清华大学计算机科学与技术系

信号处理原理

贾珈

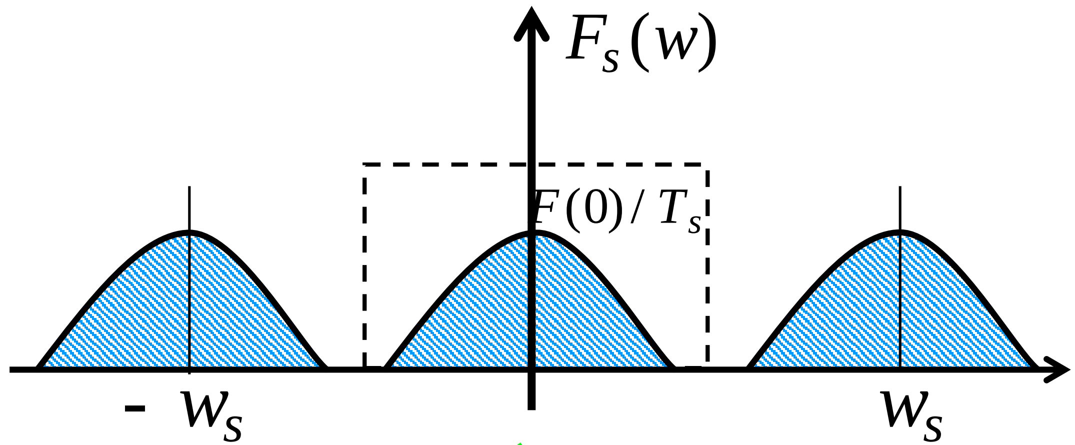
jjia@tsinghua.edu.cn

13651399048

2021.11.25

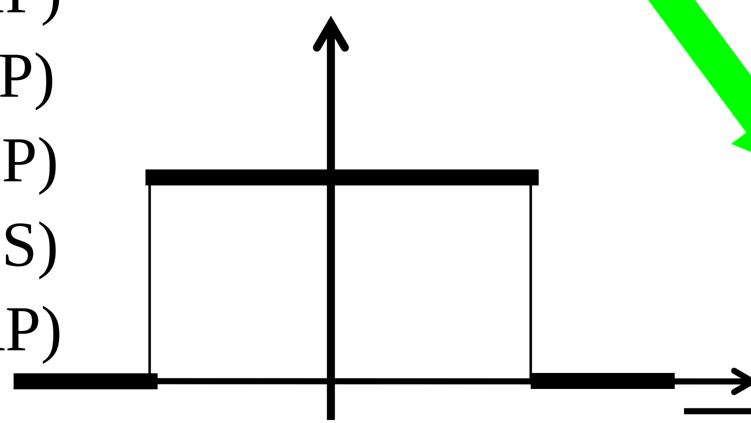
什么是滤波器?

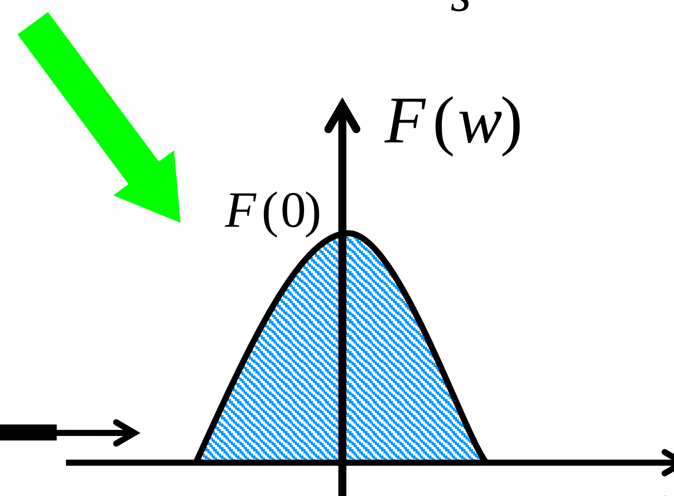
滤波器是以特定方式改变信号的频率特性，从而变换信号的处理系统。

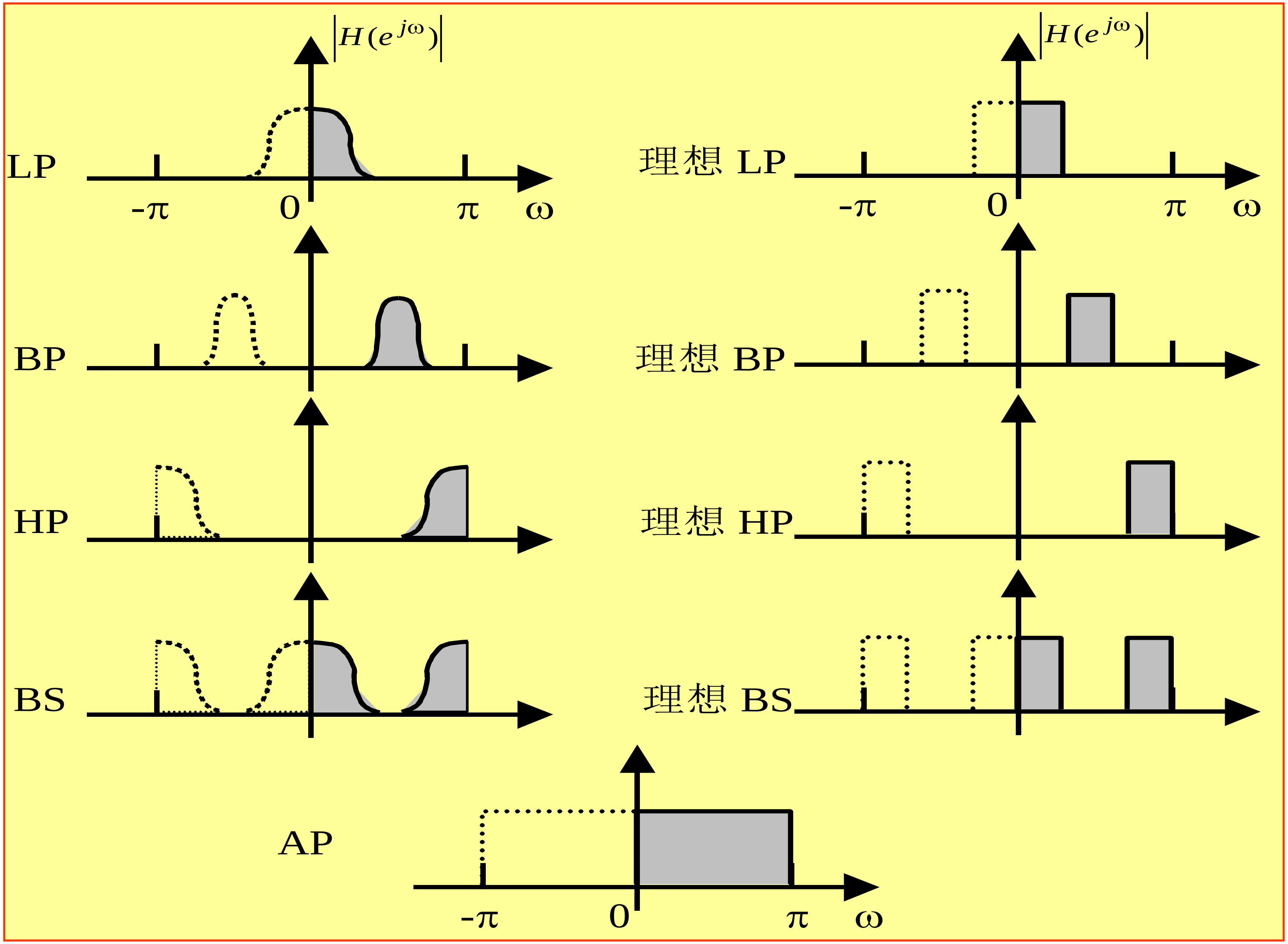


滤波器的类别?

高通滤波器(HP)
低通滤波器(LP)
带通滤波器(BP)
带阻滤波器(BS)
全通滤波器(AP)







$|H(e^{j\omega})|$
LP
-π
0
π
ω
BP
HP
BS
理想 LP
理想 BP
理想 HP
理想 BS
AP

模拟滤波器与数字滤波器

模拟滤波器是由电阻、电容、电感等部件构成的电路。滤波器特性对所用部件的物理标称值非常敏感，而且，有些部件的物理特性会随温度变化而改变。

数字滤波器是用软件实现的，很少依赖硬件。滤波软件只是一系列程序指令。虽然它是在硬件平台上运行，但是硬件平台本身并不决定滤波器的性能。数字滤波器的性能是由一组系数确定的。

数字滤波器的实现方式：

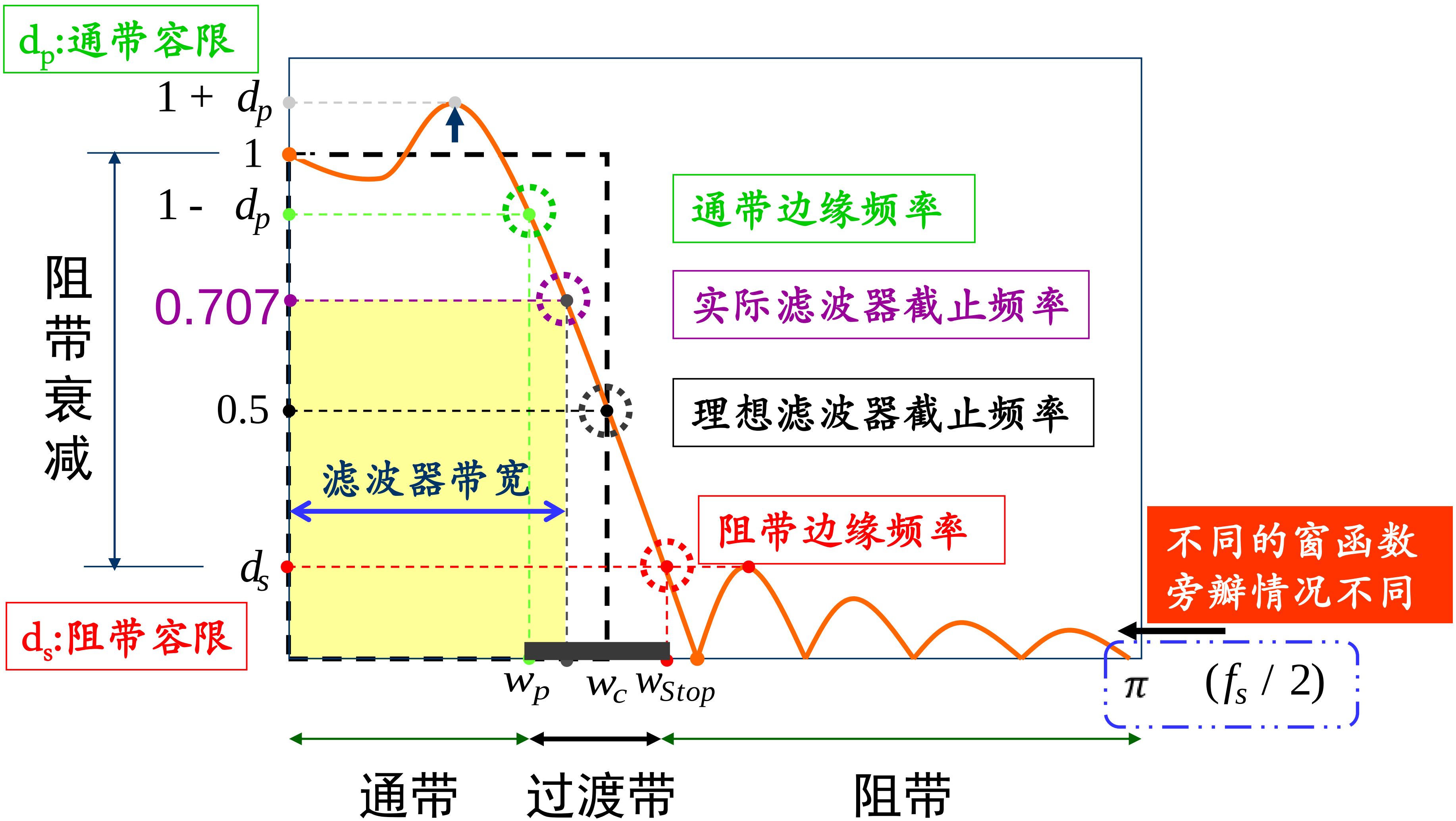
1 用流图计算滤波器的输出

2 用差分方程计算滤波器的输出

3 用卷积过程计算滤波器的输出

4 用DTFT直接改变信号频谱

滤波器的滤波特性参数（低通滤波器）



说明：虚线表示的是理想低通滤波器的滤波特性；实线表示的是窗函数法设计得到的实际的低通滤波器的滤波特性。

什么是系统?

系统是由若干相互作用和相互依赖的事物组合而成的**具有特定功能的整体**。

如计算机系统，医疗系统，雷达导航系统………

通俗地说，对信号进行处理的各种环境都可以被称为系统。在信号处理领域，系统是为了传送信号或对信号进行加工处理而构成的某种组合。

这种组合，既可以有对应的物理设备，如电容、放大器….

也可以是纯粹的算法，如计算机软件程序等。

关键是“特定功能的整体”

系统的分类

▶ 连续时间系统

- 输入和输出的信号都是连续时间信号，并且在系统内部也没有对信号进行转换的系统。

▶ 离散时间系统

- 输入和输出的信号都是离散时间信号，并且在系统内部，信号也是离散时间形式。

这两种不同的系统类别，它们的系统原理和分析方法，除少数内容外，基本上一致的。

本章主要介绍离散时间系统。

系统的分类

仅讨论线性、时不变、因果系统

线性系统

同时满足叠加性与齐次性的系统。

- 满足**叠加性**是指：当几个输入信号同时输入系统时，系统总的输出等于每个输入信号单独输入系统时的的输出信号和。
- 满足**齐次性**是指：当输入信号乘以某个常数时，系统的输出也倍乘相同的常数。

时不变系统

若系统无论何时收到输入，系统的输出都是相同的，则称系统为**时不变**系统。反之，则称为时变系统。

➔ 线性时不变系统

若系统既是线性的，又是时不变的，则我们称之为线性时不变系统。简记为LTI系统。

因果系统

如果系统的输出取决于现在和以前的输入数据，而与以后的输入数据无关，则称此系统为因果系统。

所有实际系统都是因果系统！

稳定系统

若系统的输入有界则输出也是有界的，则称此系统为稳定系统。这个性质通常被称为BIBO原则。

为什么要研究系统（滤波器）？

- 对于具有特定参数的系统，它具有何种处理信号的特性?
  - ◆即：分析信号在通过系统后，信号的性质会发生什么样的变化?
- 对于给定的传输和处理信号的要求，如何设计实现系统，使系统与其匹配?
  - ◆即：如何设计和实现系统，使系统满足信号处理的要求，系统应具有怎样的功能和参数?

系统的描述方法

用差分方程来描述线性、时不变、因果数字滤波器

示例

$$y(n)=\frac{1}{b_0}[a_0x(n)+a_1x(n-1)-b_1y(n-1)]$$

更一般的表达式

$$\sum_{k=0}^{N}b_ky(n-k)=\sum_{r=0}^{M}a_rx(n-r)$$

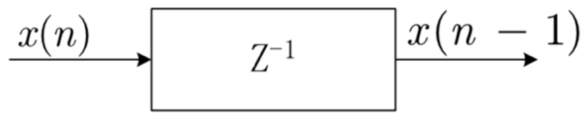
N为所需过去输出的个数，通常称为**滤波器的阶数**

为什么差分方程能表示系统呢？ →

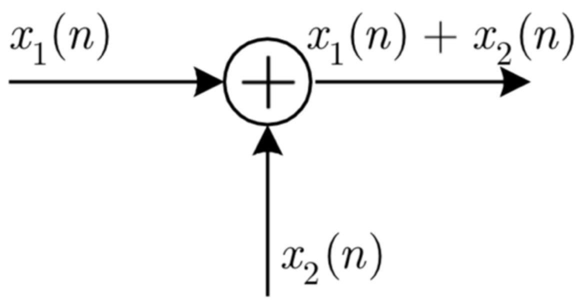
差分方程流图

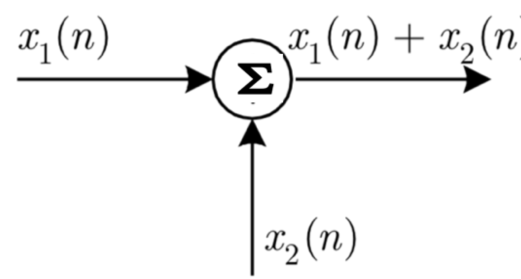
$$\sum_{k=0}^{N} b_k y(n-k) = \sum_{r=0}^{M} a_r x(n-r)$$

延时单元

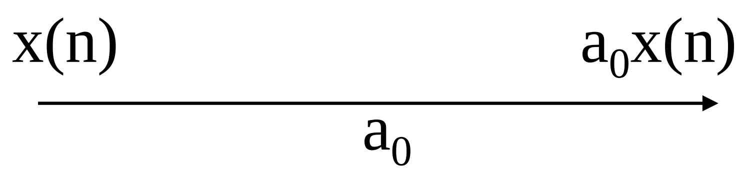


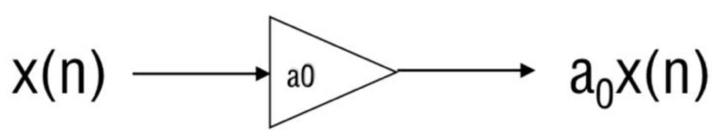
信号相加





乘上系数





主观题 1分

设置

【课堂练习1】画出下列差分方程的流图

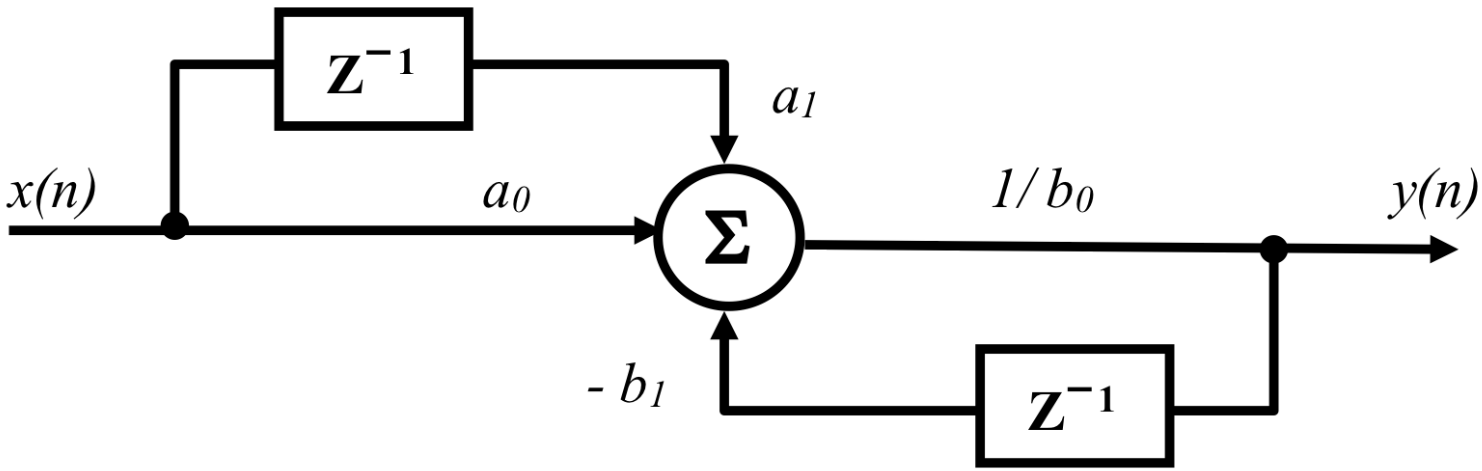
$$y(n)=\frac{1}{b_0}[a_0x(n)+a_1x(n-1)-b_1y(n-1)]$$

正常使用主观题需2.0以上版本雨课堂

作答

【课堂练习1】画出下列差分方程的流图

$$y(n) = \frac{1}{b_0}[a_0x(n) + a_1x(n-1) - b_1y(n-1)]$$



系统响应的分类

1 零输入响应

系统可能在没有给任何激励信号作用时，产生信号输出。

在这种情况下，系统的输出显然是与外界无关的(因为外界并没有给系统输入信号)，输出是由系统自身的内部信息引起的。

系统自身的内部信息，可能是先前激励（或扰动）作用的后果，不过，没有必要追究它们历史演变的详细过程，只需知道在当前激励接入系统时，系统瞬时状态即可。

由此看来，系统的零输入响应是一种纯粹由系统的起始状态所产生的响应。

2 零状态响应

系统在每一时刻都对应一种状态，开始研究系统的时刻系统所处的状态称为起始状态。

所谓系统的零状态响应，就是指系统在起始状态时状态值为零(相当于系统没有存储任何能量和信息)。

在这种状况下，给系统输入一个激励信号，则系统所产生输出响应就被称为系统的零状态响应。

滤波器的脉冲响应（冲激响应）

输入信号 → 系统 → 输出信号

激励 → 系统 → 响应

滤波器的脉冲响应（冲激响应），就是滤波器对脉冲输入的响应。即：当滤波器的输入为单位脉冲时，滤波器的输出就是单位脉冲响应。

单位冲激（单位脉冲）

$$\delta(n)=\begin{cases}1, & (n=0)\\ 0, & (n\neq 0)\end{cases}$$

内插：由样本值重建某一函数的过程。

- **理想内插:**以理想低通滤波器（频域矩形脉冲）的单位冲激响应（Sa函数形态）作为内插函数

设 $h(t)$ 为理想低通滤波器的单位冲激响应，则

$$x(t) = x_p(t) * h(t) = \sum_{n-\infty}^{\infty} x(nT)\delta(t-nT) * h(t)$$

$$= \sum_{n=-\infty}^{\infty} x(nT)h(t-nT)$$

函数与单位冲激函数的卷积

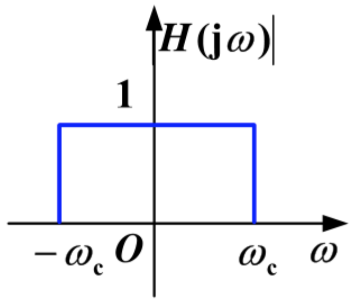
$$f(t) * \delta(t-t_0) = f(t-t_0)$$

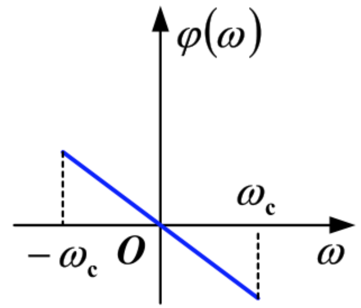
一个函数与单位冲激函数的卷积，等价于把该函数**平移**到单位冲激函数的冲激点位置。

理想低通滤波器（频域矩形脉冲）的单位冲激响应

是Sa函数形态 $h(t) = \frac{\omega_c}{\pi} Sa[\omega_c(t-t_0)]$

理想低通滤波器





$$H(j\omega)=\begin{cases}1\cdot e^{-j\omega t_0} & |\omega|<\omega_c\\ 0 & |\omega|>\omega_c\end{cases}$$

即

$$|H(j\omega)|=\begin{cases}1 & |\omega|<\omega_c\\ 0 & |\omega|>\omega_c\end{cases}$$

$$\varphi(\omega)=-\omega t_0$$

理想低通滤波器的单位冲激响应$h(t)$，即为其傅立叶逆变换。

例1. 求下列滤波器脉冲响应的前6个采样值

$$y(n) = 0.25[\,x(n) + x(n-1) + x(n-2) + x(n-3)\,]$$

用$\delta(n)$代替 $x(n)$, $h(n)$代替$y(n)$，则有：

$$h(n) = 0.25\left[\delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)\,\right]$$

$$\boxed{\delta(n) = \begin{cases} 1, & (n = 0) \\ 0, & (n \neq 0) \end{cases}} \longrightarrow$$

$$h(0) = 0.25 \quad h(1) = 0.25 \quad h(2) = 0.25$$

$$h(3) = 0.25 \quad h(4) = 0 \quad h(5) = 0$$

主观题 1分

设置

【课堂练习2】求下列滤波器脉冲响应的前6个采样值（设滤波器是因果系统，即脉冲响应在n=0之前为零）

$$y(n)-0.4y(n-1)=x(n)-x(n-1)$$

正常使用主观题需2.0以上版本雨课堂

作答

【课堂练习2】求下列滤波器脉冲响应的前6个采样值

$$y(n)-0.4y(n-1)=x(n)-x(n-1)$$

设滤波器是因果系统，即脉冲响应在n=0之前为零。

用 $\delta(n)$ 代替 $x(n)$，$h(n)$ 代替 $y(n)$，则有：

$$h(n)-0.4h(n-1)=\delta(n)-\delta(n-1)$$

$$\delta(n)=\begin{cases}1, & (n=0)\\ 0, & (n\neq 0)\end{cases}$$

$h(0)=1.0$ $h(1)=-0.6$ $h(2)=-0.24$

$h(3)=-0.096$ $h(4)=-0.0384$ $h(5)=-0.01536$

有限脉冲响应FIR和FIR滤波器

$$y(n) = \sum_{k=0}^{M} a(k)x(n-k)$$

上述系统的脉冲响应在有限个非零采样值后下降到零。这种响应被称为有限脉冲响应（finite impulse response, FIR），这种滤波器称为FIR滤波器。

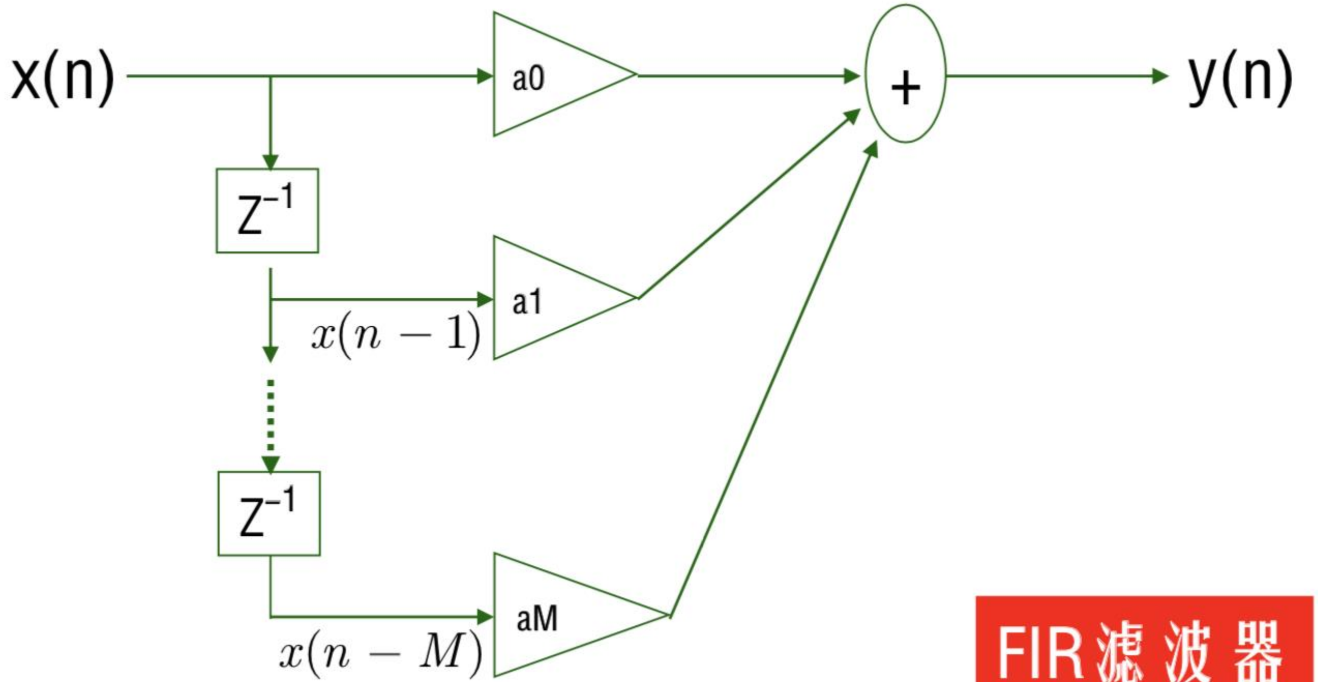
无限脉冲响应IIR和IIR滤波器

$$y(n) = \sum_{k=0}^{M} a(k)x(n-k) + \sum_{k=1}^{N} b(k)y(n-k)$$

显然，新的输出取决于过去的输出，所以脉冲响应永远不会消失，这种响应被称为无限脉冲响应（infinite impulse response, IIR），这种滤波器称为IIR滤波器。

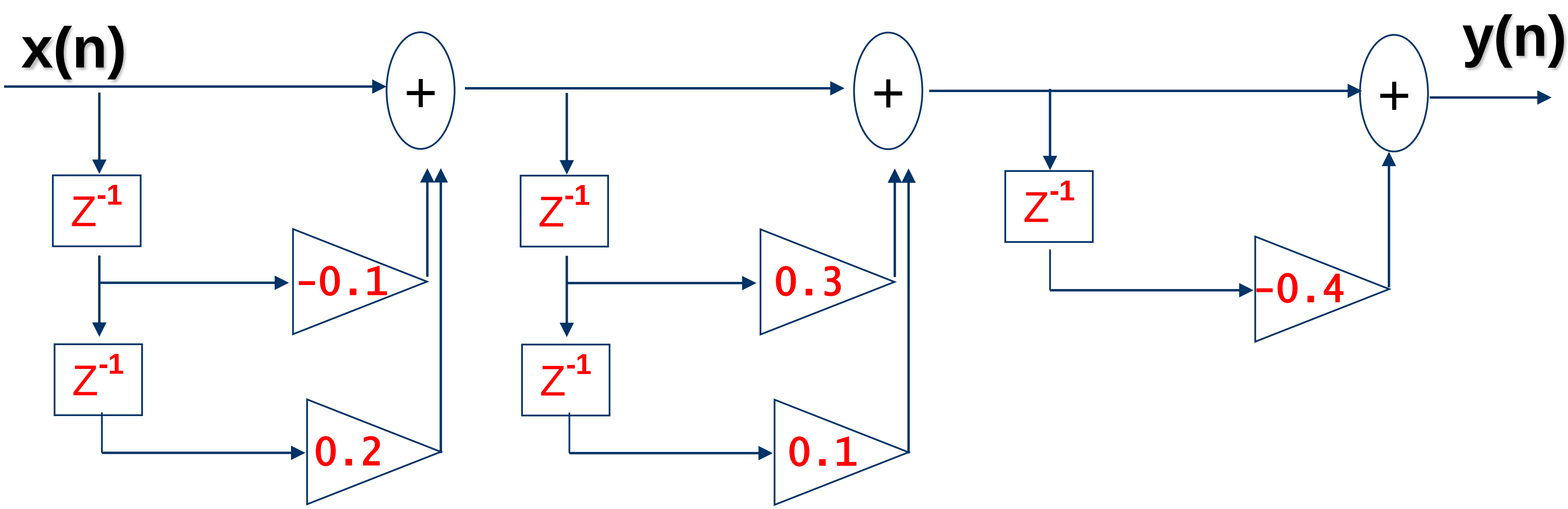
1. 差分方程 --- 非递归的差分方程

$$y(n) = \sum_{k=0}^{M} a(k) x(n-k)$$

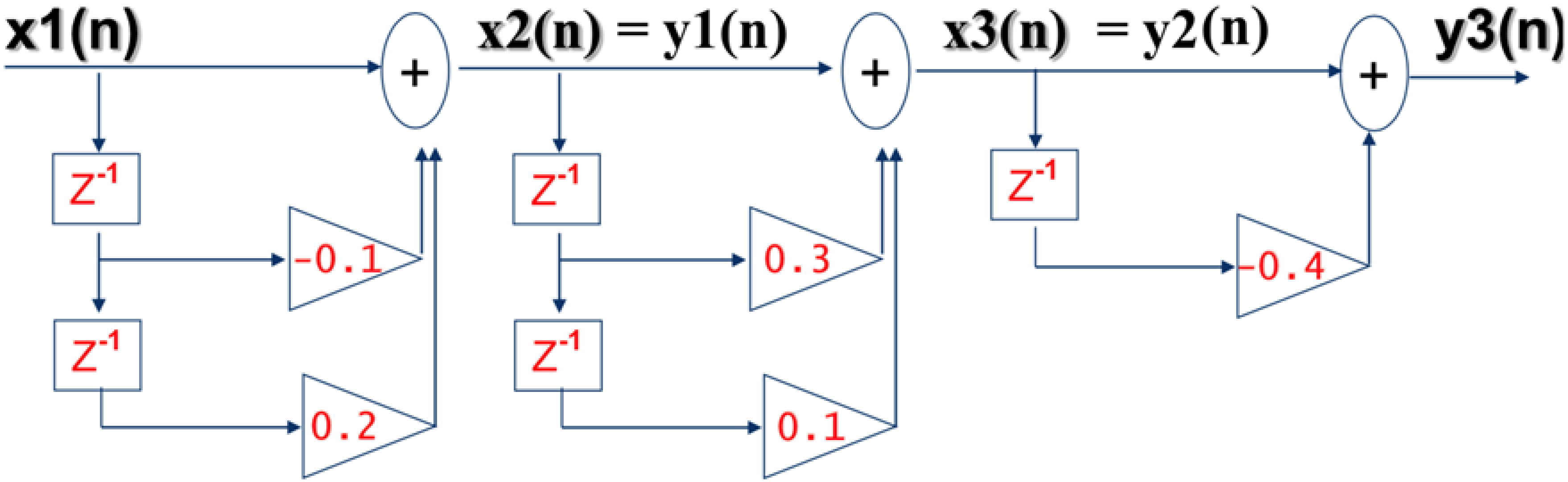


FIR滤波器

【课堂练习3】写出如图所示级联流图的差分方程



【课堂练习3】写出如图所示级联流图的差分方程



$$y_1(n) = x_1(n) - 0.1x_1(n-1) + 0.2x_1(n-2)$$

$$y_2(n) = x_2(n) + 0.3x_2(n-1) + 0.1x_2(n-2)$$

$$y_3(n) = x_3(n) - 0.4x_3(n-1)$$

$$\boxed{\begin{aligned} y_3(n) = {} & x_1(n) - 0.2x_1(n-1) + 0.19x_1(n-2) \\ & - 0.058x_1(n-3) - 0.008x_1(n-5) \end{aligned}}$$

滤波器差分方程流图优化

要求：

（1）减少存储

（2）减少运算（乘法、加法）

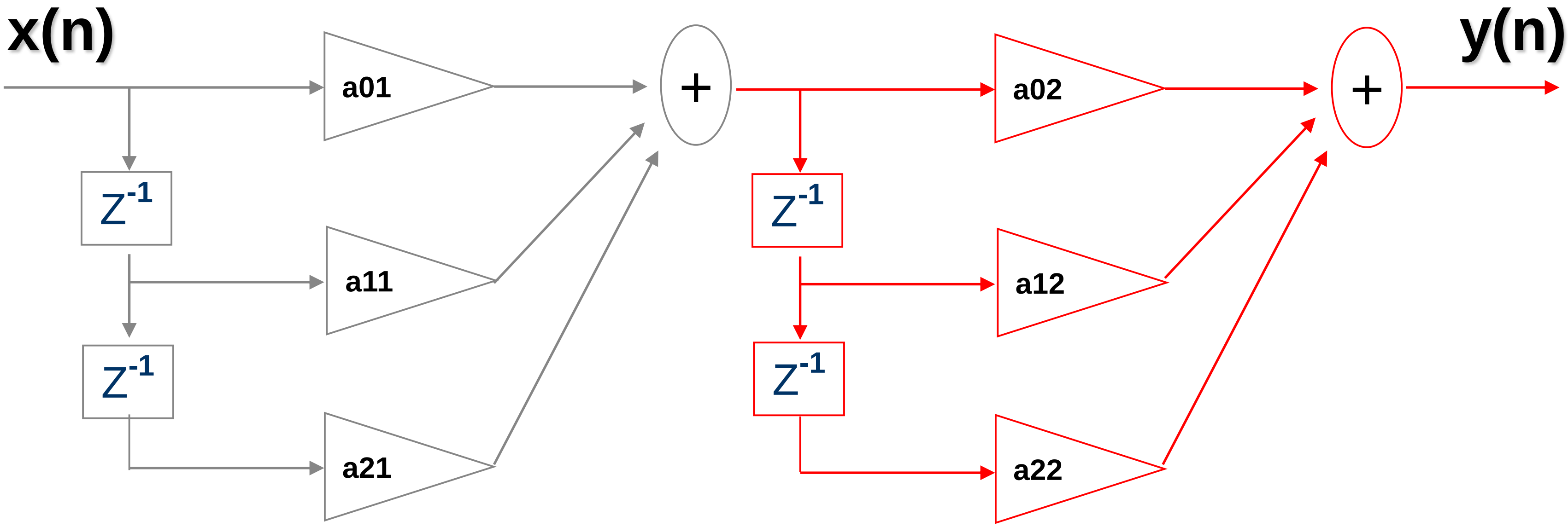
（3）减少有效字长效应（滤波器的系数必须量化，而处理器的有效比特数有限而产生的影响）

➔

高阶滤波器➔多个二阶滤波器节的级联

（滤波器系数大，对量化误差的敏感程度低）

非递归差分方程－－二阶 非递归 滤波器节 级联



$$y_1(n) = x_1(n) - 0.5x_1(n-1) + 0.1x_1(n-2)$$

$$y_2(n) = x_2(n) + 0.4x_2(n-1) + 0.2x_2(n-2)$$

$$y(n) = x(n) - 0.1x(n-1) + 0.1x(n-2)$$

$$-0.06x(n-3) + 0.02x(n-4)$$

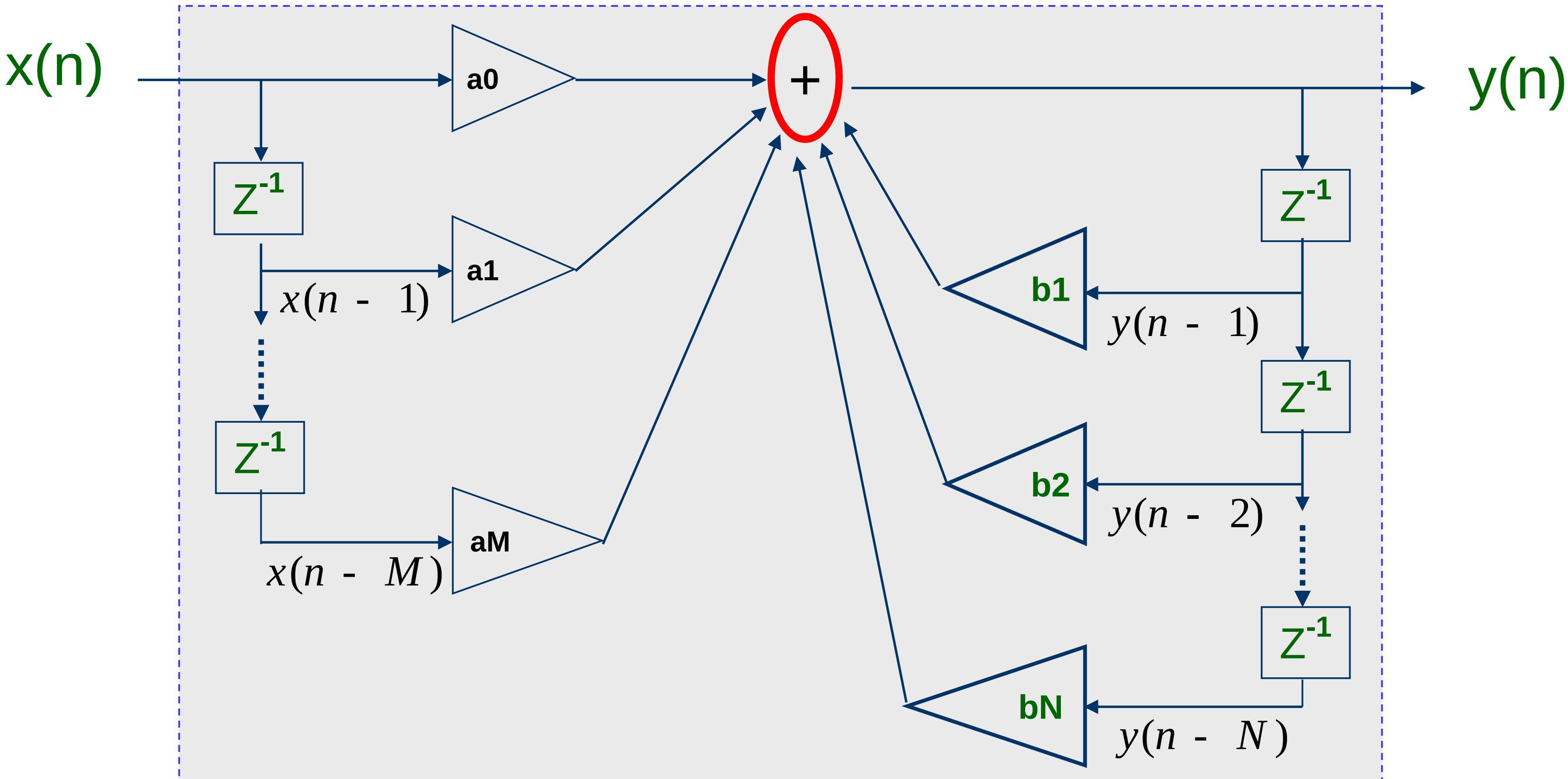
分级后，各滤波器节系数变大，对量化误差的敏感度降低

2. 差分方程 --- 递归差分方程

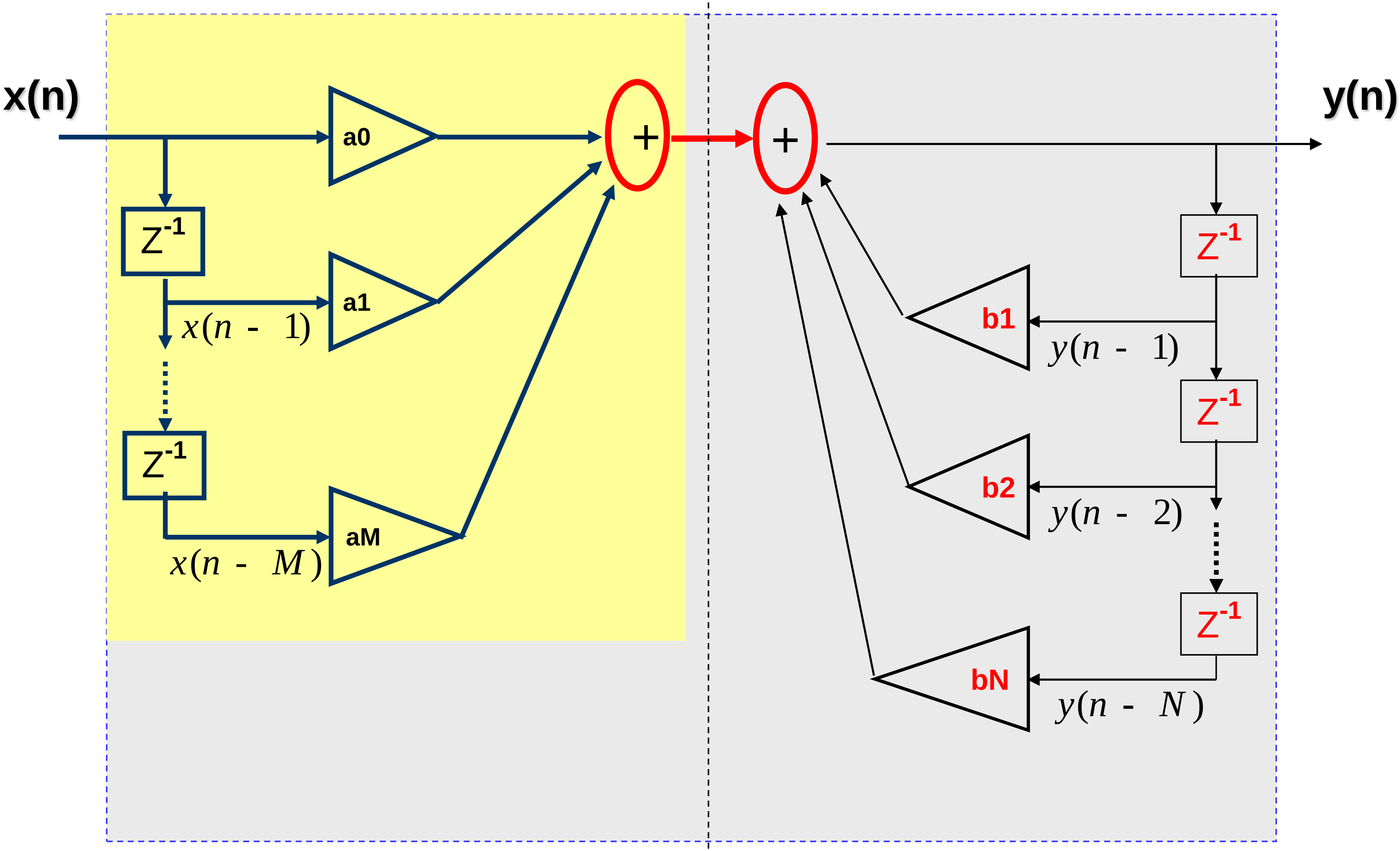
$$y(n)=\sum_{k=0}^{M}a(k)x(n-k)+\sum_{k=1}^{N}b(k)y(n-k)$$

IIR滤波器

直接I型实现

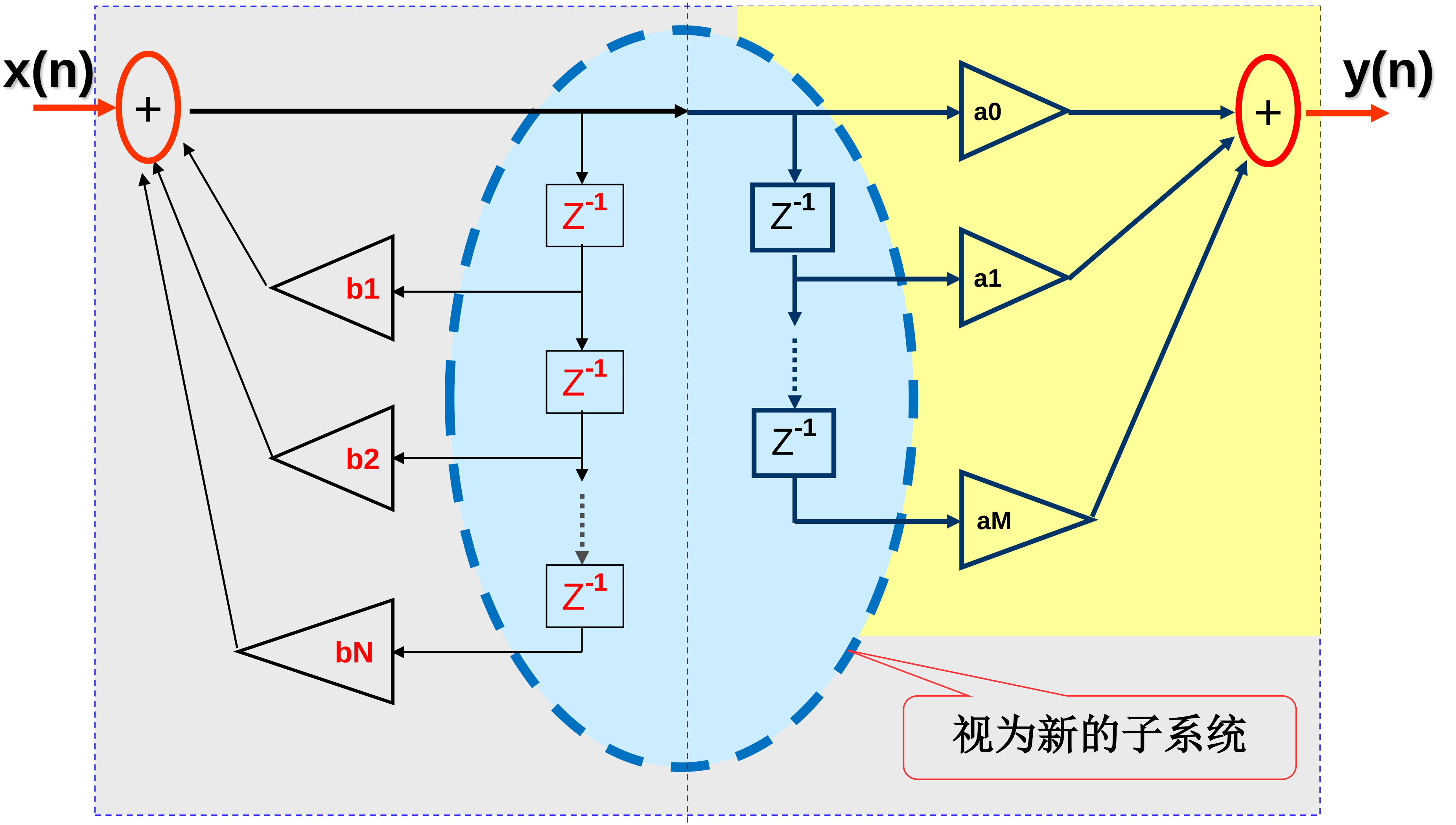


直接II型实现（STEP1）



直接II型实现（STEP2）

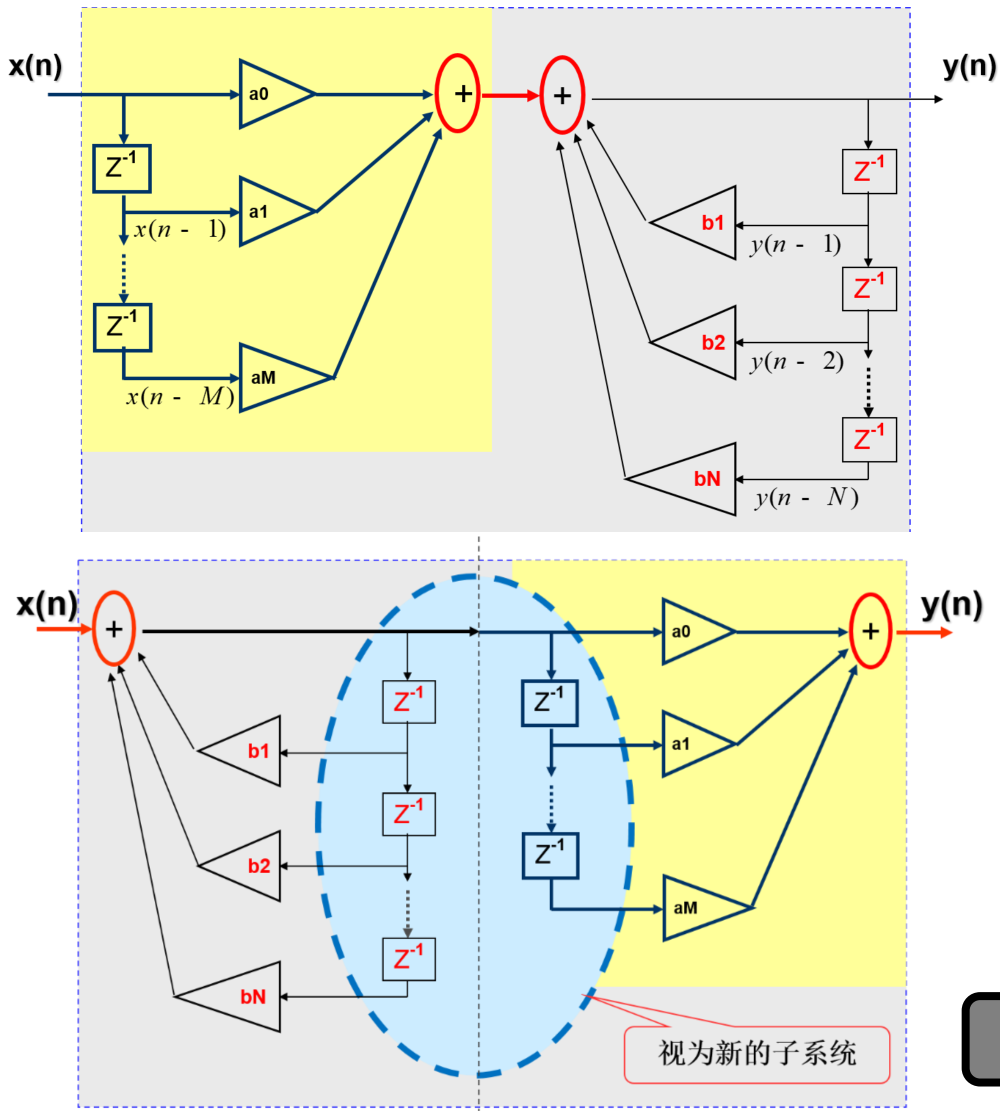
为什么交换前后系统是等效的？



主观题 1分

设置

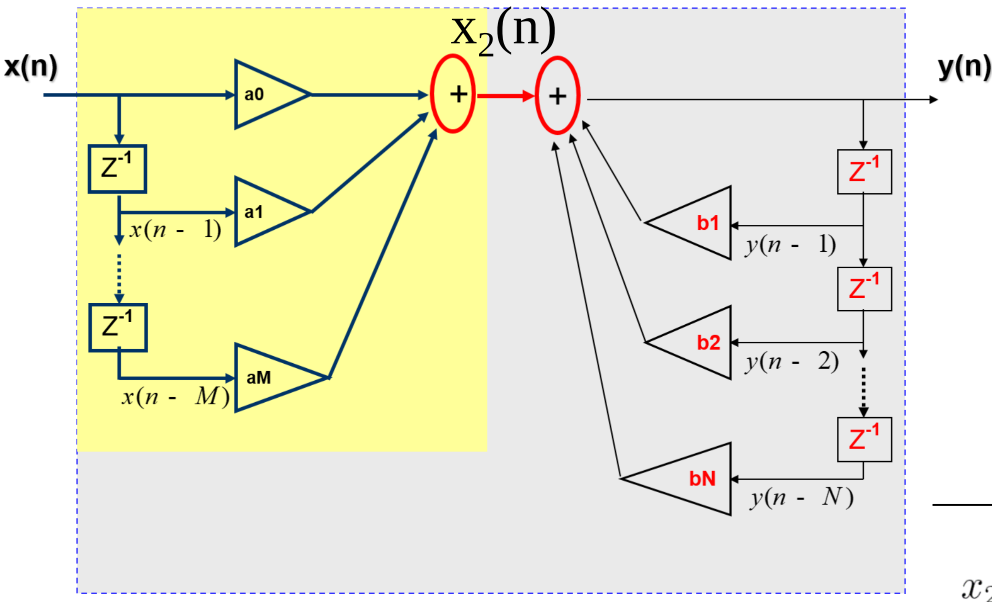
【课堂练习4】证明交换前后的两个系统是等效的



作答

课堂练习4参考答案

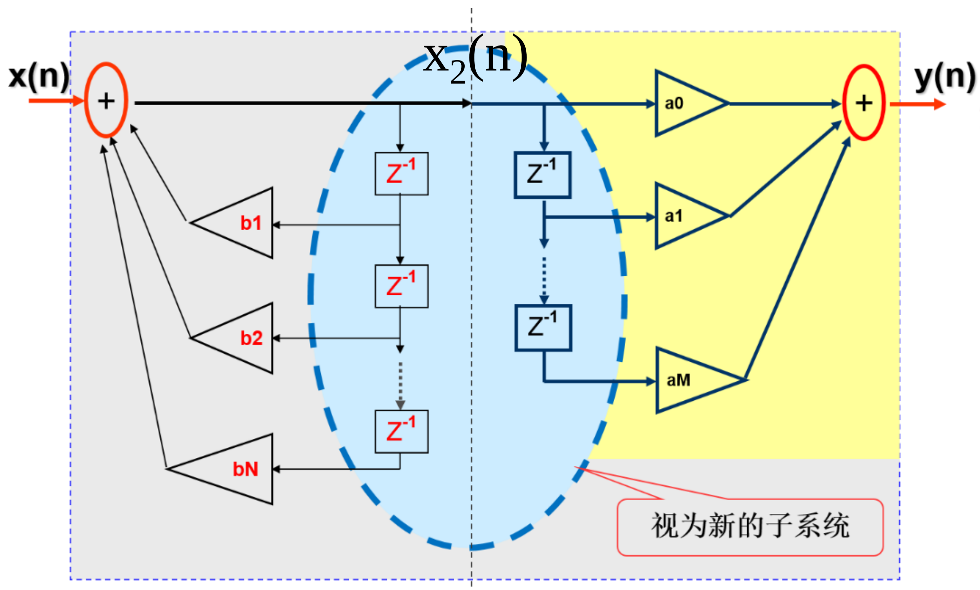
证明交换前后的两个系统是等效的



$$x_2(n)=\sum_{k=0}^{M}a_kx(n-k)$$

$$y(n)=x_2(n)+\sum_{i=1}^{N}b_iy(n-i)$$

$$=\sum_{k=0}^{M}a_kx(n-k)+\sum_{i=1}^{N}b_iy(n-i)$$



$$x_2(n)=x(n)+\sum_{i=1}^{N}b_ix_2(n-i)$$

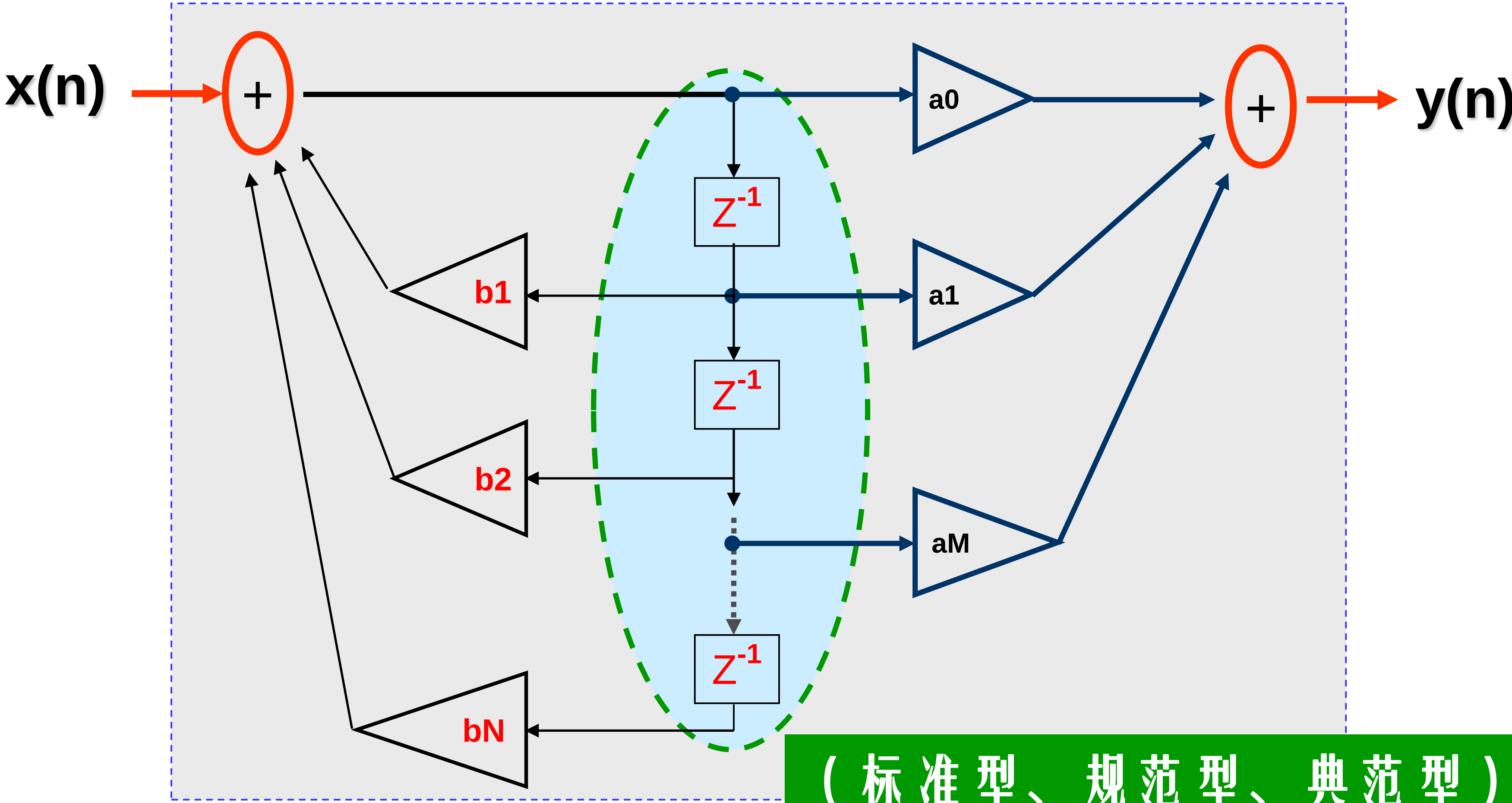
$$y(n)=\sum_{k=0}^{M}a_kx_2(n-k)$$

$$=\sum_{k=0}^{M}a_k[x(n-k)+\sum_{i=1}^{N}b_ix_2(n-i-k)]$$

$$=\sum_{k=0}^{M}a_kx(n-k)+\sum_{i=1}^{N}b_i[\sum_{k=0}^{M}a_kx_2(n-i-k)]$$

$$=\sum_{k=0}^{M}a_kx(n-k)+\sum_{i=1}^{N}b_iy(n-i)$$

直接II型实现（STEP3）



（标准型、规范型、典范型）

与直接I型相比，直接II型减少了对输入和输出状态的存储。

因需要两个加法，用DSP硬件实现时，可能引起算术溢出。尽管如此，因存储效率高，在滤波器实现广泛应用。

结束