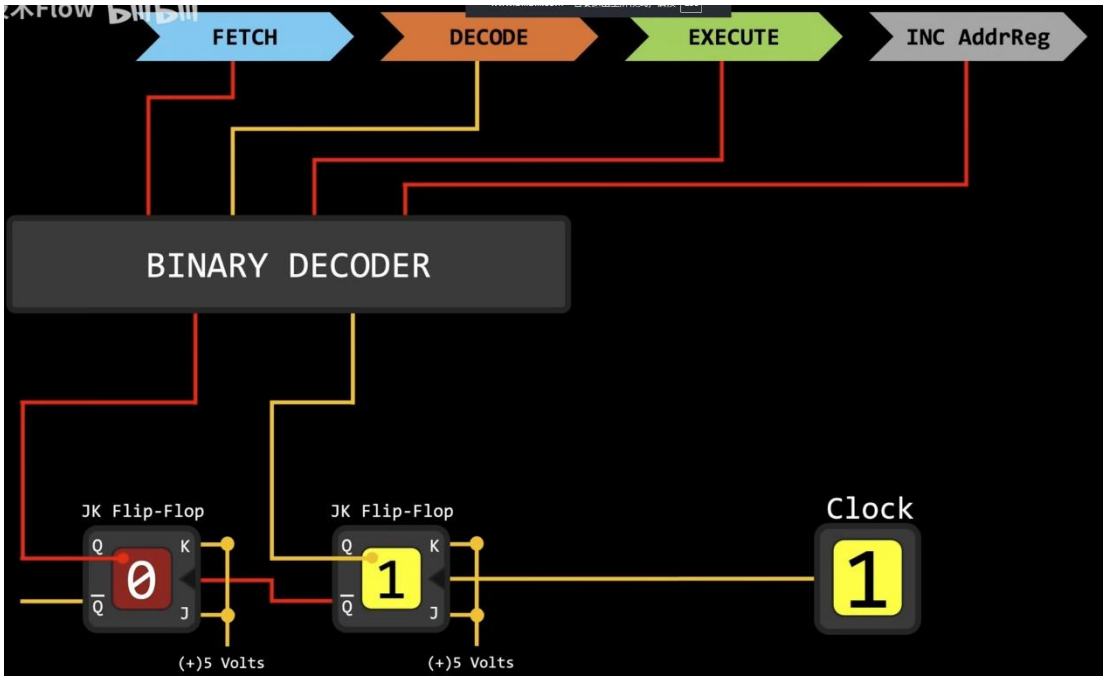
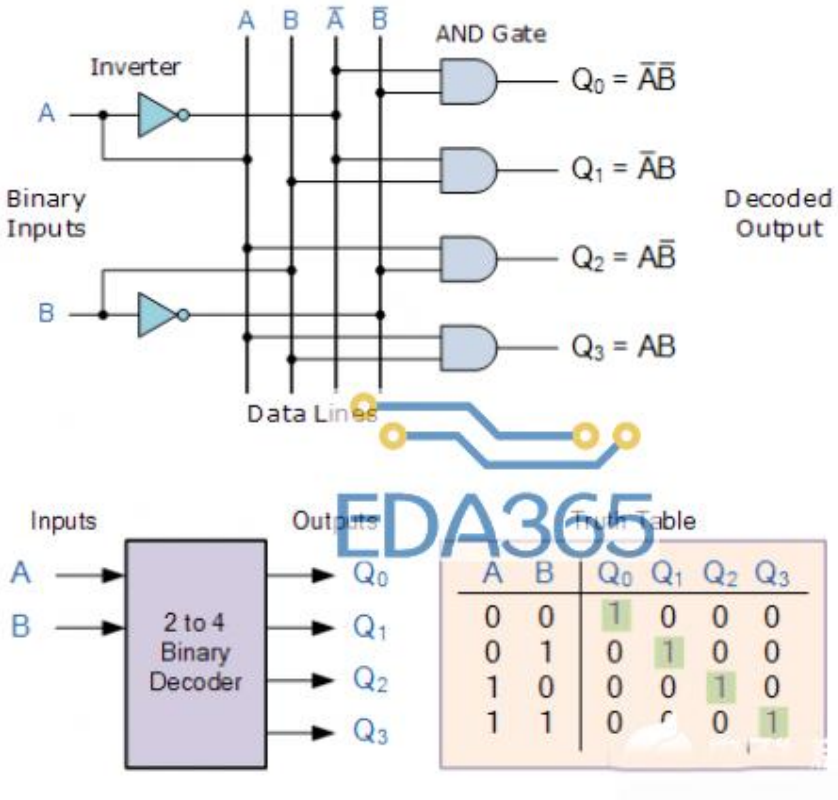
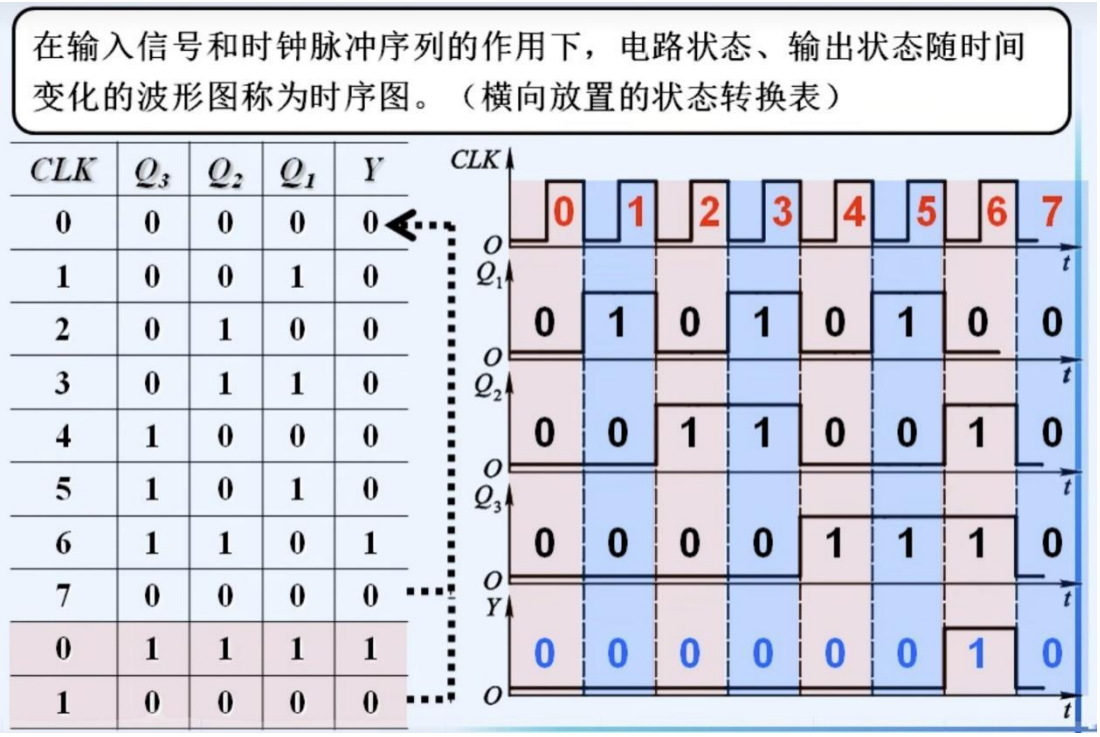
知识一：

取指，解码，执行，递增——



2-4 解码器





知识二：

单片机知识：什么是振荡周期、状态周期、机器周期和指令周期?

单片机是一种集成了微处理器、存储器、输入/输出接口和其他外围电路的微型计算机。单片机的工作速度和功能取决于它的时钟信号，指令集和数据通路。

为了更好地理解单片机的运行过程，我们需要了解以下几个重要的时间概念：

振荡周期：也称时钟周期，是指为单片机提供时钟信号的振荡源的周期，一般用 T 表示。振荡周期是单片机中最基本的时间单位，它决定了单片机能够完成的最小操作。振荡周期等于振荡频率的倒数，例如，如果单片机外接一个 6MHz 的晶振，那么一个振荡周期就是 1/6M 秒。

状态周期：每个状态周期为时钟周期的 2 倍，也称为节拍。一个状态周期是单片机内部数据传输和运算所需的最小时间单位。一个状态周期可以完成一个或多个微操作，例如，从寄存器到总线、从总线到寄存器、从寄存器到寄存器等。一个状态周期由两个时钟周期组成。

机器周期：一个机器周期包含 6 个状态周期，也称为基本操作或总线周期。一个机器周期是单片机完成一个基本操作所需的时间单位，例如，从存储器中读取或写入一个字节、从输入/输出端口中读取或写入一个字节等。一个机器周期由 6 个状态周期组成。

指令周期：执行一条指令所需要的时间，是从取指令、分析指令到执行完指令所需的全部时间。一个指令周期由若干个机器周期组成，不同类型和长度的指令所需的机器周期数不同。例如，在 8051 系列单片机中，有些简单的单字节指令只需要一个机器周期就可以完成，有些复杂的多字节指令则需

要两个或四个机器周期才能完成。

根据上述定义，我们可以得到以下关系：

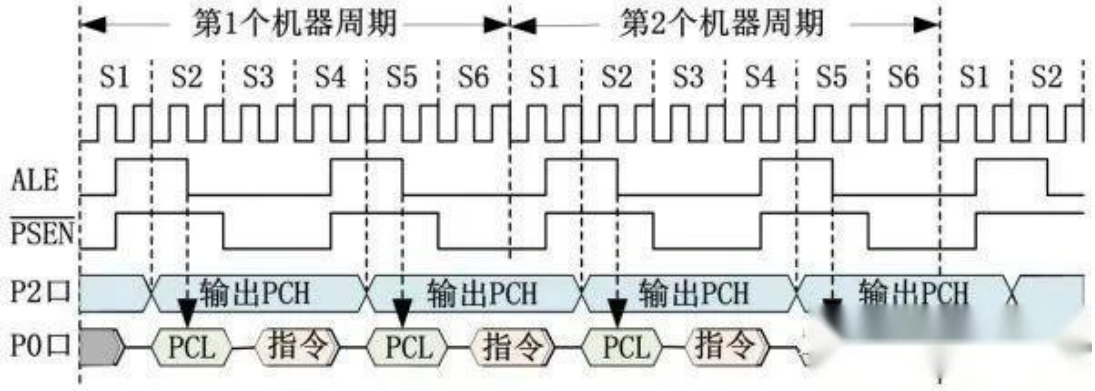
1 个振荡周期=1 个时钟周期

1 个状态周期=2 个时钟周期

1 个机器周期=6 个状态周期=12 个时钟周期

1 个指令周期=N 个机器周期=6N 个状态周期=12N 个时钟周期

这些时间概念有助于我们分析和优化单片机的性能和功耗。例如，我们可以通过提高振荡频率来缩短振荡周期，从而提高单片机的工作速度；我们也可以通过设计更简洁和高效的指令集来减少每条指令所需的机器周期数，从而提高单片机的执行效率；我们还可以通过采用流水线技术来重叠不同指令的不同阶段，从而提高单片机的吞吐量。



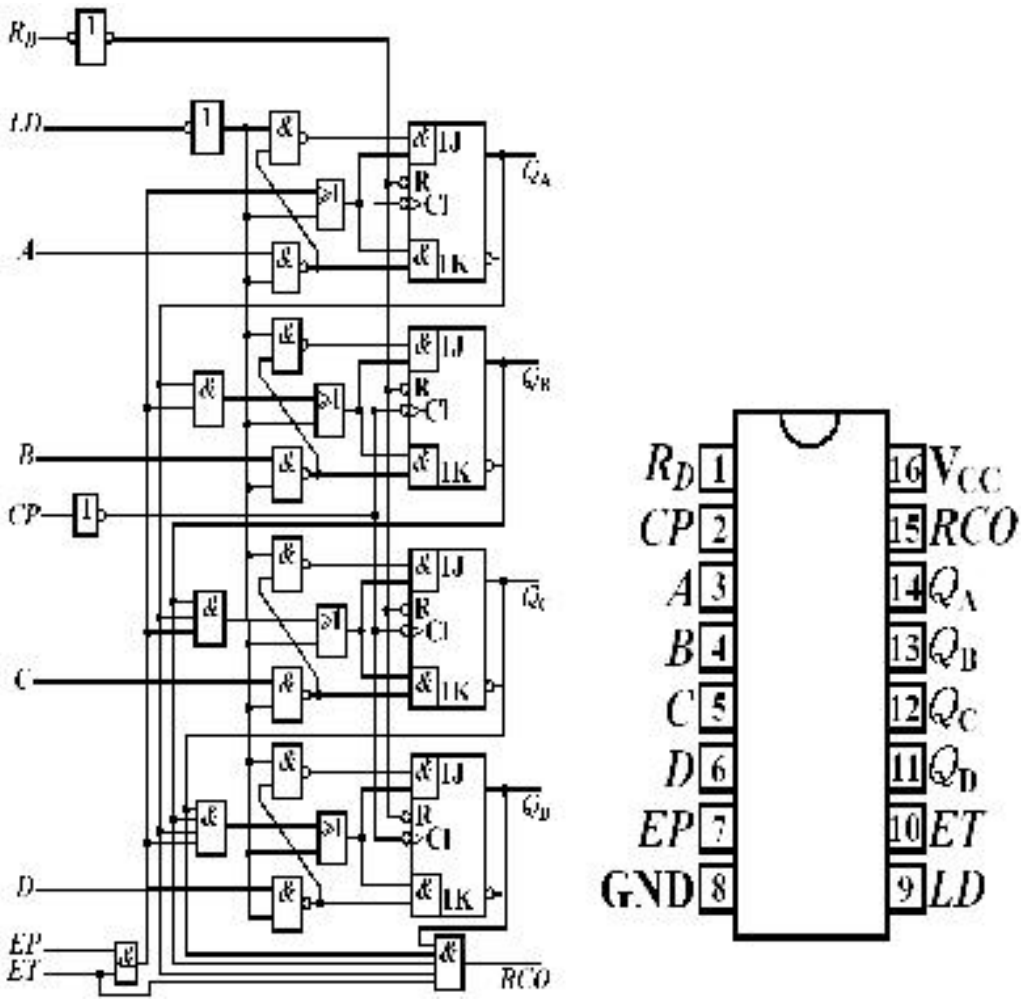
一般情况下，一个机器周期由若干个时钟周期组成或者说一个机器周期由若干个 S 周期（状态周期）组成。而一个 S 周期（状态周期）包括两个时钟周期。对于 8051 单片机来说，一个机器周期由 12 个时钟周期（振荡周期）组成，或者说由 6 个状态周期组成。

不用的指令的时钟周期不一样，8051 内核，同一指令的时钟周期也不一样，由下图就可以知道，MOV 指令将寄存器值放入累加器需要一个机器周期，MOV 指令将直接地址中的值放入累加器需要两个周期；同一指令，8051 和 90 系列单片机的指令周期也不同，8051 系列执行单指令需要 12 个时钟周期，90 系列执行单指令需要 6 个周期，他们都是需要一个机器周期，但是 90 系列一个机器周期只需要 6 个时钟周期。我们常用的 NOP 指令就是一个单指令，需要一个机器周期，8051 需要晶振产生 12 个脉冲。

知识三：

1.集成计数器 74161、74LS193、74LS290

（l）74161 的功能



（a）逻辑电路图　　（b）引脚图

图 7.1.11 74161 的逻辑电路图和引脚图

74161 是 4 位二进制同步加计数器。图 7.1.11a、b 分别是它的逻辑电路图和引脚图，其中凡是异步清零端，LD 是预置数控制端，A、B、C、D 是预置数据输入端，EP 和 ET 是计数使能（控制）端，RCO（$=ET\cdot Q_A\cdot Q_B\cdot Q_C\cdot Q_D$）是进位输出端，它的设置为多片集成计数器的级联　提供了方便。

表 7.1.4 74161 的功能表

清零	预置	使能		时钟	预置数据输入				输出			
R_D	LD	EP	ET	CP	A	B	C	D	Q_A	Q_B	Q_C	Q_D
L	×	×	×	×	×	×	×	×	L	L	L	L
H	L	×	×	∫	A	B	C	D	A	B	C	D
H	H	L	×	×	×	×	×	×	保			持
H	H	×	L	×	×	×	×	×	保			持
H	H	H	H	∫	×	×	×	×	计			数

表 7.1.4 是 74161 的功能表。由表可知，74161 具有以下功能：

①异步清零：当 R_D=0 时，不管其他输入端的状态如何（包括时钟信号 CP），计数器输出将被直接置零，称为异步清零。

②同步并行预置数：在 R_D=1 的条件下，当 LD=0、且有时钟脉冲 CP 的上升沿作用时，A、B、C、D 输入端的数据将分别被 Q_A～Q_B 所接收。由于这个置数操作要与 CP 上升沿同步，且 A～D 的数据同时置入计数器，所以称为同步并行预置。

③保持：在 R_D=LD=1 的条件下，当 ET·EP=0， 即两个计数使能端中有 0 时，不管有无 CP 脉冲作用，计数器都将保持原有状态不变（停上计数）。需要说明的是， 当 EP=0，ET=1 时，进位输出 RCO 也保持不变；而当 ET=0 时，不管 EP 状态如何，进位输出 RCOCO=0。

④计数：当 R_D=LD=EP=ET＝1 时， 74161 处于计数状态，其状态表与表 7.1.1 相同。

图 7.1.12 是 74161 的时序图。由时序图可以清楚地看到 74161 的功能和各控制信号间的时序关系。

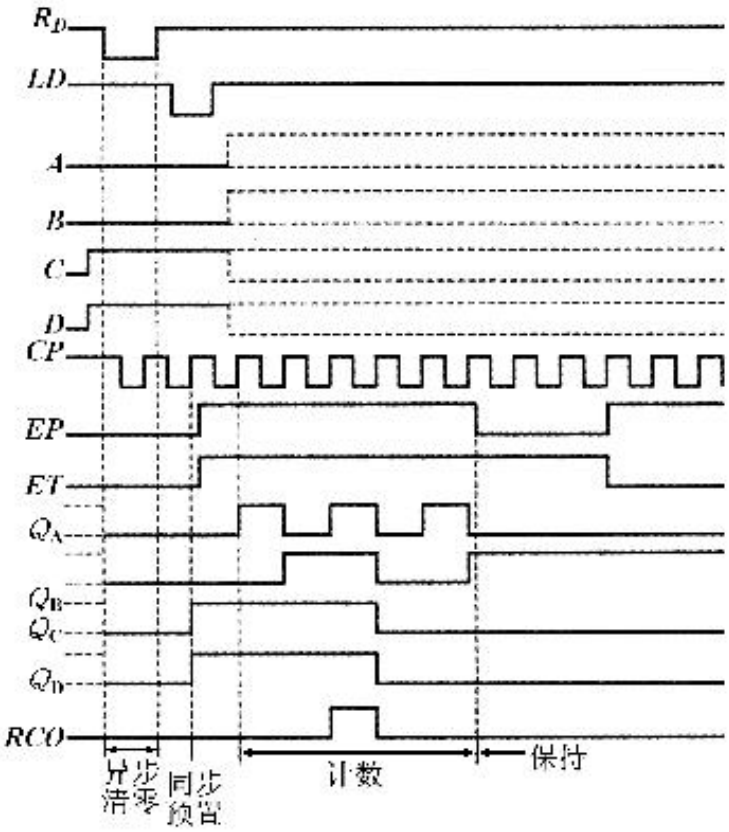


图 7.1.12 74161 的时序图

由图可知，首先加入一清零信号 R_D=0， 使各触发器的状态为 0，即计数器清零。R_D 变为 1 后，加入一置数信号 LD=0， 该信号需维持到下一个时钟脉冲的正跳变到来后。在这个置数信号和时钟脉冲上升沿的共同作用下，各触发器的输出状态与预置的输入数据相同（图中为 DCBA=1100），这就是预置操作。 接着是 EP=ET=1，在此期间 74161 处于计数状态。这里是从预置的 DCBA=1100 开始计数，直到 EP=0，ET=1，计数状态结束，转为保持状态，计数器输出保持 EP 负跳变前的状态不变，图中为 $Q_DQ_CQ_BQ_A$=010，RCO=0。

高速 CMOS 集成器件 74HC161、74HCT161 的逻辑功能、外形和尺寸、引脚排列顺序等与 74161 完全相同。

知识四:

指令字长是指一条机器指令中二进制代码的总位数，它决定了指令的复杂度和 CPU 执行效率。下面从定义、影响因素、与其他概念的关系和实际应用展开说明：

指令字长是什么？

- **核心定义**：指令字长是机器指令（CPU 执行的命令）的二进制位数，例如一条 16 位指令包含 16 个 0 或 1 的代码。
- **作用**：字长越长，指令能包含更多操作细节（如复杂运算或地址），但可能降低执行速度；字长短则效率高但功能受限。

哪些因素决定指令字长？

指令字长主要由三部分构成：

1. **操作码长度**：指定指令类型（如加法或跳转），通常占 4-8 位。
2. **操作数地址长度**：指向数据在内存中的位置，每个地址需 6-16 位（取决于寻址范围）。
3. **地址数量**：指令涉及的操作数越多（如双地址指令），字长越长。

指令字长和其他字长有什么区别？

1. ** vs 机器字长**：
    - 机器字长是 CPU 一次处理的数据位数（如 64 位 CPU），而指令字长是命令的位数，两者无必然联系。
    - 例如：64 位 CPU 可能用 32 位指令（半字长指令）简化操作。
2. ** vs 存储字长**：
    - 存储字长是内存单元存储的位数（如 8 位/字节），指令字长通常为其整数倍（如 16 位指令对应 2 字节存储）。
    - 固定指令字长时（如 RISC-V），两者相等。

2. 指令字长 (Instruction Length):

定义：指的是一条指令所占用的二进制位的数目。一条指令包含操作码（Opcode）和操作数地址（Operand Addresses)。操作码指定要执行的操作，操作数地址指定操作数在内存或寄存器中的位置。

影响：指令字长影响了指令集的复杂性和指令的执行效率。较长的指令字长可以支持更复杂的指令，但可能导致指令译码时间更长；较短的指令字长则相反。 一些体系结构使用变长指令，指令长度不固定。

相关寄存器：指令寄存器（IR）的位数通常与最大指令字长相同或略大于最大指令字长，以容纳各种长度的指令。

电路层面：指令字长决定了指令译码器的复杂度。指令译码器需要根据指令字长对指令进行解码，以确定操作码和操作数地址，从而控制 CPU 执行相应的操作。

80386 处理器一条指令通常需要多个时钟周期，一般在**几个到十几个时钟周期不等**，具体取决于指令的复杂性、操作数类型以及是否涉及内存访问，但早期的 80386 设计中，一条指令执行通常需要至少两个时钟周期（一个用于取指/译码，一个用于执行），而后面通过流水线优化（如到 486 时代）才能实现某些指令一个周期执行。

知识五：

时钟脉冲类型和作用介绍

时钟脉冲是什么信号

时钟脉冲是一种周期性重复的信号，用于定时和同步电子系统中的操作。它通常是一个方波信号，具有固定的频率和占空比。时钟脉冲主要用于同步处理器的操作和调度系统中的各种事件。

在数字电路中，时钟脉冲被用作触发器和寄存器的控制信号，用于同步和调度数据的传输和处理。它起到了分时和协调不同部件的作用，确保它们按照特定的时间顺序执行。

时钟速率决定了处理器和其他电子系统的工作速度，常用的时钟速率单位是赫兹（Hz）。较高的时钟频率通常意味着更快的数据处理能力，而较低的时钟频率则意味着更慢的处理能力。

时钟脉冲是用于定时和同步电子系统中操作的周期性信号，它起到了协调和同步不同部件的作用。

时钟脉冲有哪三种

在电子系统中，常见的时钟脉冲主要有以下三种类型：

1. 单相位时钟脉冲 （Single-phase clock）： 这是最常见的时钟类型，它包含高电平和低电平两个状态。数据传输和操作通常在时钟的上升沿或下降沿发生。单相位时钟脉冲常用于数字电路和处理器中。

2. 双相位时钟脉冲 （Dual-phase clock）： 双相位时钟有两个互补的时钟信号，通常称为时钟和反向时钟。这两个时钟具有 90 度的相位差，也就是说在一个时钟信号的上升沿或下降沿之间，另一个时钟信号正好位于相位差的另一半。双相位时钟脉冲可以用于高性能的时序电路、存储器以及许多通信接口。

3. 多相位时钟脉冲 （Multi-phase clock）： 多相位时钟脉冲是一种复杂的时钟类型，它由多个时钟信号组成，每个时钟信号具有不同的相位。多相位时钟常用于高速和高复杂度的系统，可以提高性能和减少功耗。

这些不同类型的时钟脉冲可以根据具体的应用要求灵活使用，以实现合适的数据传输和处理操作。

时钟脉冲怎么算

时钟脉冲的计算通常涉及两个主要参数：时钟频率和占空比。下面是一些计算时钟脉冲的常用公式：

1. 周期（Period）：周期是指时钟脉冲从一个边沿到下一个边沿的时间。周期可以使用以下公式计算：

周期 = 1 / 频率

其中频率是指每秒钟的时钟脉冲个数，单位为赫兹（Hz）。

2. 频率（Frequency）：频率是指每秒钟的时钟脉冲个数。频率可以使用以下公式计算：

频率 = 1 / 周期

其中周期是指时钟脉冲的时间长度，单位为秒（s）。

3. 占空比（Duty cycle）：占空比是指时钟脉冲高电平（或低电平）状态所占的比例。占空比通常以百分比表示。占空比可以使用以下公式计算：

占空比 = （高电平时间 / 周期） * 100%

其中高电平时间是指时钟脉冲处于高电平状态（逻辑“1”）的时间，周期已通过上述第一个公式计算。

根据具体的参数和要求，使用上述公式可以计算出时钟脉冲的周期、频率和占空比。这些计算可以帮助我们确定适合特定应用的时钟参数。

时钟脉冲的主要作用

时钟脉冲在电子系统中具有多种主要作用，包括：

1. 同步操作：时钟脉冲用于同步电子系统中的各个部件，确保它们在正确的时间进行数据的传输和处理。通过统一的时钟信号，不同部件在特定的时刻进行操作，避免数据冲突和混乱。

2. 时间控制：时钟脉冲提供定时机制，用于控制电子系统中各种操作和事件的发生时间。通过设定合适的时钟频率和占空比，可以精确地控制数据的传输速度和处理速度，确保系统的稳定性和可靠性。

3. 数据传输：时钟脉冲提供数据传输的时序，将数据按照时钟的上升沿或下降沿进行采样和传输。通过时钟信号的边沿触发，确保数据在正确的时间点被读取和写入，避免数据错误和不一致性。

4. 节拍控制：时钟脉冲用于控制处理器和其他部件的节拍，以确保它们按照特定的计时序列执行指令和操作。通过时钟信号的变化，控制处理器的时序逻辑，使得指令的执行在正确的时间发生，并且保持同步。

时钟脉冲在电子系统中起到了同步各个部件、控制时间、实现数据传输和节拍控制的重要作用。它是电子系统运行的基础，确保各个部件按照正确的时序工作，以实现正确、稳定和可靠的系统操作。