Implementation of Digital Control algorithms on a computer

- It consists of two steps
- 1) Block diagram realization of the transfer function (obtained by the discretization of analog controller) or by the direct digital design that represents the control algorithm
- 2) software design based on block diagram realization

Different structures are possible for block diagram realizations of digital controllers

using delay elements, adders and multipliers.

Are equivalent from the input –output point of view assuming that calculations are done with infinite precision

With finite precision choice of realization is very important

Bad choice of realization may give a controller that is very sensitive to errors in the computation.

- Lets realize the controller
- $D(z) = \frac{U(z)}{E(z)} = \frac{\beta_0 z^n + \beta_1 z^{n-1} + \cdots\ldots \beta_{n-1} z + \beta_n}{z^n + \alpha_1 z^{n-1} + \cdots\ldots \alpha_{n-1} z + \alpha_n}$
- $\alpha_i$and β_j are real coefficients (some of them may be zero)

- It can be equivalently written as
- $D(z) = \frac{U(z)}{E(z)} = \frac{\beta_0 + \beta_1 z^{-1} + \cdots\ldots \beta_{n-1} z^{-(n-1)} + \beta_n z^{-n}}{1 + \alpha_1 z^{-1} + \cdots\ldots \alpha_{n-1} z^{-(n-1)} + \alpha_n z^{-n}}$
- Methods of realizing digital system can be divided into two classes
- Recursive
- Nonrecursive

- Functional relation in e(k) input and u(k) output for a recursive realization
- u(k)=f(u(k-1), u(k-2),……e(k),e(k-1),….)
- For above system

- u(k)=-$\alpha_1 u(k-1)$—.. $\alpha_{n-1} u(k-(n-1)-\alpha_0 u(k-n)+\beta_0 e(k)+\beta_1 e(k-1)+\cdots+\beta_n u(k-n)$
- IIR
- Nonrecursive realization
- u(k)=$\beta_0 e(k)+\beta_1 e(k-1)+\cdots+\beta_n u(k-N)$
- FIR

- Recursive Realization
- Direct Realization
- $D(z) = \frac{U(z)}{E(z)} = \frac{\beta_0+\beta_1 z^{-1}+\cdots\ldots\ \beta_{n-1}z^{-(n-1)}+\beta_n z^{-n}}{1+\alpha_1 z^{-1}+\cdots\ldots\ \alpha_{n-1}z^{-(n-1)}+\alpha_n z^{-n}} \frac{X(z)}{X(z)}$

- U(z)= $(\beta_0 + \beta_1 z^{-1} + \cdots \ldots\ \beta_{n-1}z^{-(n-1)} + \beta_n z^{-n})$X(z)
- E(z)=$(1 + \alpha_1 z^{-1} + \cdots \ldots\ \alpha_{n-1}z^{-(n-1)} + \alpha_n z^{-n})$X(z)
- Writing above equation in a cause and effect relation
- X(z)=E(z)-$(\alpha_1 z^{-1} + \cdots \ldots\ \alpha_{n-1}z^{-(n-1)} + \alpha_n z^{-n})$X(z)
- Block diagram portraying above 2 equations can be drawn
- To understand lets take n=3

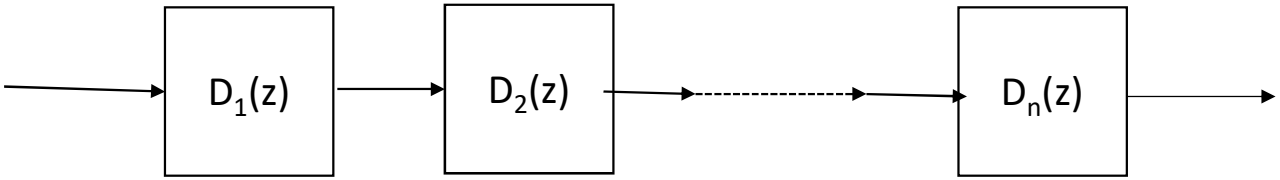
- X(z)=E(z)-$\alpha_1 z^{-1} X(z) - \alpha_2 z^{-2} X(z) - \alpha_3 z^{-3}$X(z)
- U(z)= $\beta_0 X(z) + \beta_1 z^{-1} X(z) + \beta_2 z^{-2}$X(z) + $\beta_3 z^{-3}$X(z)

- Sources of error that affect the accuracy of realization
- quantization of the input signal into a finite number of discrete levels
- accumulation of round off errors in the arithmetic operations in the digital systems
- Quantization of coefficients $\alpha_i$and β_j . Will be large for higher order transfer function. Small errors in coeff. Cause large error in the location of poles and zeros of the controller.

These three errors arise of practical limitations of the number of bits that represent various signal samples and coefficients.

Third error can be reduced by mathematically decomposing a higher order transfer function into a combination of lower order transfer function

- It will make less sensitive to coefficient inaccuracies
- Cascade realization
- Parallel Realization
- **<u>Cascade realization</u>**
- Implementing D(z) as a cascade connection of first order and second order tr.fn.
- $D(z)=D_1(z)\ D_2(z)\ D_3(z)\ldots..\ D_n(z)$



- In general D(z) can be decomposed as follows

- $D(z)=\prod_{i=1}^{p}\frac{1+b_i z^{-1}}{1+a_i z^{-1}}\prod_{j=p+1}^{n}\frac{1+e_j z^{-1}+f_j z^{-2}}{1+c_j z^{-1}+d_j z^{-2}}$

- Block diagram for
- $D_i(z)=\frac{1+b_i z^{-1}}{1+a_i z^{-1}}$
- $D_j(z)=\frac{1+e_j z^{-1}+f_j z^{-2}}{1+c_j z^{-1}+d_j z^{-2}}$

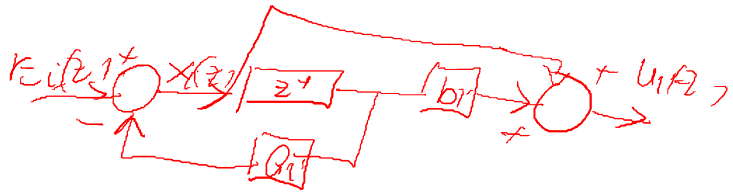
Realization of individual component

$$D_i(z) = \frac{1 + b_i z^{-1}}{1 + a_i z^{-1}} = \frac{U_i(z)}{F_i(z)}$$

$$U_i(z) = (1 + b_i z^{-1}) X_i(z)$$

$$F_i(z) = (1 + a_i z^{-1}) X_i(z)$$

$$X_i(z) = F_i(z) - a_i z^{-1} X_i(z)$$



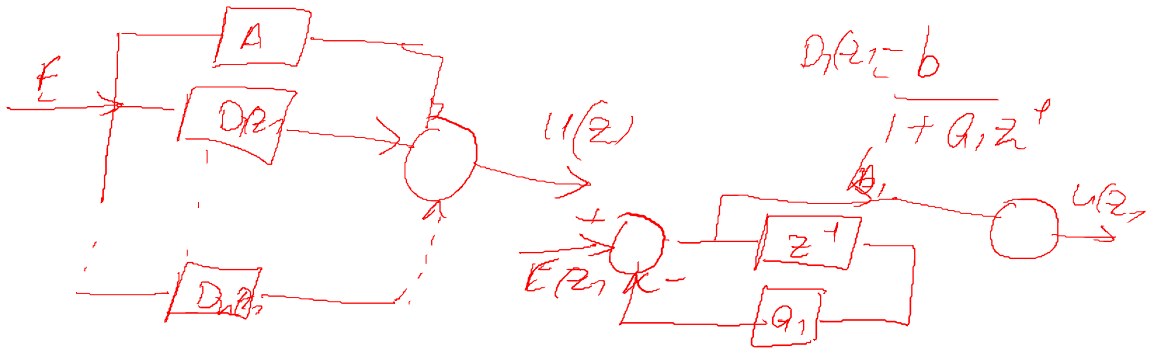
$$D_j(z) = \frac{1 + e_j z^{-1} + f_j z^{-2}}{1 + c_j z^{-1} + d_j z^{-2}}$$

2 delays
4 multipliers
4 adders

Parallel Decomposition

- Expand D(z) into partial fraction
- $D(z)=A+D_1(z)+D_2(z)+D_3(z)\ldots..+D_n(z)$

- $D(z)=D(z)=A+\sum_{i=1}^{q}\frac{b_i}{1+a_iz^{-1}}+\sum_{j=q+1}^{r}\frac{e_j+f_jz^{-1}}{1+c_jz^{-1}+d_jz^{-2}}$
- Type equation here.



$$D(z) = \frac{b}{1 + a_1 z^{-1}}$$

Non recursive realization

- In this commonly used are direct and cascade
- Parallel is not used since it requires more elements

$$\frac{N(z)}{D(z)} = \frac{\text{Polynomial}}{\text{Polynomial}}$$

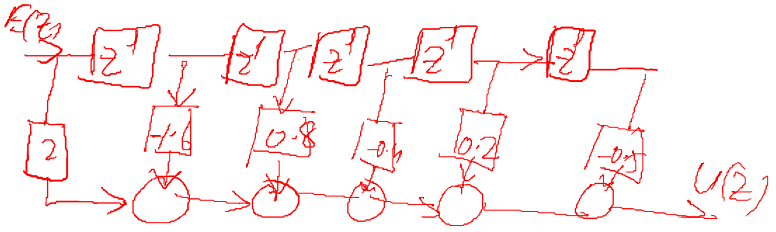
$$\frac{N(z)}{D(z)} = \frac{\text{Factored}}{\text{Factored}}$$

$$\frac{N(z)}{D(z)} = \frac{\text{Polynomial}}{\text{factored}}$$

$$\frac{U(z)}{E(z)} = \frac{2 - 0.6z^{-1}}{1 + 0.5z^{-1}}$$

$$1 + 0.5z^{-1} \overline{)\, 2 - 0.6z^{-1}} \quad 2 - 1.6z^{-1} + 0.8z^{-2} - 0.4z^{-3} + 0.2z^{-4} - 0.1z^{-5} \ldots$$

$$U(z) = 2E(z) - 1.6z^{-1}E(z) + 0.8z^{-2}E(z) - 0.4z^{-3}E(z) + 0.2z^{-4}E(z) - 0.1z^{-5}E(z)$$



- Advantage lack of feedback accumulation of errors in past outputs is avoided in the processing of the signal

large no. of delays → Stability