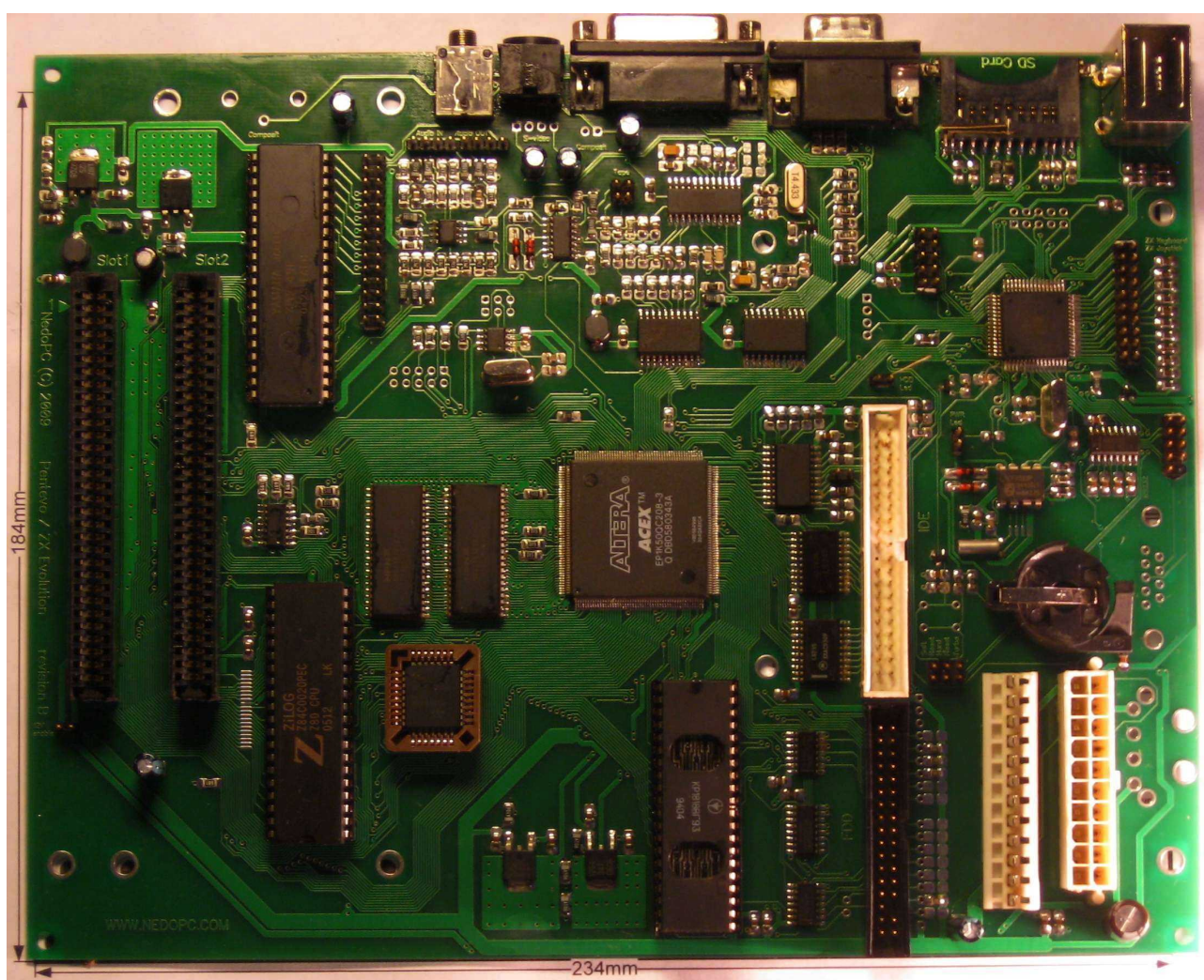


Базовая конфигурация для ZX Evolution.

Руководство пользователя.



(версия от 11.09.2010)

www.nedopc.com

Оглавление

1 Введение.....	3
2 Управляющие контакты компьютера.....	4
2.1 Hard reset (полный сброс).....	4
2.2 Soft reset (мягкий сброс).....	4
3 Дополнительные функции PS2 клавиатуры.....	4
3.1 Мягкий сброс компьютера (soft reset).....	5
3.2 Полный сброс компьютера (hard reset).....	5
3.3 Переключение режима TV/VGA.....	5
3.4 Индикация режима TV/VGA.....	5
3.5 Подача NMI на процессор Z80.....	5
4 Порты ввода-вывода.....	6
4.1 Порт Kempston Joystick.....	6
4.2 Порты Kempston Mouse.....	6
4.3 Порты управления часами и NVRAM.....	7
4.4 Чтение версии базовой конфигурации/бутлоадера.....	8
4.5 Доступ к SD-карте.....	9
4.6 Доступ к IDE-устройствам.....	10
5 История версий.....	13
5.1 Версия.....	13
5.2 Версия 30.03.2010.....	13
5.3 Версия 28.03.2010.....	13
5.4 Версия 24.03.2010.....	13
5.5 Версия 10.03.2010.....	13
5.6 Версия 04.02.2010.....	14

1 Введение

Под базовой конфигурацией компьютера ZX Evolution подразумевается конфигурация, которая поставляется и поддерживается производителем. Производитель осуществляет поддержку и развитие этой конфигурации и исправление найденных в ней ошибок. Группа NedoPC не поддерживает конфигурации сторонних разработчиков, в случае проблем с такими конфигурациями, обращайтесь непосредственно к их разработчикам.

Базовая конфигурация состоит из следующих компонент:

- Конфигурация для FPGA EP1K50QC208;
- Программа для микроконтроллера ATMEGA128.

Конфигурация поставляется в виде бинарного файла (zxevo_fw.bin), который можно загрузить в микроконтроллер ATMEGA128 с помощью BOOTLOADER¹. Обновление конфигураций, также производится с помощью BOOTLOADER.

Конфигурация FPGA интегрирована в программу микроконтроллера. Конфигурирование FPGA производится при включении или железном сбросе компьютера.

Изменение версии базовой конфигурации подразумевает изменение любой из или обеих компонент.

¹ BOOTLOADER — базовый загрузчик. Неизменяемая часть памяти микроконтроллера ATMEGA128, которая предназначена для загрузки или обновления памяти программ ATMEGA128.

2 Управляющие контакты компьютера

Управляющие группы контактов компьютера, предназначены для управления работой компьютера с помощью подсоединенных к ним элементов управления (например кнопок на корпусе компьютера).

2.1 Hard reset (полный сброс)

Группа контактов «Hard reset» осуществляет полный сброс компьютера (с перезапуском прошивки микроконтроллера и перепрограммированием FPGA) при замыкании контактов. Функционирование не зависит от используемой конфигурации компьютера.

Рекомендуется при использовании стандартного (AT или ATX) корпуса подсоединить к кнопке «Reset» корпуса.

Внимание! Hard Reset *не обеспечивает* сохранность содержимого памяти компьютера (например, RAM-дисков или страничек с сорцами в аласме).

2.2 Soft reset (мягкий сброс)

Функции группа контактов «Soft reset» зависят от конфигурации.

В базовой конфигурации на эту группу контактов возложены две функции:

- мягкий сброс компьютера (сброс процессора и внутренних регистров fpga);
- включение/выключение ATX блока питания.

Рекомендуется при использовании стандартного ATX корпуса подсоединить к кнопке «Power» корпуса. В этом случае включение компьютера осуществляется однократным нажатием кнопки. Отключение компьютера осуществляется длительным нажатием в течении 5 секунд.

Кратковременное замыкание контактов будет осуществлять мягкий сброс компьютера и установку нулевой страницы ROM.

Мягкий сброс обеспечивает сохранность содержимого памяти компьютера.

3 Дополнительные функции PS2 клавиатуры

При использовании PS2 клавиатуры с компьютером ZX Evolution можно использовать дополнительные возможности по управлению компьютером:

- мягкий сброс компьютера;
- полный сброс компьютера;
- переключение режима TV/VGA;
- индикация режима TV/VGA;
- подача NMI на процессор Z80.

3.1 Мягкий сброс компьютера (soft reset)

Мягкий сброс компьютера осуществляется с помощью кратковременного (менее 5 секунд) нажатия клавиши «F12».

3.2 Полный сброс компьютера (hard reset)

Полный сброс компьютера с клавиатуры можно осуществить двумя способами:

- Длительным нажатием клавиши «F12» (длительностью 5 сек), если используется АТХ блок питания, то будет осуществлено отключение питания;
- Нажатием комбинации «Ctrl-Alt-Del», отключение питания в этом случае не производится.

3.3 Переключение режима TV/VGA

Переключение режима TV/VGA осуществляется с помощью клавиши «Scroll Lock».

3.4 Индикация режима TV/VGA

Индикация режима осуществляется с помощью светодиода «Scroll Lock» клавиатуры. Светящееся состояние светодиода соответствует VGA режиму компьютера, отсутствие свечения — TV режиму.

3.5 Подача NMI на процессор Z80

Подача NMI на процессор Z80 осуществляется с помощью клавиши «Print Screen» клавиатуры.

4 Порты ввода-вывода.

Концепция портов для ZX Evolution существенно отличается от одной в обычных спектрумах. Прежде всего, дешифрация всегда происходит как минимум по 8 младшим битам (никаких портов #FC и подобного!). Существенно отличается логика блокирования портов картами ZX-Bus: если в спектрумах карты блокируют порты материнской платы (и друг друга, при наличии нескольких слотов), то в ZX Evolution, наоборот, все порты, присутствующие на материнской плате, не 'доходят' до карт ZX-Bus. Механизм IORQGE используется картами для блокирования друг друга (в порядке расположения в слотах). Отсюда следует важный вывод — любые карты, дублирующие какую-либо функциональность материнской платы (контроллеры клавиатуры, мышки, IDE, SD-карт и т. д.) работать *не будут*.

Далее приводится информация о портах.

Обозначения:

- «#xxFE» — порт дешифруется только по младшему байту адреса.
- «RO», «WO», «RW» - порт только для чтения, только для записи, для чтения и записи соответственно.
- «noshad» - порт не доступен из теневого режима.
- «always» - порт доступен всегда.
- «shadow» - порт доступен только в теновом режиме.

4.1 Порт Kempston Joystick

Kempston Joystick — пятикнопочный манипулятор. Позволяет управлять программами, в которых реализована поддержка этого манипулятора. Обращение к порту манипулятора происходит в безвайтовом режиме.

Порт	Описание								
#xx1F RO noshad	Чтение значения из порта позволяет получить состояние игрового манипулятора. Формат байта полученного значения:								
	<table><tr><td>7: 0</td><td>6: 0</td><td>5: 0</td><td>4: Fire</td><td>3: Up</td><td>2: Down</td><td>1: Left</td><td>0: Right</td></tr></table>	7: 0	6: 0	5: 0	4: Fire	3: Up	2: Down	1: Left	0: Right
	7: 0	6: 0	5: 0	4: Fire	3: Up	2: Down	1: Left	0: Right	
<ul style="list-style-type: none">• Fire – 1: нажата кнопка «Огонь» на манипуляторе, 0: не нажата;• Up – 1: нажата кнопка «Вверх» на манипуляторе, 0: не нажата;• Down – 1: нажата кнопка «Вниз» на манипуляторе, 0: не нажата;• Left – 1: нажата кнопка «Влево» на манипуляторе, 0: не нажата;• Right – 1: нажата кнопка «Вправо» на манипуляторе, 0: не нажата.									

4.2 Порты Kempston Mouse

Kempston Mouse — интерфейс манипулятора типа «мышь». Позволяет управлять программами, в которых реализована поддержка этого манипулятора. Обращение к порту манипулятора происходит в безвайтовом режиме.

Поддерживаются трехкнопочные мыши, а так же мышь с колесом.

Порт	Описание					
#FADF RO always	Чтение порта позволяет определить состояние кнопок и получить счетчик поворота колеса. Формат байта полученного значения:					
	<table><tr><td>7..4: Wheel counter</td><td>3: 1</td><td>2: MMB</td><td>1: RMB</td><td>0: LMB</td></tr></table>	7..4: Wheel counter	3: 1	2: MMB	1: RMB	0: LMB
	7..4: Wheel counter	3: 1	2: MMB	1: RMB	0: LMB	
<ul style="list-style-type: none">• Wheel counter – счетчик поворота колеса;• MMB – 0: нажата средняя кнопка мыши, 1: не нажата;• RMB – 0: нажата правая кнопка мыши, 1: не нажата;• LMB – 0: нажата левая кнопка мыши, 1: не нажата;						
#FBDF RO always	Чтение порта позволяет определить значение счетчика координаты X положения мыши.					
#FCDF RO always	Чтение порта позволяет определить значение счетчика координаты Y положения мыши.					

В случае если мышь не обнаружена, из портов считывается значение 0xFF.

4.3 Порты управления часами и NVRAM

Доступ и управление часами и NVRAM программно совместим с широкой распространенной в экс-ссср схемой Gluk Clock (основанной на микросхеме K512ВИ1 или DALLAS 12877, 12877A).

Порт	Описание
#EFF7 WO noshad	Запись в порт управляет доступом к ячейкам часов: <ul style="list-style-type: none"> установка бита 7 — разрешает доступ; снятие бита 7 — запрещает доступ.
#DFF7 WO noshad	Запись в порт устанавливает адрес ячейки (0x00..0x3F).
#BFF7 RW noshad	Чтение/запись ячейки.

Так как часы компьютера ZX Evolution основаны на микросхеме PCF8583, то совместимость обеспечивается путем эмуляции.

Особенности эмуляции:

- Поддерживается запись/чтение всех ячеек NVRAM (0x0E...0x3F);
- Поддерживается чтение всех ячеек, несущих дополнительные функции (0x00..0x0D);
- Поддерживается запись ячеек, управляющих временем:
 - 0x00 – регистр секунд;
 - 0x02 – регистр минут;

3. 0x04 – регистр часов;
 4. 0x06 – регистр дня недели;
 5. 0x07 – регистр дня месяца;
 6. 0x08 – регистр месяца;
 7. 0x09 – регистр года.
- Статусные ячейки возвращают следующие значения:
 1. 0x0A – возвращает значение 0x00;
 2. 0x0B – возвращает значение 0x02;
 3. 0x0C – возвращает значение 0x00 (поддерживается состояние update flag);
 4. 0x0D – возвращает значение 0x80.
 - Операция чтения значения ячейки является вайтовой и занимает небольшое кол-во тактов (чтение происходит из внутренних регистров ATMEGA128, дублирующих значения в PCF8583);
 - Операции записи значения ячейки является вайтовой и занимает достаточно большое кол-во тактов (цикл записи в ячейку включает этап записи значения в PCF8583).

4.4 Чтение версии базовой конфигурации/бутлоадера.

Прикладное ПО имеет возможность считывать версию/название базовой конфигурации и бутлоадера ZX Evolution. Обращение к этому функционалу происходит через порты управления часами и NVRAM (см. п4.3).

Записью значения по любому из адресов 0xF0..0xFF можно выбрать, что будет считываться. Допустимы следующие значения:

- 0 — версия базовой конфигурации;
- 1 — версия бутлоадера.

Длина данных версии составляет 16 байт (соответственно адреса ячеек 0xF0..0xFF) и имеет следующий формат:

- 0xF0..0xFB – 12 байт, название версии (если название короче 12 байт, то дополняется нулями);
- 0xFC..0xFD – 2 байт (LE), дата и бит релиза (если бит не установлен, то считается, что версия тестовая beta);
- 0xFE..0xFF – 2 байт (LE), CRC значение прошивки.

Формат упаковки даты и бита релиза:

- 7 бит 0xFD – 0: тестовая версия; 1: официальный релиз;
- 6..1 биты 0xFD – год выпуска релиза, значение 0..63 соответствует 2000..2063 годам;
- 0 бит 0xFD и 7..5 биты 0xFC – месяц выпуска релиза (допустимые значения 1..12);
- 4..0 биты 0xFC – день выпуска релиза (допустимые значения 1..31).

Пример:

50 65 6E 74 31 6D 00 00 00 00 00 00 7B 14 3C B1

50 65 6E 74 31 6D 00 00 00 00 00 00 — строка «Pent1m»;

7B 14 – beta, 10 год, 03 месяц, 27 день (27.03.2010);

3C B1 — CRC.

Какой полином CRC? Как его считать?

4.5 Доступ к SD-карте.

Происходит аналогично доступу в Z-контроллере от KOE. В обычном режиме:

Порт	Описание		
#xx77 RW noshad	Запись: управление сигналом \overline{CS} на SD-карту:		
	Биты 7..2: устанавливайте в 0 для совместимости	Бит 1: сигнал \overline{CS} , 1 после сброса, устанавливать в 0 для выбора SD-карты	Бит 0: устанавливайте в 1 для совместимости с Z-контроллером
	Чтение: всегда 0 (карта всегда вставлена и в режиме R/W — в соответствии с интерпретацией в Z-контроллере). Реальное наличие карты следует проверять попыткой её инициализации с таймаутом.		
#xx57 RW noshad	Запись: отсылка байта в SD-карту по SPI, одновременно принятый байт можно в дальнейшем считать из этого же порта.		
	Чтение: считать ранее принятый байт, отослать #FF в карту. Вновь принятый байт доступен при повторном чтении.		

Примечание: в цикле обмена по SPI, инициируемом записью или чтением порта #xx57, происходит одновременно отсылка байта в SD-карту и приём байта от неё. Отсылаемый байт тот же, что записан в этот порт (если цикл обмена инициирован записью), или #FF, если цикл обмена инициирован чтением порта.

Принятый байт запоминается во внутреннем буфере и доступен для последующего чтения из этого же порта. Данное чтение вновь инициирует цикл обмена и т. д.

Допускается читать/писать порт #xx57 командами **INIR** и **OTIR**. Пример чтения сектора:

```
LD    C, #57
LD    B, 0
INIR
INIR
```

SD-карта также доступна в shadow-режиме. В этом режиме порт #xx77 используется для других целей, потому остаётся только порт #xx57 и адресация меняется следующим образом.

Порт	Описание
#xx57 RW shadow	<p>Запись: если A15=1, то отсылка байта в SD-карту по SPI, одновременно принятый байт можно в дальнейшем считать из этого же порта — аналогично записи в #xx57 в noshade-режиме. Если A15=0 — управление сигналом \overline{CS}, аналогично записи в #xx77 в noshade-режиме.</p> <p>Чтение: считать ранее принятый байт, отослать #FF в карту. Вновь принятый байт доступен при повторном чтении. Чтение полностью аналогично noshade-режиму.</p>

Таким образом полноценный доступ к SD-карте сохраняется и в shadow-режиме. Запись командой **OTIR** 512 байт выглядит так (**OTIR** работает с предекрементом **B**):

```
LD    C, #57
LD    B, #80
OTIR
LD    B, #80
OTIR
LD    B, #80
OTIR
LD    B, #80
OTIR
```

4.6 Доступ к IDE-устройствам.

IDE-интерфейс сделан по стандарту nemo-ide с некоторыми расширениями. В отличие от оригинала (и Z-контроллера), IDE-порты доступны и в shadow-режиме. Однако, в отличие от оригинала, поддерживаются лишь порты **#10**, **#11**, **#30**, **#50**, **#70**, **#90**, **#B0**, **#D0**, **#F0** и **#C8**, при этом осуществляется полная дешифрация (по 8 младшим битам адреса). Расширенный режим передачи данных по сравнению с оригиналом заключается в возможности читать и писать сектора полностью через порт **#10** командами **INIR** и **OTIR**, причём в последнем случае нет надобности переставлять старший и младший байты. Расширенный режим не требует какого-либо специального включения, т. е. работает одновременно со старым (nemo-ide).

Базовая конфигурация для ZX Evolution

Порт	Описание
#xx10 RW always	Порт для записи или чтения данных, читается/пишется как младшая часть 16-битного слова (режим совместимости с nemo-ide), так и старшая тоже (расширенный режим).
#xx11 RW always	Чтение и запись старшей части 16-битного слова (режим совместимости с nemo-ide). Чтение происходит после чтения #10 , запись сюда — до записи в #10 . Обращение к этому порту не приводит к физическому обращению в IDE-устройства.
#xx30 RW always	Порт ошибок/свойств. Этот и дальнейшие порты см. в документации на протокол ATA. Биты ША Z80 7..5 маются на биты адреса шины ATA 2..0, порты #10 , #30 , ..., #F0 маются на $\overline{CS0}$ шины ATA, порт #C8 — на $\overline{CS1}$.
#xx50 RW always	Счётчик секторов
#xx70 RW always	Номер сектора (CHS) или биты 7..0 LBA-адреса
#xx90 RW always	Биты номера цилиндра 7..0 (CHS) или биты 15..8 LBA-адреса
#xxB0 RW always	Биты номера цилиндра 15..8 (CHS) или биты 23..16 LBA-адреса
#xxD0 RW always	Номер головки (CHS) или биты 27..24 LBA-адреса, выбор устройства (master/slave), выбор режима (LBA/CHS)
#xxF0 RW always	Регистр состояния/регистр команд
#xxC8 RW always	Регистр состояния/управления

Чтение данных в режиме nemo-ide выглядит так:

```

IN    A, (#10)    ;читаем младший байт, одновременно старший
                    ;защёлкивается в #11

LD     (HL), A
INC    HL
IN     A, (#11)    ;читаем ранее защёлкнутый старший байт
LD     (HL), A
INC    HL

```

Запись данных в режиме nemo-ide:

Базовая конфигурация для ZX Evolution

```
LD    D, (HL)
INC   HL
LD    A, (HL)
INC   HL
OUT   (#11), A    ;пишем в защёлку старший байт
LD    A, D
OUT   (#10), A    ;пишем младший байт одновременно со старшим
                     ;из защёлки
```

Запись и чтение в расширенном режиме:

```
LD    C, #10
LD    B, 0
INIR  ;Читаем сектор из #10 (2 раза по 256 байт)
INIR
...
LD    C, #10
LD    B, 0
OTIR  ;пишем сектор в #10 (2 раза по 256 байт)
OTIR
```

Внимание! В расширенном режиме синхронизация выдачи младшего или старшего байтов (а также записи сначала младшего, а потом старшего байта) в начале чтения или записи сектора осуществляется в момент доступа в порты **#30**, **#50** и т.д. (любые IDE-порты кроме **#10** и **#11**).

ВНИМАНИЕ! На момент написания доки расширенная функциональность контроллера IDE *не протестирована!* Так что багрепорты велком.

TODO: остальные порты

5 История версий

5.1 Версия

- Поддержка загрузки с магнитофонного входа.
- Скорректировано управление ресетами с PS/2 клавиатуры («F12» и «CTRL-ALT-DEL»).
- Реализован менеджер для всей памяти (ROM и RAM).
- Оптимизирована работа с PS/2 устройствами.
- Поддержан маппер PS/2 в ZX клавиатуру (пока без возможности редактирования).

5.2 Версия 30.03.2010

- Исправлена ошибка загрузки FPGA.

5.3 Версия 28.03.2010

- Исправлена обработка ошибок PS/2 мыши. Реинициализация мыши в случае ошибок обмена.
- Поддержка получения версии базовой конфигурации и бутлоадера;
- Исправлена обработка ошибок таймаута обмена с PS/2 устройствами.

5.4 Версия 24.03.2010

- Исправлено поведение светодиода питания (Pwr Led), теперь выключается если основное питание отключено.
- Управление сигналом NMI на ZX с помощью кнопки «Print Screen» PS2 клавиатуры;
- Мягкий сброс компьютера (без перезагрузки fpga) при замыкании контактов «soft reset» на плате.

5.5 Версия 10.03.2010

- Добавлена индикация режима TV/VGA на индикаторе «Scroll Lock» PS2 клавиатуры (горящий светодиод — VGA, не горящий TV);
- Исправлена ошибка отображения NVRAM, используемой в GLUK, в NVRAM PCF8583;
- Исправлена ошибка в эмуляции часов GLUK, реализована эмуляция Update flag в регистре C;
- Реализовано сохранение режима компьютера в NVRAM;
- Добавлена поддержка механической ZX клавиатуры;

Базовая конфигурация для ZX Evolution

- Добавлена поддержка Kempston джойстика.

5.6 Версия 04.02.2010

Базовая версия (1.00 в SVN).