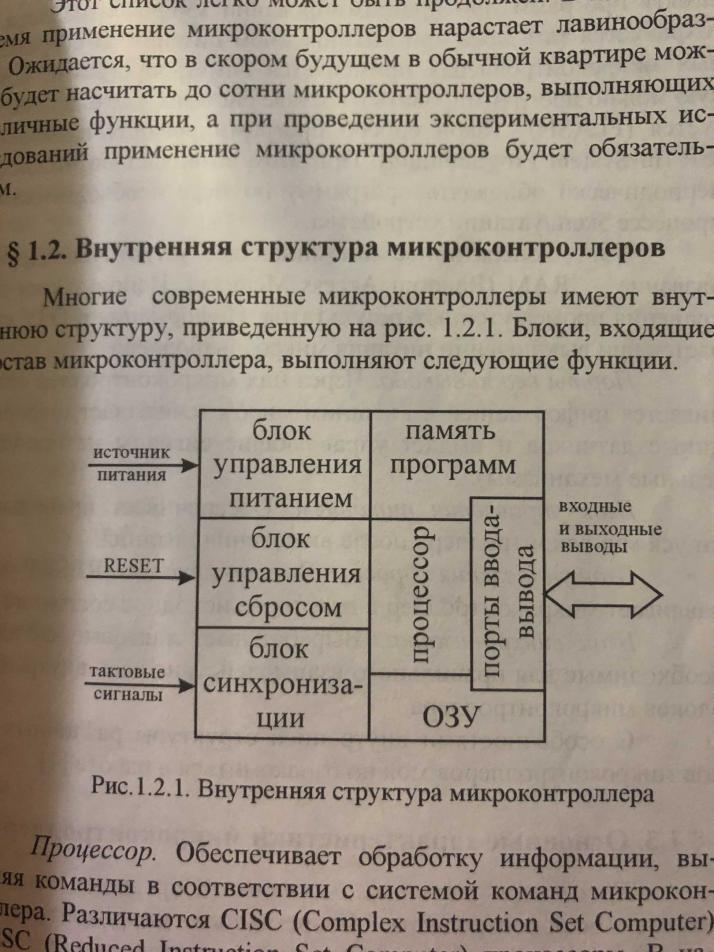
1. МП - чистое ядро, без таймеров, портов ввода-вывода, АЦП. МК это всё имеет

2. Произвольное количество. Обычно 8, 16, 32, 64, 128

3. 1, 2, 4 МГц от внутреннего генератора, от 0.1 до 16 МГц от внешнего

4. 1 МГц от внутренней RC-цепочки

5. 

6. Tim0 - 8-bit; Tim1 - 16-bit; Tim2 - 8-bit

7. Самый сложный: Tim1. Использует 4 16-битных регистра и 4 8-битных

8. 21 прерывание

9. Меньше номер - приоритетнее прерывание

10. sei(cli) - разрешить(запретить) прерывание глобально

11. Режимы: нормальный, со сбросом счётчика, ШИМ и фазово-подстроенный ШИМ; Два прерывания: по сравнению и переполнению; Именно таймеру принадлежит контрольный регистр, регистр-счётчик и регистр для порога. Так же по два бита в маске разрешённых прерываний и в регистре флагов прерываний

12. ldi r16, 1(0) << TOIE0; out TIMSK, r16 - разрешить(запретить) прерывание по переполнению

13.

#include <avr/io.h>

#include <avr/interrupt.h>

.global TIMER0\_COMP\_vect

TIMER0\_COMP\_vect:

in r16, PORTA

eor r16, r17

out PORTA, r16

reti

.global main

main:

sei ; interrupts enabled

sbi DDRA, DDA0

cbi PORTA, PORTA0

ldi r17, 1 << PORTA0

ldi r16, 1 << OCIE0

out TIMSK, r16

ldi r16, 0x7f

out OCR0, r16

ldi r16, 1 << WGM00 | 1 << CS00 ; phase-correct PWM

out TCCR0, r16

loop:

nop

nop

nop

rjmp loop

14. 1, 8, 64, 256, 1024

15. В любом: в не-ШИМ режимах достаточно поставить OC0 изменяться при совпадении с порогом; в ШИМ режимах достаточно выставить порог в половину максимального(скважность = 0.5)

16. Нули

17. Ни одного прерывания; три регистра: PORTA, DDRA, PINA; DDRn - на вход или выход работает вывод, PORTn - выходное значение, PINn - входное значение

18. Для того, чтобы корректно инициализировать работу МК

19. bit 0 - C: бит переноса

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 прерываний

20. 8 бит

21. BST r16, 4 ; 4-ый бит r16 -> 1-ый бит r17

BLD r17, 1

22. написать в биты 2:0 регистра TCCR0 значение от 1 до 5

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

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

25. 2 режим: сброс по совпадении. Счётчик считатет от 0 до OCR0. Генерируется только прерывание по сравнении (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 сравнении генерируется один раз за период (сам ШИМ не удастся сделать на прерываниях, но можно сконфигурировать OC0, чтобы обнулялся при

переполнении и сбрасывался при сравнении: получится желаемый ШИМ)

27. Можно, но это черевато пропуском возможного прерывания

28. Выставив биты 2:0 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 обоих ШИМ режимах таймера

31. Либо с предделителя, либо с PB0 после детектора фронта

32. 65 мкс

33. Когда счётчик равен нулю

34. Специфичный набор команд и регистров для уменьшения времени обработки сигналов, в то время, как у МК - основная задача - работа с периферией

35. При обработке прерывания разрешить глобально прерывания

36. Никто не гарантирует правильное значение после reset, хотя оно и нулевое. Но для верности выставляем, как надо - на конец памяти

37. Сторожевой таймер может перезагрузить МК если за время тайм-аута он не был сброшен. Время тайм-аута можно выставить от десятков мс до нескольких секунд

38. Максимально при питании 5В - 2.1 с

39. WDR

40. SPI - serial peripheral interface

41.передача начнется, как только будет записан байт в SPDR у MASTER .В результате MASTER  и SLAVE обменяются байтами и в каждом из них выработается прерывание SPI

42. Одно прерывание: SPI\_STC; 3 регистра: контрольный, состояния и под данные

43. f/2^n, где f - частота тактирования МК, n = 1..7, но подчинённый гарантировано работает при скорости < f/4

45. физически, 0 - низкое напряжение, 1 - высокое

46. логически, 0 - длинный импульс физического нуля (60 мкс), 1 - короткий (15 мкс)

47. 64 бита: 8 бит - код семейства, 48 бит - серийный номер, 8 бит - контрольная сумма - уникальный идентификатор устройства, чтобы можно было выбрать устройство

48. Search ROM

49. Долгий импульс нуля (480 мкс), потом долгий импульс единицы, в течении которой master проверяет, есть ли кто-нибудь в сети

50. Это режим счета по модулю, который определяется содержимым регистра OCR1A или ICR1. В этом режиме каждый входной импульс увеличивает содержимое счетчика на единицу