логической 1 (активное состояние) на любом из входов модуль должен вызвать векторное прерывание.

Присвоим регистру управления адрес 175300, регистру данных адрес 175302. Прерывание будет использовать вектор 310.

Формат регистра CSR:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
									IE	IRQ					
									R/W	R/W					

Бит 6 (**IE**) — разрешение прерывания. Это бит есть в регистре CSR всех стандартных устройств.

Бит 5 (**IRQ**) — признак запроса прерывания. Если наступило событие, приводящее к прерыванию, но самого прерывания еще не произошло, это флаг устанавливается в 1. Запись 0 в этот разряд приводит к сбросу ожидающего прерывания.

Конечно, в данном случае регистр упрощен до предела и вообще не особо нужен. В более сложных устройствах обычно есть бит 15 (BSY) — признак занятости модуля, бит 0 (GO) — триггер запуска команды на выполнение, и бит 7 (DRQ) — признак готовности данных для чтения-записи процессором.

Формат регистра DR:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												O2	O1	I2	I1
												R/W	R/W	R	R

Биты 3 и 2 (**O2** и **O1**) — управление каждым из каналов дискретного вывода. Записанная в этот регистр 0 или 1 переключает уровень сигнала на выходе, например включает-выключает реле.

Биты 1 и 0 (**I2** и **I1**) отражают состояние соответствующих дискретных входов. Запись в эти разряды игнорируется.

3. Внешние порты модуля

Для стыковки с шиной wishbone модуль должен иметь стандартный набор сигналов этой шины. Кроме того, в список внешних портов включаются 2 линии ввода и 2 линии вывода дискретных данных. Вот заголовок нашего модуля:

```
module dio (
    input          wb_clk_i,    // тактовая частота шины
    input          wb_rst_i,    // сброс
    input [1:0]    wb_adr_i,    // адрес
```

```

input      [15:0]      wb_dat_i,    // входные данные
output reg [15:0]      wb_dat_o,    // выходные данные
input      wb_cyc_i,    // начало цикла шины
input      wb_we_i,    // разрешение записи (0 - чтение)
input      wb_stb_i,    // строб цикла шины
input      [1:0]       wb_sel_i,    // выбор байтов для записи
output reg          wb_ack_o,    // подтверждение выбора устройства

// обработка прерывания
output reg          irq,          // запрос
input              iack,          // подтверждение

// интерфейс ввода-вывода дискретных сигналов
output reg          do1,          // выходная линия 1
output reg          do2,          // выходная линия 2
input              di1,          // входной сигнал 1
input              di2           // входной сигнал 2
);

```

Из всей адресной шины нам требуются только 2 бита, поскольку в адресном пространстве модуль представлен двумя регистрами (диапазон адресов 175300-175303). Назначение всех сигналов шины и ее протокол можно найти в описании шины wishbone.

4.