

Ответы на 100 вопросов

1. Микроконтроллеру не нужна продвинутая обвязка, как внешняя память и т.д. Он может работать сам по себе.
2. сколько угодно. Обычно от 4 до 100, 200
3. 1, 2, 4 МГц от внутреннего генератора и от 0.1 до 16+ от внешнего
4. 1 МГц по умолчанию
5. ядро - периферия
6. tim0 , 1 , 2
7. tim1 имеет 4 16-битных и 4 8-битных регистра
8. 21
9. чем меньше номер, тем приоритетнее
10. sei / cli - разрешить / запретить
11. Режимы:
12. ldi r8, 0 << TOIE0; out TIMSK, r8; для запрета

```
13. #include <avr/io.h>
#include <avr/interrupt.h>

.global main
main:
    sei;

    sbi DDRA, DDA0
    cbi PORTA, PORT0

    ldi r17, 1 << PORTA0

    ldi r16, 1 << OCIE0
    out TIMSK, r16

    ldi r16, 0x7f
    out OCR0, r16

    ldi r16, 1 << WGM00 | 1 << CS00
    out TCCR0, r16

loop:
    rjmp loop

.global TIMER0_COMP_vect
TIMER0_COMP_vect:
    in r16, PORTA
    eor r16, r17
    out PORTA, r16
    reti
    ````
```

14. 1 8 64 256 1024

15. в любом

16. 0x00

17. Регистры PORTA, DDRA, PINA, DDR\_, PORT\_, PIN\_

18. чтобы инициализация мк прошла успешно и ничего ей не мешало

- 19.
- bit 1 - Z: бит нулевого результата
  - bit 2 - N: бит отрицательного результата
  - bit 3 - V: бит переполнения
  - bit 4 - S: бит знака
  - bit 5 - H: бит переноса полубайта
  - bit 6 - T: бит для копирования отдельного бита
  - bit 7 - I: бит прерывания

20. 8

21. BST r8, 5 ; 4ый бит r8 в 1ый r17

22. записать в 0-2 биты регистра TCCR0

23. 0 режим: нормальный режим. Счётчик считает от 0 до 255. При переполнении генерируется прерывание по переполнению; при совпадении с порогом - прерывание по сравнению

24. 1 режим: фазово-корректный ШИМ. Счётчик считает от 0 до 255 и обратно. Прерывание по переполнению генерируется при проходе нуля

25. 2 режим: сброс по совпадении. Счётчик считает от 0 до OCR0. Генерируется только прерывание по сравнению (другое только если OCR0 = 255, что полностью совпадает с режимом 0)

26. 3 режим: быстрый ШИМ. Счётчик считает как и в нормальном режиме. Прерывание по сравнению генерируется один раз за период (сам ШИМ не удастся сделать на прерываниях, но можно сконфигурировать OCR0, чтобы обнулялся при переполнении и сбрасывался при сравнении: получится желаемый ШИМ)

27. Можно, но это чревато пропуском возможного прерывания

28. Выставив биты 0-2 регистра TCCR0 в нули

29. 21 прерывание. При генерации разрешённого прерывания, все прерывания запрещаются глобально и исполнение переходит в вектор прерывания, обычно находящийся в начале программной памяти. При выходе командой reti, восстанавливается ход исполнения и включаются глобально прерывания. Прерывания генерируются выставлением бита в маске сгенерированных прерываний.

30. В ШИМ режимах

31. С делителя или детектора фронта

32. 65мкс

33. когда счётчик становится равным 0

34. Мк более многофункциональное устройство, тогда как у сигнального процессора ограничен спектр команд и периферии для более обработки сигналов

35. разрешить глобальные прерывания при обработке прерывания и правильно настроить приоритеты прерываний

36. т.к. не гарантируется его значения после reset

37. вызывает reset через определенное время, если не был перезагружен

38. 2.1 секунды

39. WDR

40. serial peripheral interface. Позволяет подключить периферийные устройства к МК по последовательной шине данных.

41. выставить уровень CS в логический ноль

42. прерывания - 1, регистра - 3: контрольный, состояния и данных

43.  $f/2$ , но это теоретически

44. Вопросы по 1-wire

45. 0 - низкое, 1 - высокое напряжение

46. 0 - импульс 60 мкс, 1 - импульс 15 мкс

47. 64 пакет. 8 на код семейства, 48 - серийный номер, 8 - контрольная сумма

48. Search ROM

49. долгий ноль 480 мкс, потом долго единица

50. как просто и надежно реализовать прием данных по UART, когда размер данных не известен. Те как `serial.read()` в Ардуино.