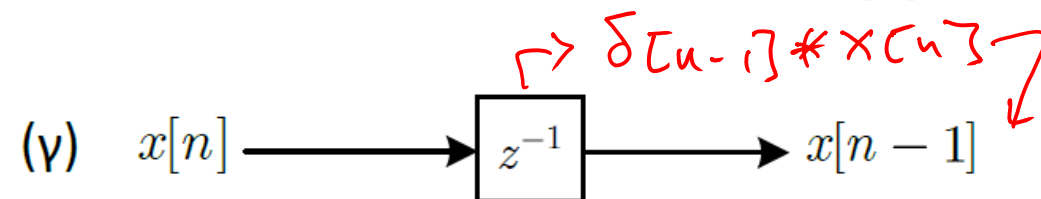
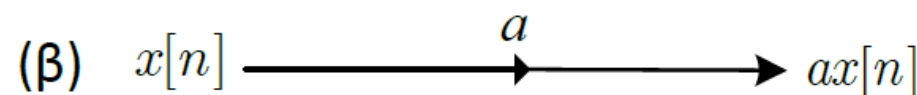
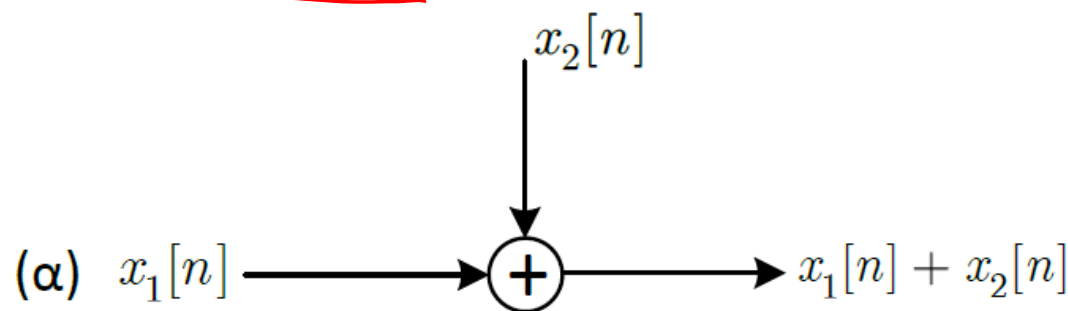


Ψηφιακή Επεξεργασία Σήματος

ΔΙΑΛΕΞΗ 18^Η

- Υλοποίηση Συστημάτων Διακριτού Χρόνου

- Υλοποίηση συστημάτων διακριτού χρόνου
- Για να υλοποιήσουμε ένα σύστημα πρέπει να μετατρέψουμε τη συνάρτηση μεταφοράς ή την εξίσωση διαφορών σε μια δομή που πραγματοποιείται από την υπάρχουσα τεχνολογία
- Βασικοί δομικοί λίθοι:
 - Πρόσθεση
 - Πολλαπλασιασμός
 - Καθυστέρηση (αποθήκευση στη μνήμη)



- Υλοποίηση συστημάτων διακριτού χρόνου

- Έστω η εξίσωση διαφορών $y[n] - a_1 y[n-1] - a_2 y[n-2] = b_0 x[n]$

$$\underbrace{y[n] - a_1 y[n-1] - a_2 y[n-2]}_{\text{Z}} = b_0 x[n]$$

- Έχουμε πολλές επιλογές για την υλοποίησή της

$$Y(z) - a_1 z^{-1} Y(z) - a_2 z^{-2} Y(z)$$

$$= b_0 X(z) \Rightarrow$$

$$Y(z) (1 - a_1 z^{-1} - a_2 z^{-2}) =$$

$$X(z) b_0$$

- Κόστος:

- Πλήθος πράξεων

- Πλήθος θέσεων μνήμης

- Ευρωστία σε πεπερασμένη αριθμητική ακρίβεια

- Η συνάρτηση μεταφοράς δίνεται ως

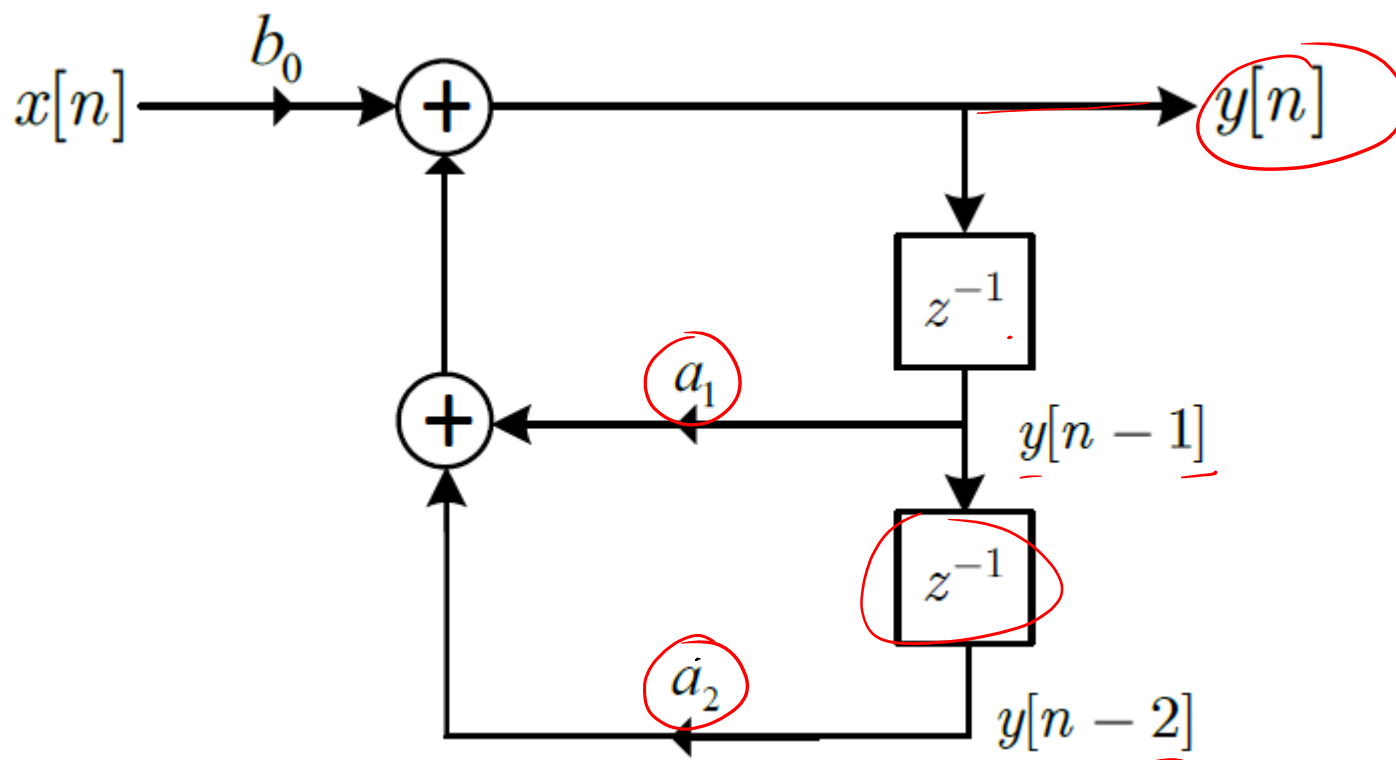
$$\frac{Y(z)}{X(z)} = H(z) = \frac{b_0}{1 - a_1 z^{-1} - a_2 z^{-2}} = \frac{b_0}{1 - (a_1 z^{-1} + a_2 z^{-2})}$$

- Με βάση την εξίσωση διαφορών, μπορούμε κατασκευάσουμε μια δομή που την υλοποιεί

- Υλοποίηση συστημάτων διακριτού χρόνου

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + b_0 x[n]$$

$$H(z) = \frac{b_0}{1 - a_1 z^{-1} - a_2 z^{-2}} = \frac{b_0}{1 - (a_1 z^{-1} + a_2 z^{-2})}$$



- Υλοποίηση συστημάτων διακριτού χρόνου
- Για μεγαλύτερης τάξης εξίσωσης διαφορών

$$y[n] - \sum_{k=1}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k]$$

$$\Rightarrow y[n] = \sum f(y) + \sum g(x)$$

με συνάρτηση μεταφοράς

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}}$$

- Ξαναγράφουμε την εξίσωση διαφορών

$$y[n] = \sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^M b_k x[n-k]$$

$u[n]$

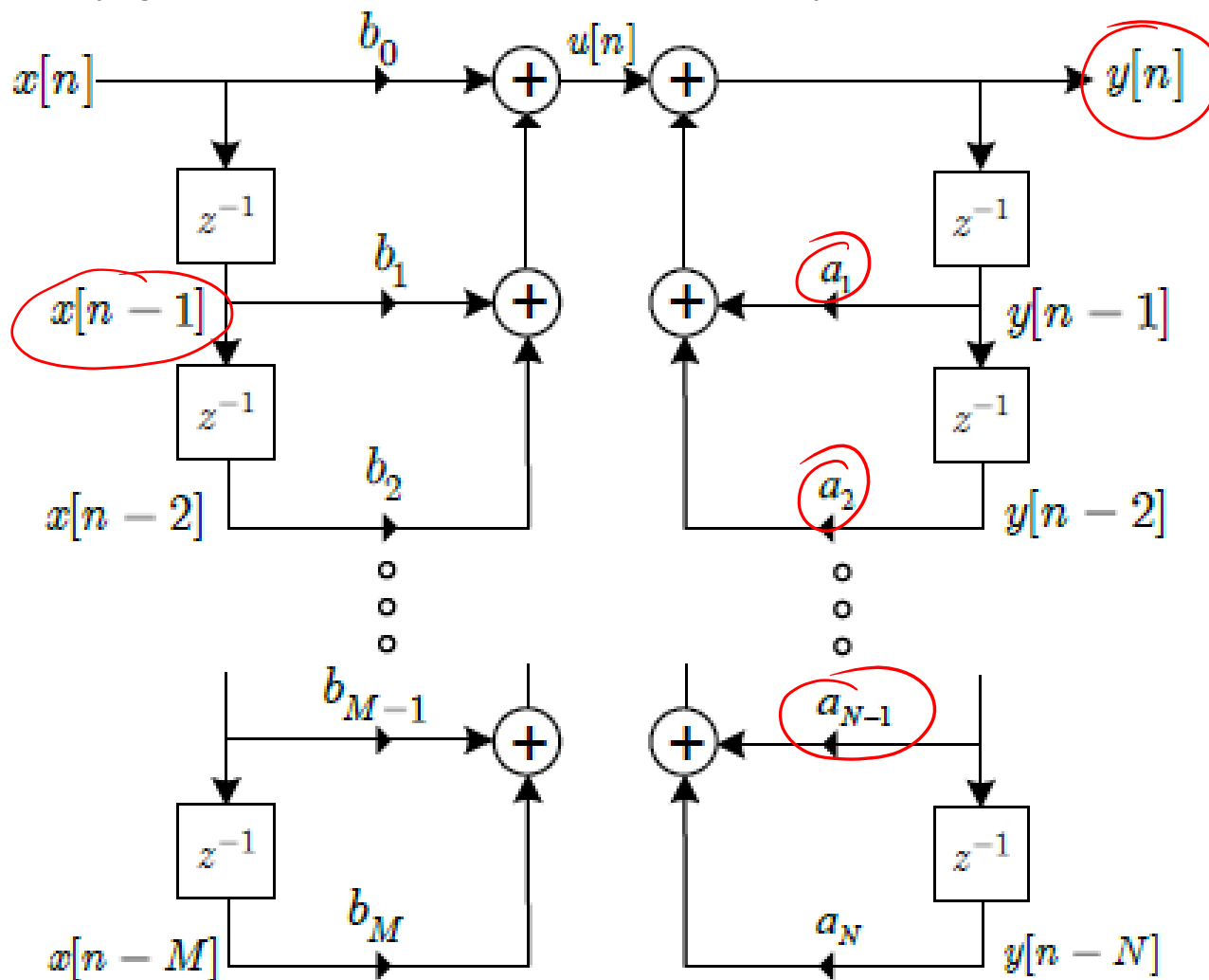
- Δυο υπο-εξισώσεις:

$$u[n] = \sum_{k=0}^M b_k x[n-k]$$

$$y[n] = \sum_{k=1}^N a_k y[n-k] + u[n]$$

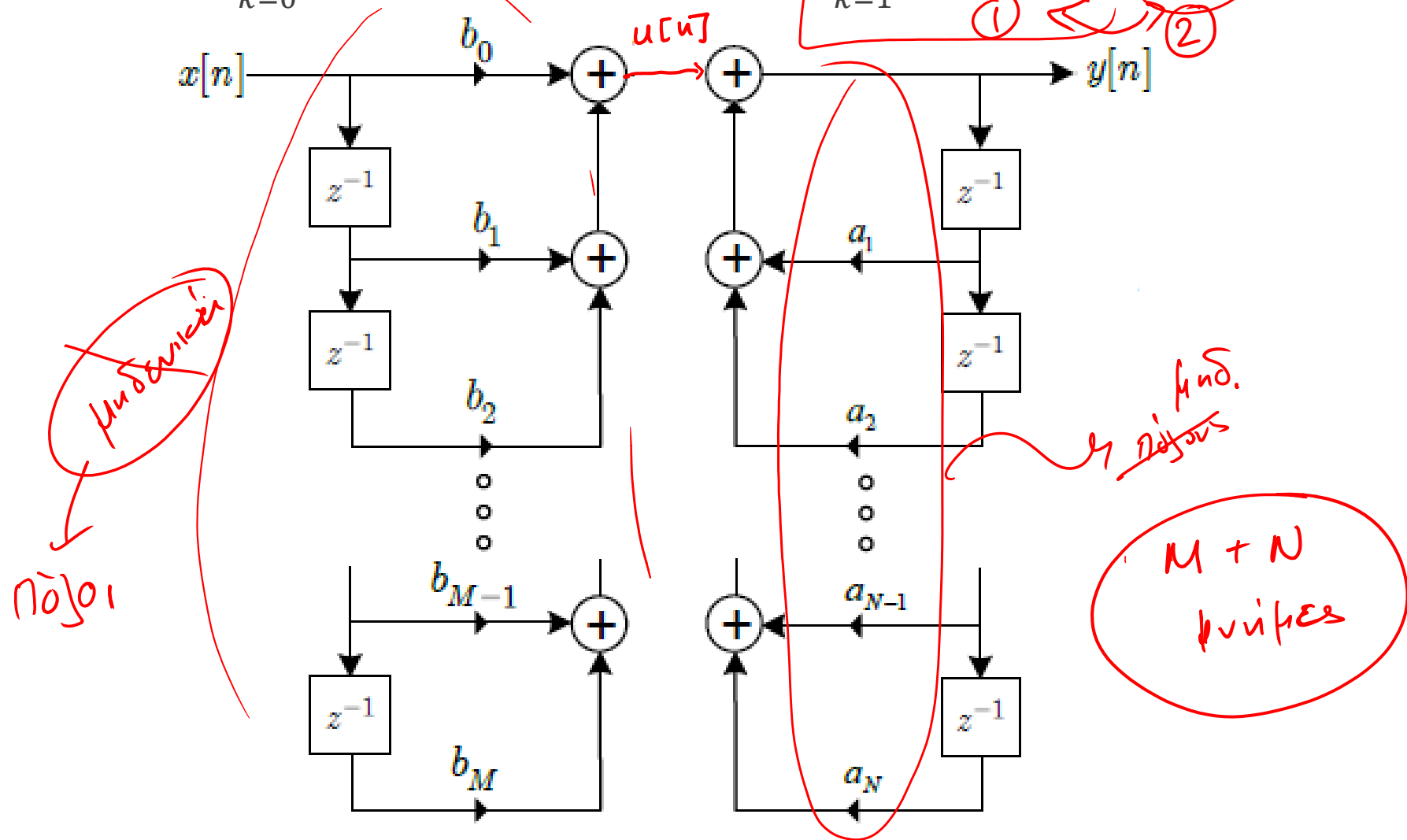
- Υλοποίηση συστημάτων διακριτού χρόνου

$$u[n] = \sum_{k=0}^M \underbrace{b_k}_{\text{μυθενικά}} x[n-k] \quad , \quad y[n] = \sum_{k=1}^N \underbrace{a_k}_{\text{πόσες}} y[n-k] + u[n]$$



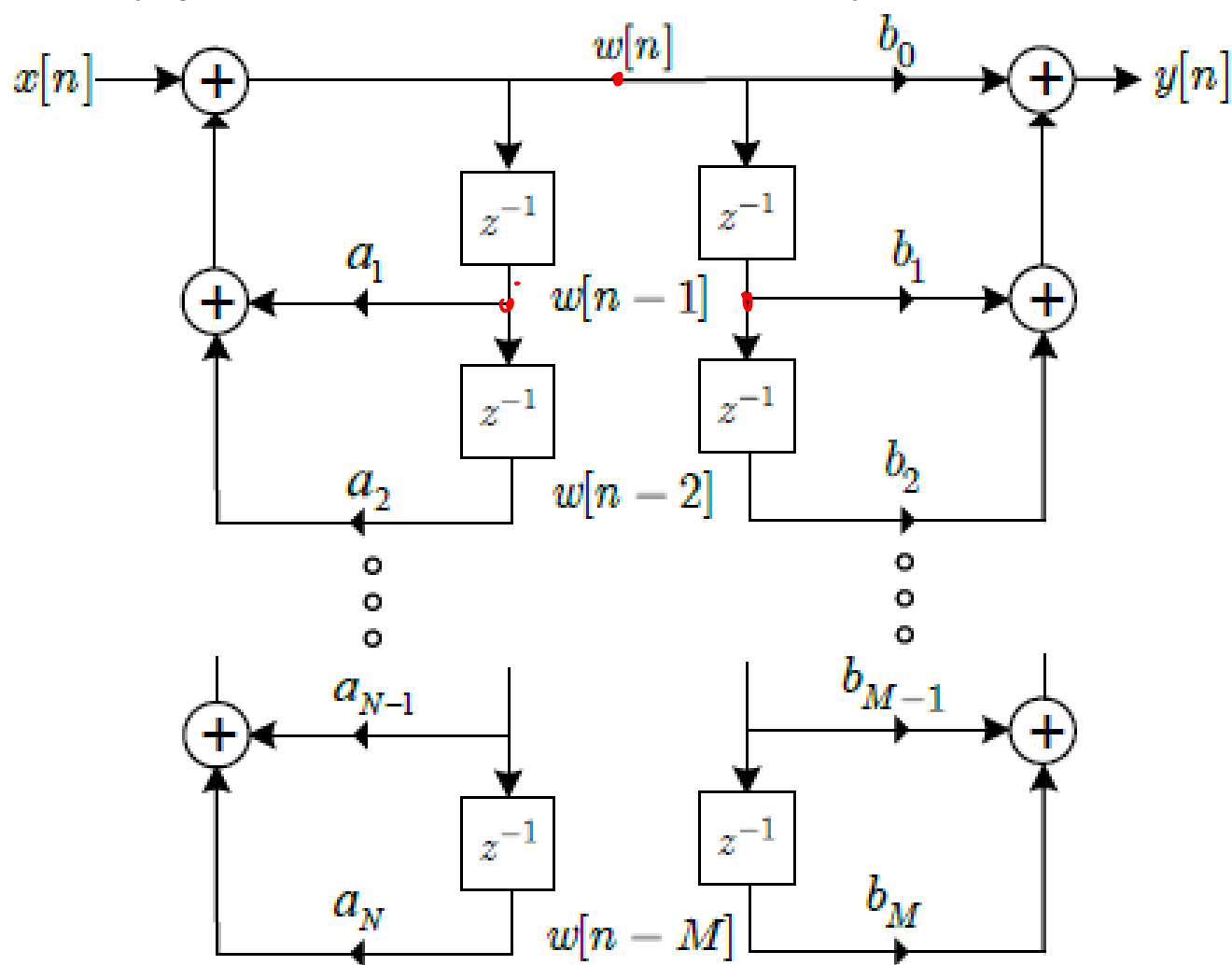
• Υλοποίηση συστημάτων διακριτού χρόνου

$$u[n] = \sum_{k=0}^M b_k x[n-k] \quad , \quad y[n] = \sum_{k=1}^N a_k y[n-k] + u[n]$$

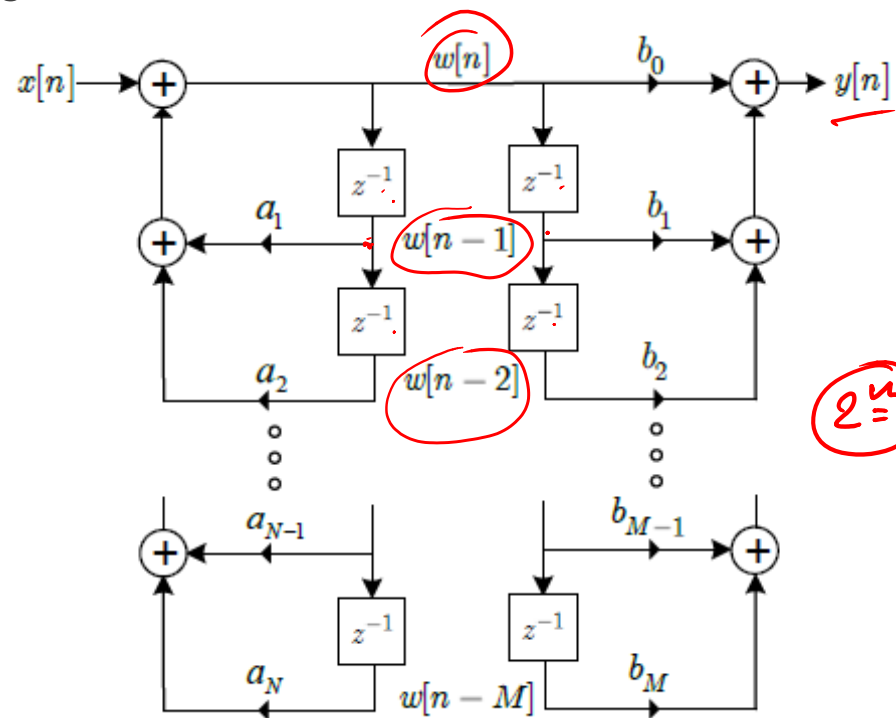
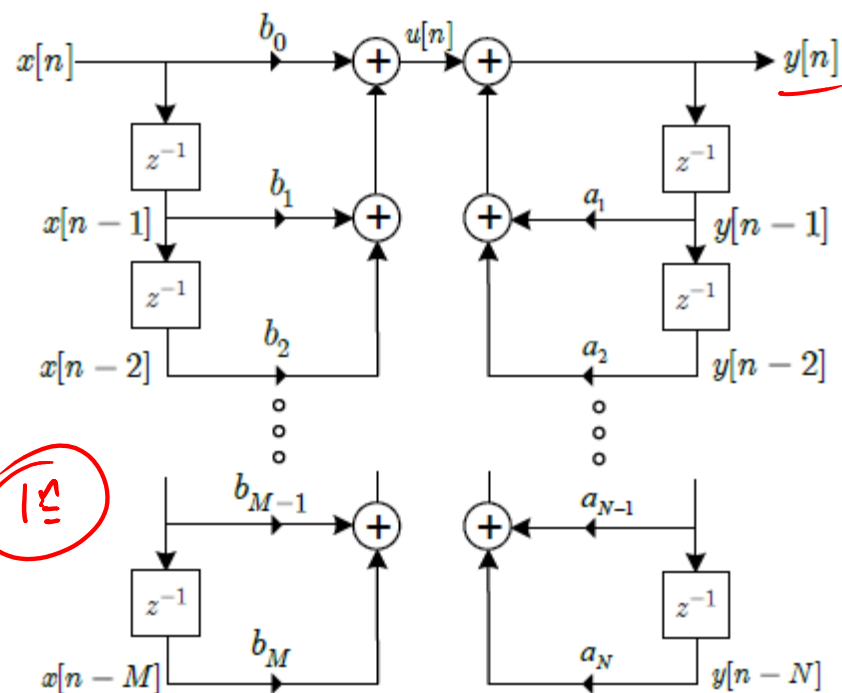


- Υλοποίηση συστημάτων διακριτού χρόνου

$$u[n] = \sum_{k=0}^M b_k x[n-k] \quad , \quad y[n] = \sum_{k=1}^N a_k y[n-k] + u[n]$$



• Υλοποίηση συστημάτων διακριτού χρόνου



$1 \pm$

$2 \pm$

• Βασικές διαφορές των δυο υλοποιήσεων

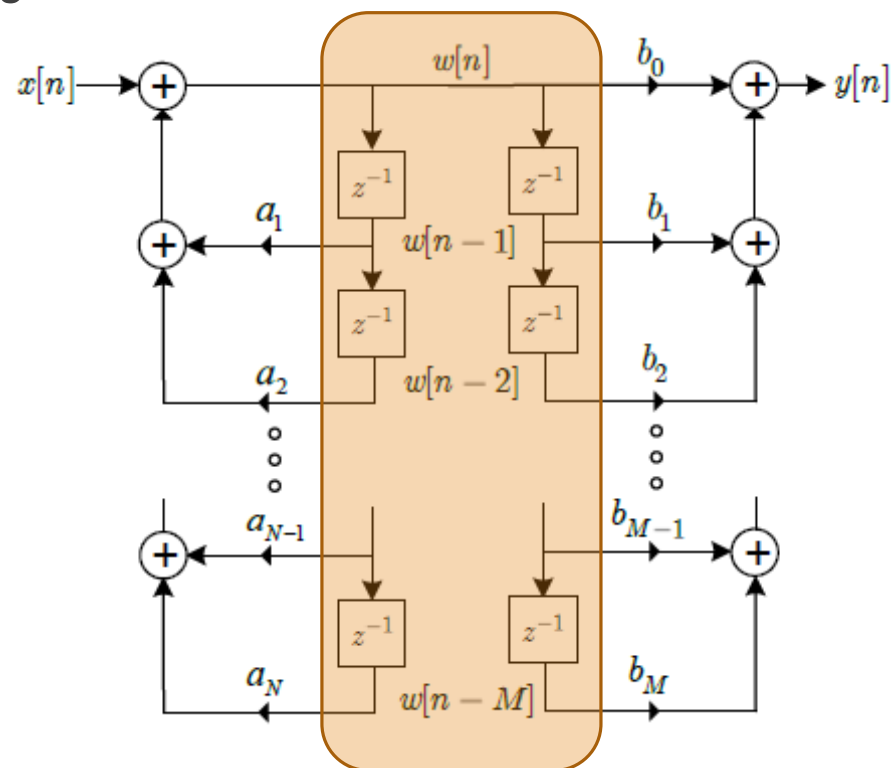
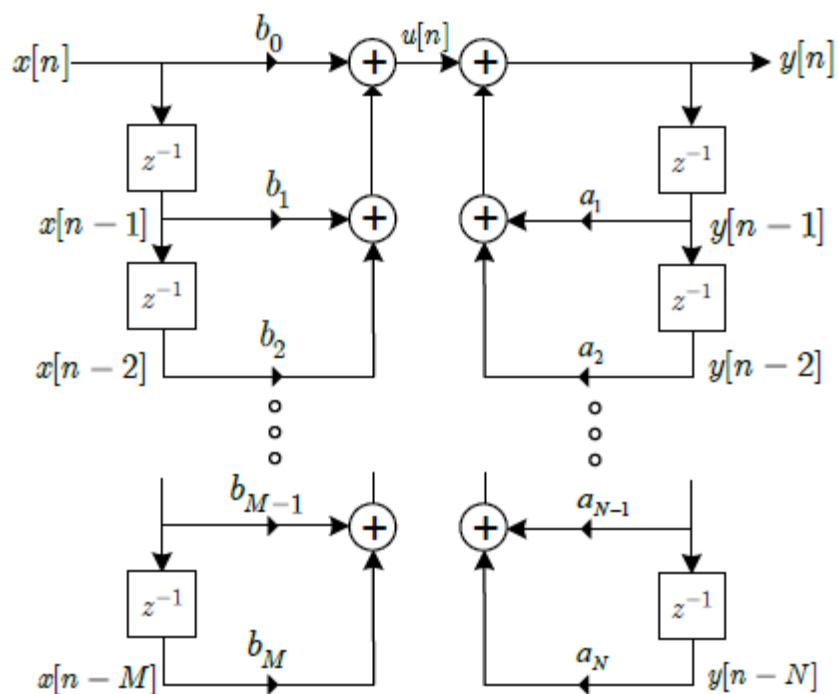
- Στην πρώτη, υλοποιούνται πρώτα τα μηδενικά και μετά οι πόλοι
- Στη δεύτερη, υλοποιούνται πρώτα οι πόλοι και μετά τα μηδενικά

$$y[n] = \sum f(\nu) + \sum f(x)$$

• Θεωρητικά, οι δυο υλοποιήσεις είναι ισοδύναμες

- Πρακτικά, μπορεί να υπάρχουν σημαντικές διαφορές! (λόγω υλοποίησης σε πεπερασμένη ακρίβεια)

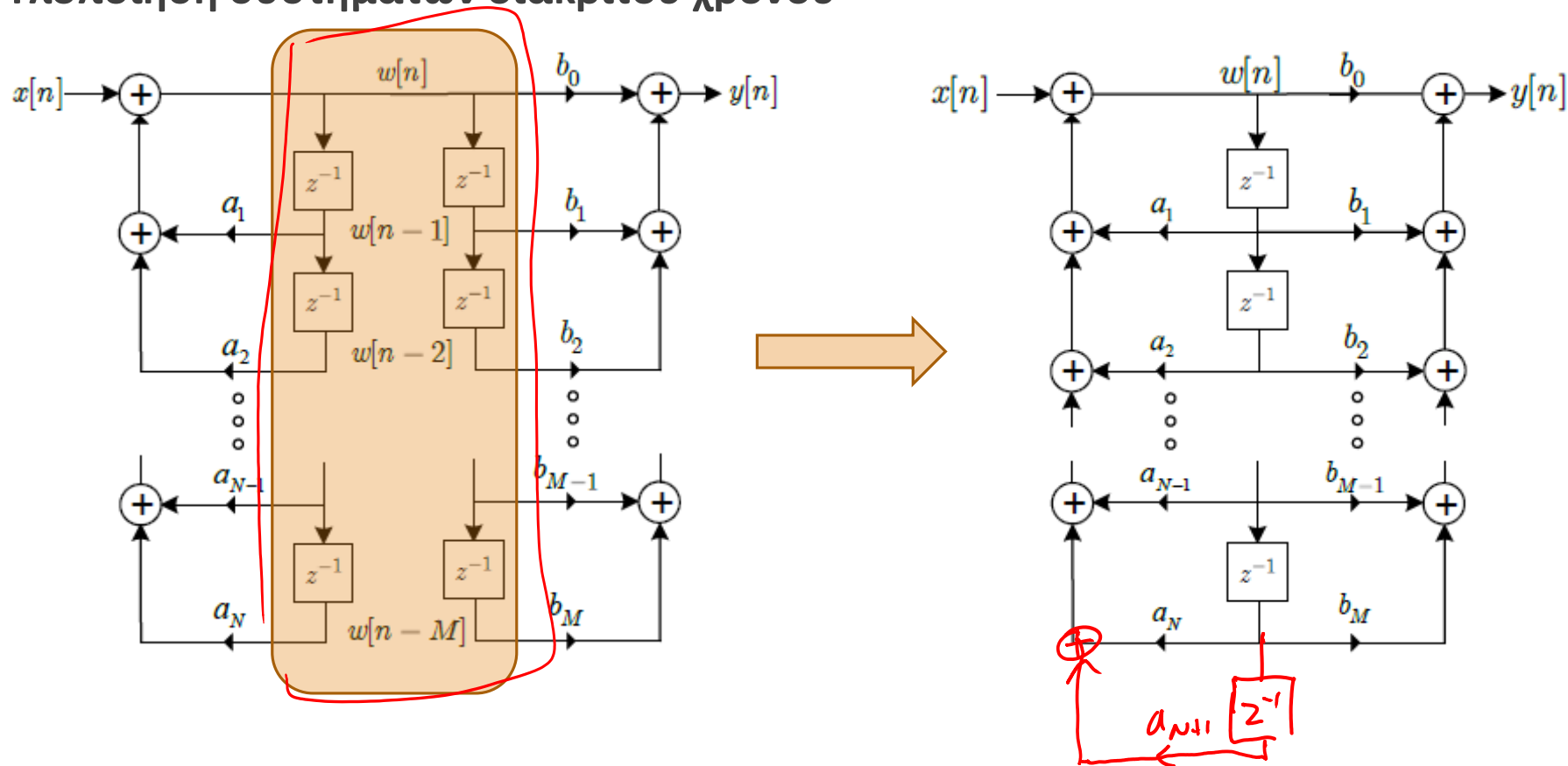
• Υλοποίηση συστημάτων διακριτού χρόνου



• Παρατήρηση:

- Οι μεταβλητές $w[n - k]$ αποθηκεύονται δυο φορές στη δεύτερη υλοποίηση!
- Μπορούμε να «γλιτώσουμε» τις μισές θέσεις μνήμης!
- Μπορούμε να μοιράσουμε τις ίδιες θέσεις μνήμης και στους δυο κλάδους

- Υλοποίηση συστημάτων διακριτού χρόνου



- Η νέα υλοποίηση, που χρησιμοποιεί τον ελάχιστο αριθμό στοιχείων καθυστέρησης (μνήμης) αναφέρεται ως **κανονική μορφή** (canonical form) ή **Direct Form II** $\rightarrow \max(M, N)$
- Η μη κανονική μορφή στο αριστερό σχήμα ονομάζεται **Direct Form I** $\rightarrow M+N$

• Υλοποίηση συστημάτων διακριτού χρόνου

• Παράδειγμα:

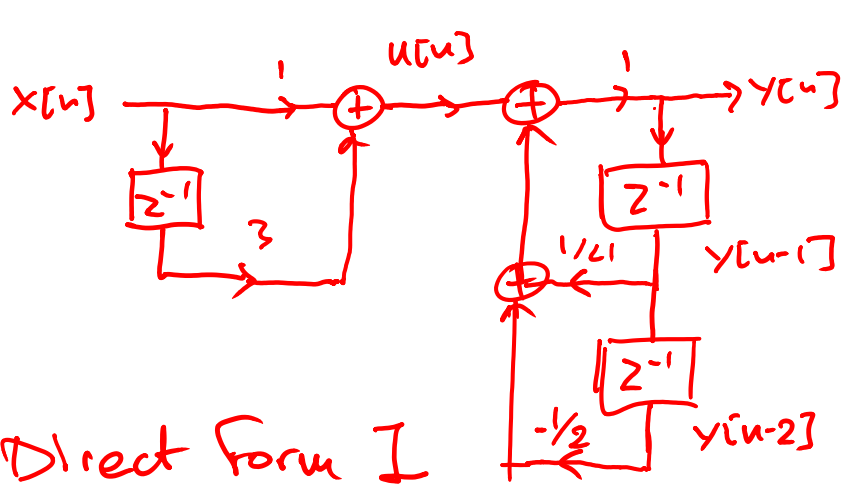
$$z^{-1} \rightarrow y[n] - \frac{1}{4} y[n-1] + \frac{1}{2} y[n-2] = x[n] + 3x[n-1] \Rightarrow$$

$$\Rightarrow y[n] = \frac{1}{4} y[n-1] - \frac{1}{2} y[n-2] + x[n] + 3x[n-1]$$

○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 3z^{-1}}{1 - \frac{1}{4}z^{-1} + \frac{1}{2}z^{-2}} = \frac{Y(z)}{X(z)} \Rightarrow Y(z) - \frac{1}{4}z^{-1}Y(z) + \frac{1}{2}z^{-2}Y(z) = X(z) + 3z^{-1}X(z)$$

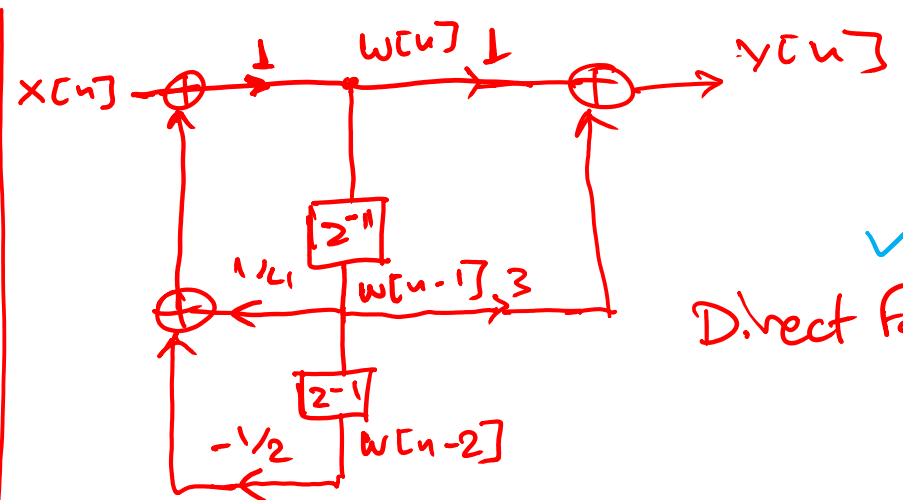
Σχεδιάστε τις υλοποιήσεις Direct Form I II



Direct Form I

3 μνήμες

3 αθροιστές
3 πολλαπλασιαστές
3 καθυστερητές



Direct Form II ✓

3 αθροιστές
3 πολλαπλασιαστές
2 μνήμες

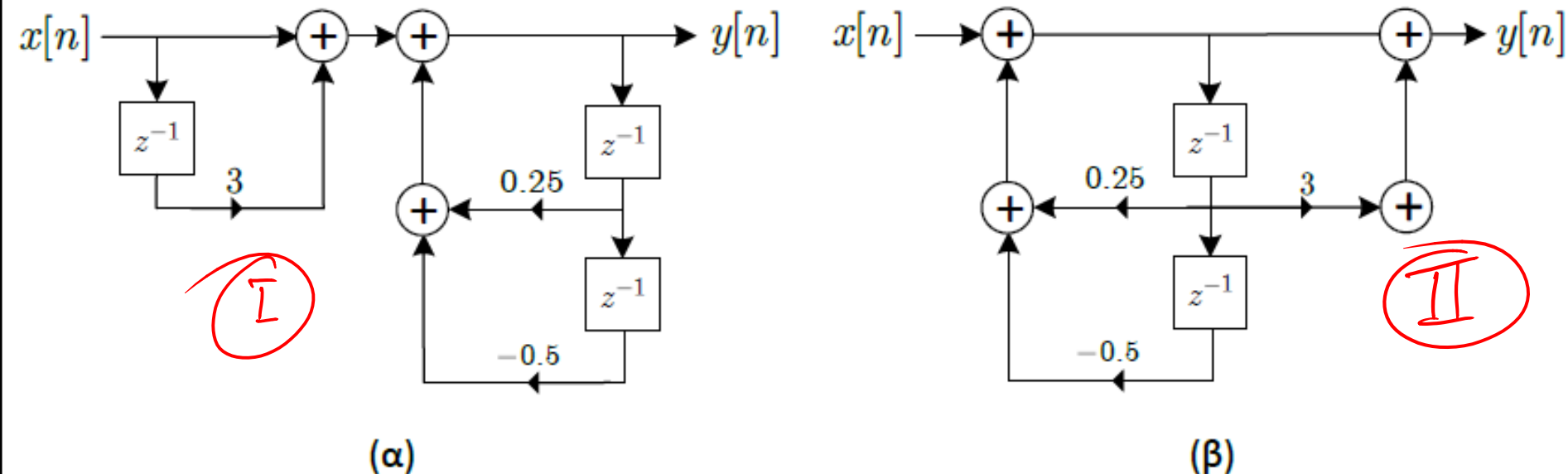
- Υλοποίηση συστημάτων διακριτού χρόνου

- Παράδειγμα:

○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 3z^{-1}}{1 - \frac{1}{4}z^{-1} + \frac{1}{2}z^{-2}}$$

Σχεδιάστε τις υλοποιήσεις Direct Form I, II



- **Υλοποίηση συστημάτων διακριτού χρόνου**

- Γνωρίζουμε ότι τα συστήματα χωρίζονται σε δυο βασικές κατηγορίες: FIR, IIR

- Ας δούμε πόσο διαφέρουν οι υλοποιήσεις ανάλογα με την κατηγορία

- Ξεκινάμε με τα IIR

- Θα δούμε ότι υπάρχουν πολλές διαφορετικές υλοποιήσεις ενός IIR συστήματος

- Πώς επιλέγουμε την κατάλληλη;

- ✓ Πλήθος πολλαπλασιασμών → χρονοβόρα πράξη

- ✓ Πλήθος καθυστερήσεων → κόστος σε μνήμη

- ✓ Εμβαδό, απλότητα, και αρθρωτή υλοποίηση → σημασία σε VLSI υλοποιήσεις

- ✓ Καταμερισμός αλγορίθμου, επικοινωνία επεξεργαστών → πολυεπεξεργαστικό περιβάλλον

- ✓ Ευρωστία σε πεπερασμένη ακρίβεια → προτίμηση από πιο οικονομικές υλοποιήσεις

- Υλοποίηση IIR συστημάτων διακριτού χρόνου
- Μορφή σε σειρά (cascade form)
- Συνάρτηση μεταφοράς

$$H(z) = A \frac{\prod_{k=1}^{M_1} (1 - e_k z^{-1}) \prod_{k=1}^{M_2} (1 - g_k z^{-1})(1 - g_k^* z^{-1})}{\prod_{k=1}^{N_1} (1 - c_k z^{-1}) \prod_{k=1}^{N_2} (1 - d_k z^{-1})(1 - d_k^* z^{-1})}$$

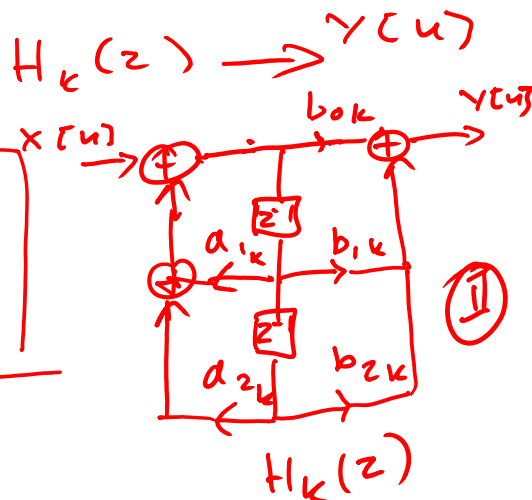
με $N = N_1 + 2N_2$, $M = M_1 + 2M_2$

$$x[n] \rightarrow H_1(z) \rightarrow H_2(z) \rightarrow \dots \rightarrow H_k(z) \rightarrow y[n]$$

- Εναλλακτικά

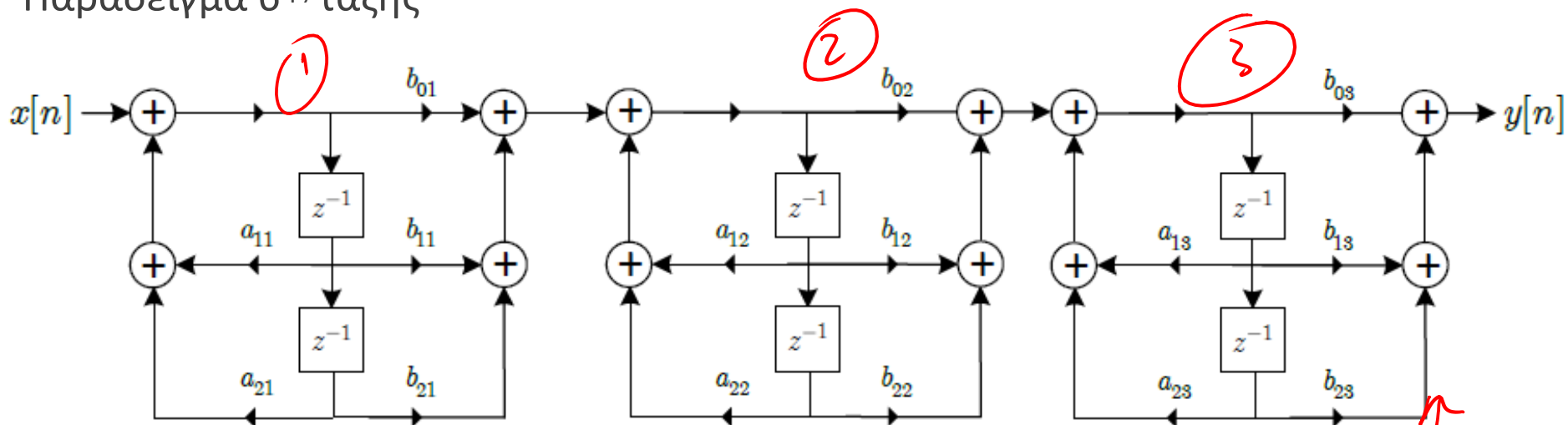
$$H(z) = \prod_{k=1}^{N_s} \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}}$$

$H_k(z)$



με $N_s = \left\lfloor \frac{N+1}{2} \right\rfloor$

- Υλοποίηση IIR συστημάτων διακριτού χρόνου
- Μορφή σε σειρά (cascade form)
- Παράδειγμα 6ης τάξης



- Οργανωμένο σε υποσυστήματα δευτέρας τάξης Direct Form II

$$H(z) = \prod_{k=1}^3 \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}}$$

- Υλοποίηση IIR συστημάτων διακριτού χρόνου

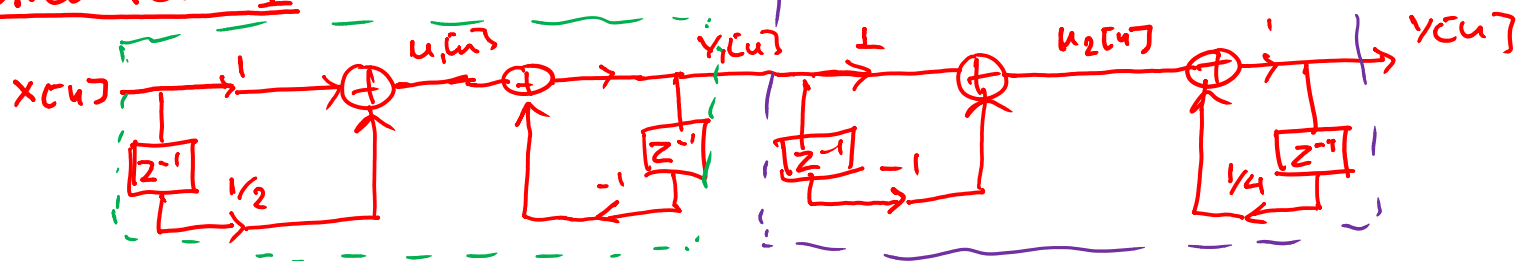
- Παράδειγμα:

○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 - 0.5z^{-1} - 0.5z^{-2}}{1 + 0.75z^{-1} - 0.25z^{-2}} = \frac{(1 + \frac{1}{2}z^{-1})(1 - z^{-1})}{(1 + z^{-1})(1 - \frac{1}{4}z^{-1})}$$

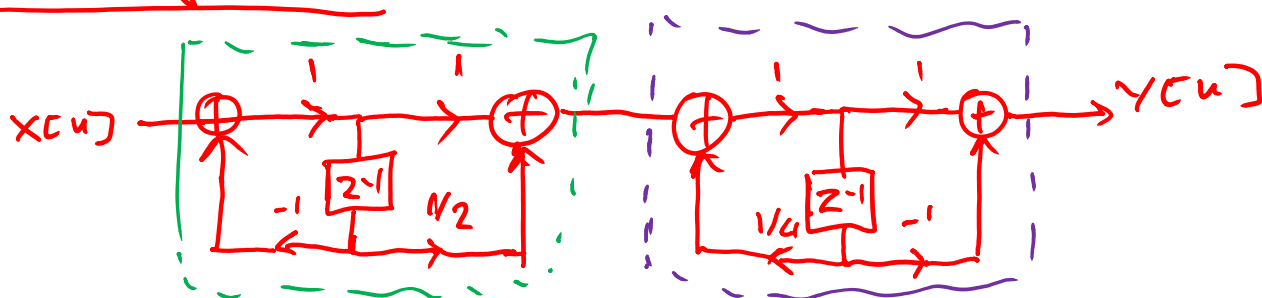
Σχεδιάστε τις υλοποιήσεις σε σειρά με χρήση Direct Form I, II υποσυστημάτων και αναφέρετε το κέρδος σε πολλαπλασιασμούς ή/και θέσεις μνήμης

Direct Form I



4 μνήμες

Direct form II



2 μνήμες ✓

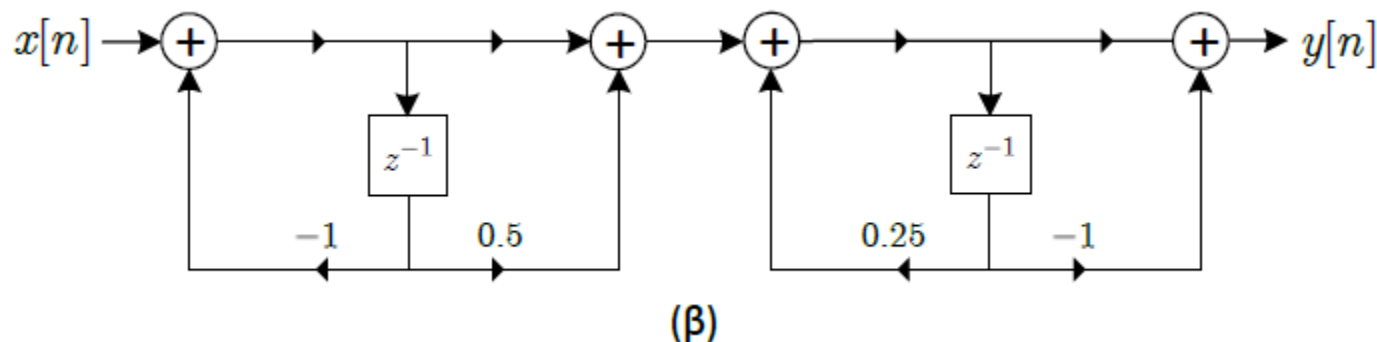
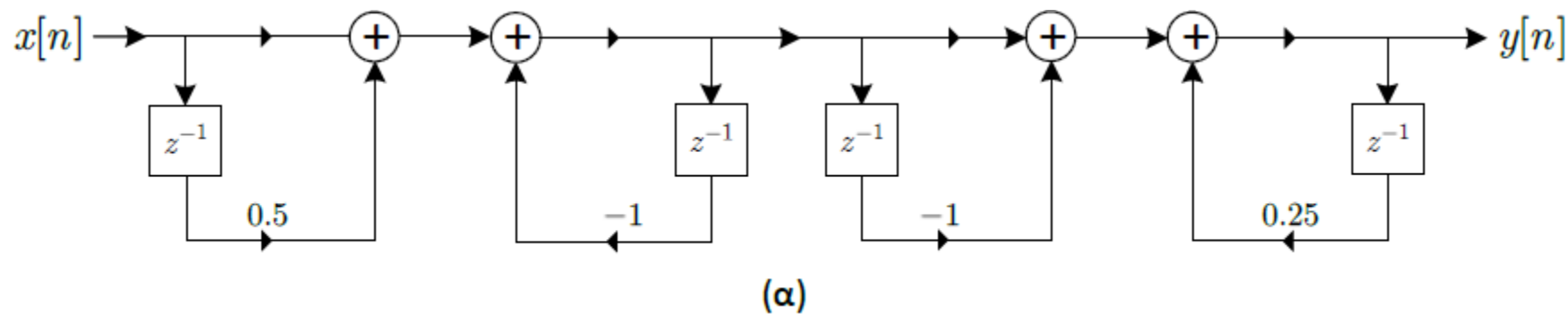
- Υλοποίηση IIR συστημάτων διακριτού χρόνου

- Παράδειγμα:

○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 - 0.5z^{-1} - 0.5z^{-2}}{1 + 0.75z^{-1} - 0.25z^{-2}}$$

Σχεδιάστε τις υλοποιήσεις σε σειρά με χρήση Direct Form I, II υποσυστημάτων



- **Υλοποίηση IIR συστημάτων διακριτού χρόνου**
- Παράλληλη Μορφή (parallel form)
 - Προκύπτει από το ανάπτυγμα σε μερικά κλάσματα
- Συνάρτηση μεταφοράς

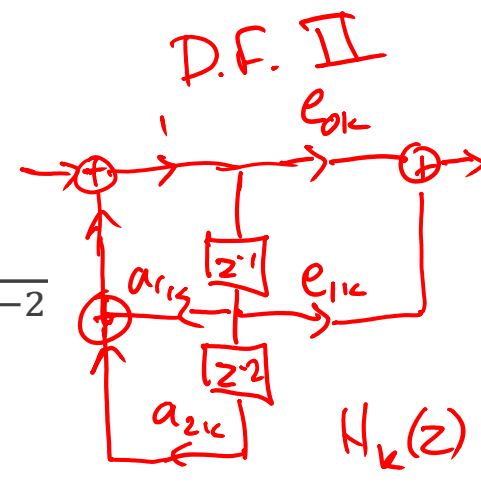
$$H(z) = \underbrace{\sum_{k=0}^{N_p} C_k z^{-k}}_{N_p = M-N} + \sum_{k=1}^{N_1} \frac{A_k}{1 - c_k z^{-1}} + \sum_{k=1}^{N_2} \frac{B_k (1 - e_k z^{-1})}{(1 - d_k z^{-1})(1 - d_k^* z^{-1})}$$

$$\text{με } N = N_1 + 2N_2$$

- Εναλλακτικά

$$H(z) = \sum_{k=0}^{N_p} C_k z^{-k} + \sum_{k=1}^{N_s} \frac{e_{0k} + e_{1k} z^{-1}}{\underbrace{1 - a_{1k} z^{-1} - a_{2k} z^{-2}}_{H_k(z)}}$$

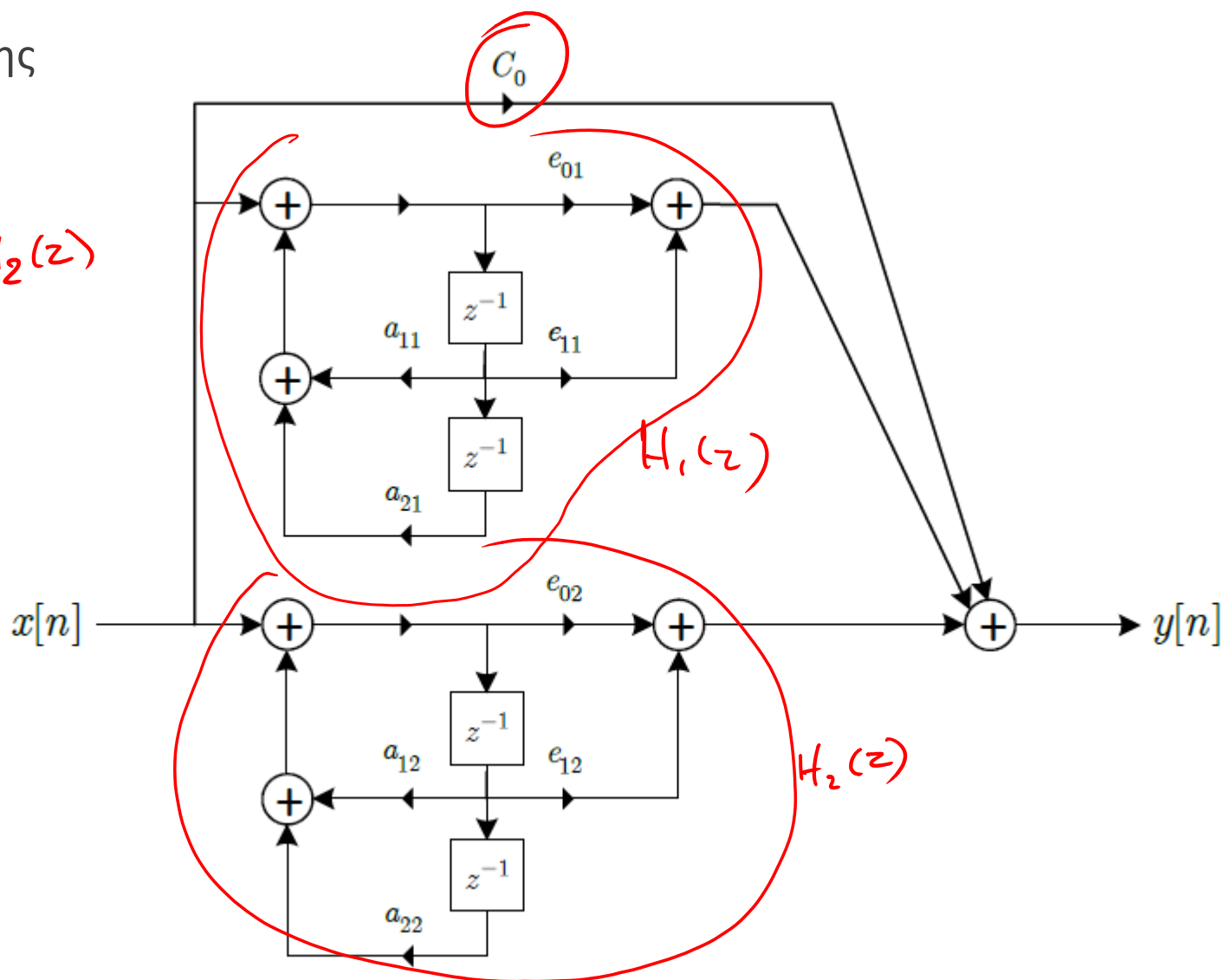
$$\text{με } N_s = \left\lfloor \frac{N+1}{2} \right\rfloor$$



- Υλοποίηση IIR συστημάτων διακριτού χρόνου
- Παράλληλη Μορφή (parallel form)
- Παράδειγμα 4^{ης} τάξης

$$M = N = 4$$

$$H(z) = C_0 + H_1(z) + H_2(z)$$



- Υλοποίηση IIR συστημάτων διακριτού χρόνου

$$N = M = 2$$

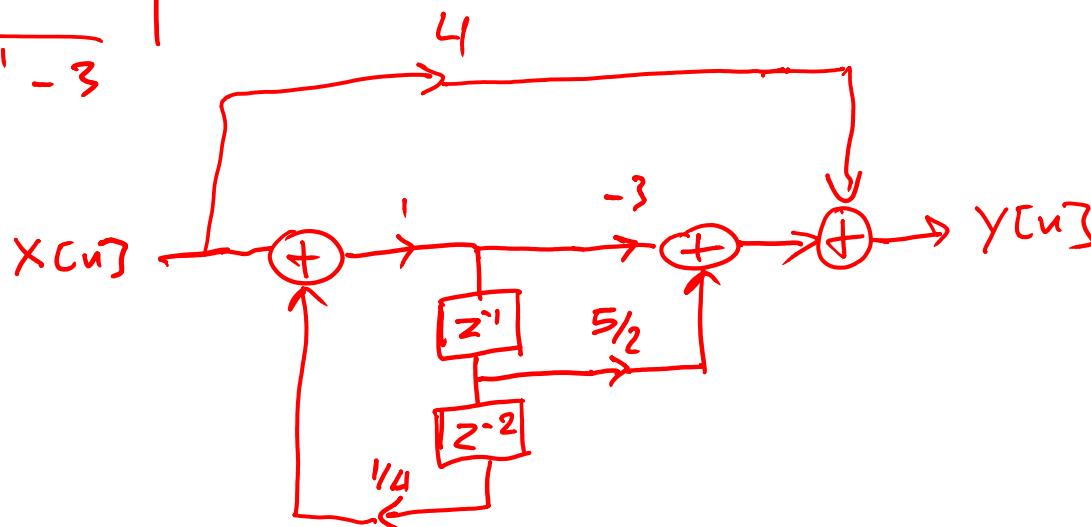
- Παράδειγμα:

○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 2.5z^{-1} - z^{-2}}{1 - 0.25z^{-2}} = 4 + \frac{\frac{5}{2}z^{-1} - 3}{1 - \frac{1}{4}z^{-2}} = 4 + \frac{A}{1 - \frac{1}{2}z^{-1}} + \frac{B}{1 + \frac{1}{2}z^{-1}}$$

Σχεδιάστε την υλοποίηση σε παράλληλη μορφή με υποσυστήματα 1^{ης} και 2^{ης} τάξης

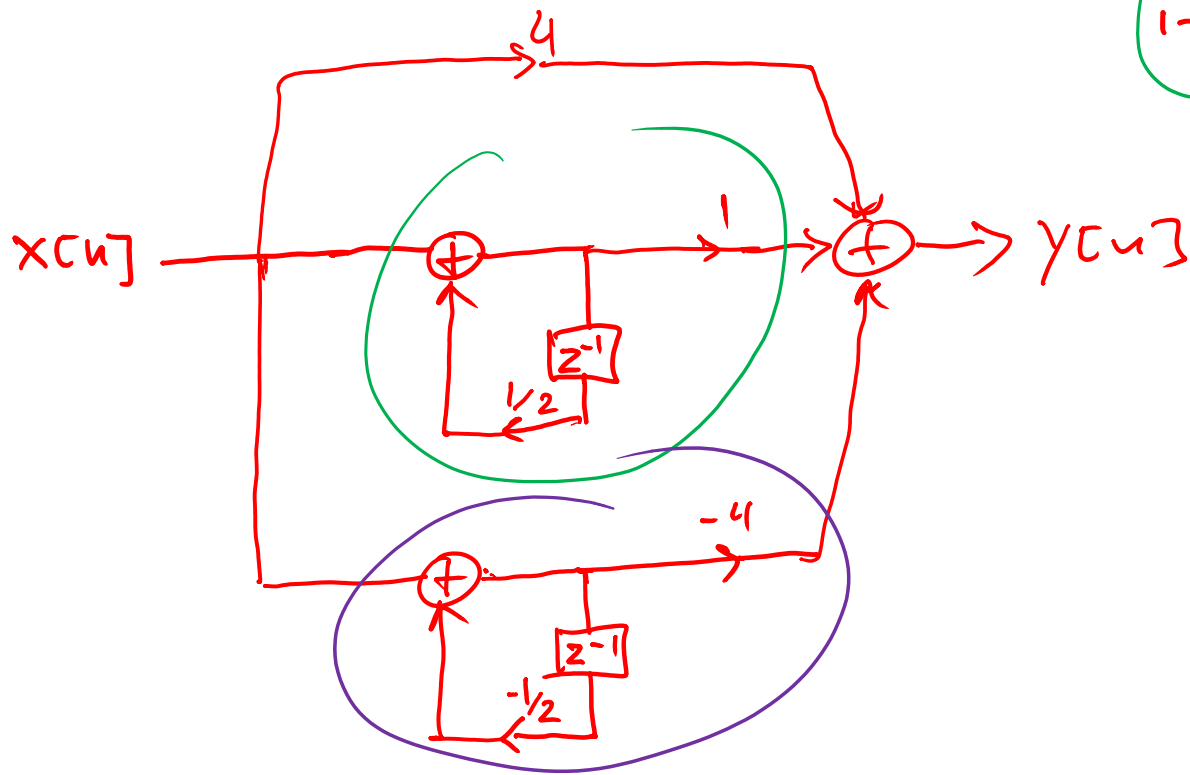
$$\begin{array}{r|l} -z^2 + 2.5z^{-1} + 1 & 1 - 0.25z^{-2} \\ +z^2 + 0 & -4 \\ \hline & 2.5z^{-1} - 3 \end{array}$$



- Υλοποίηση IIR συστημάτων διακριτού χρόνου

- Παράδειγμα:

$$\begin{aligned}
 H(z) &= 4 + \frac{-3 + \frac{5}{2}z^{-1}}{1 - \frac{1}{4}z^{-2}} = 4 + \frac{-3 + \frac{5}{2}z^{-1}}{(1 - \frac{1}{2}z^{-1})(1 + \frac{1}{2}z^{-1})} = 4 + \frac{A}{1 - \frac{1}{2}z^{-1}} + \frac{B}{1 + \frac{1}{2}z^{-1}} = \\
 &= 4 + \frac{1}{1 - \frac{1}{2}z^{-1}} - \frac{4}{1 + \frac{1}{2}z^{-1}}
 \end{aligned}$$



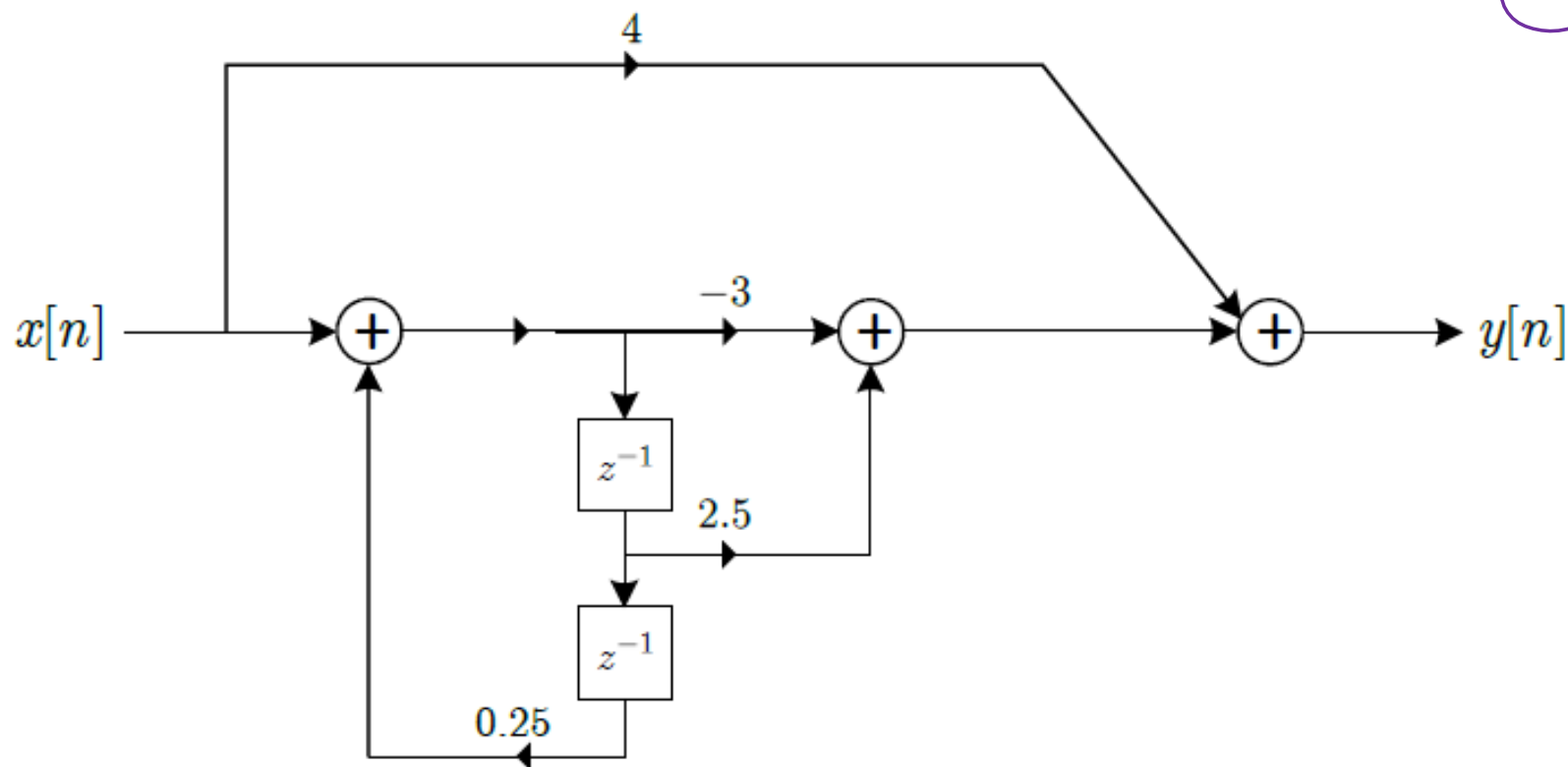
- Υλοποίηση IIR συστημάτων διακριτού χρόνου

- Παράδειγμα:

○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 2.5z^{-1} - z^{-2}}{1 - 0.25z^{-2}}$$

Σχεδιάστε την υλοποίηση σε παράλληλη μορφή με υποσυστήματα 1^{ης} και 2^{ης} τάξης



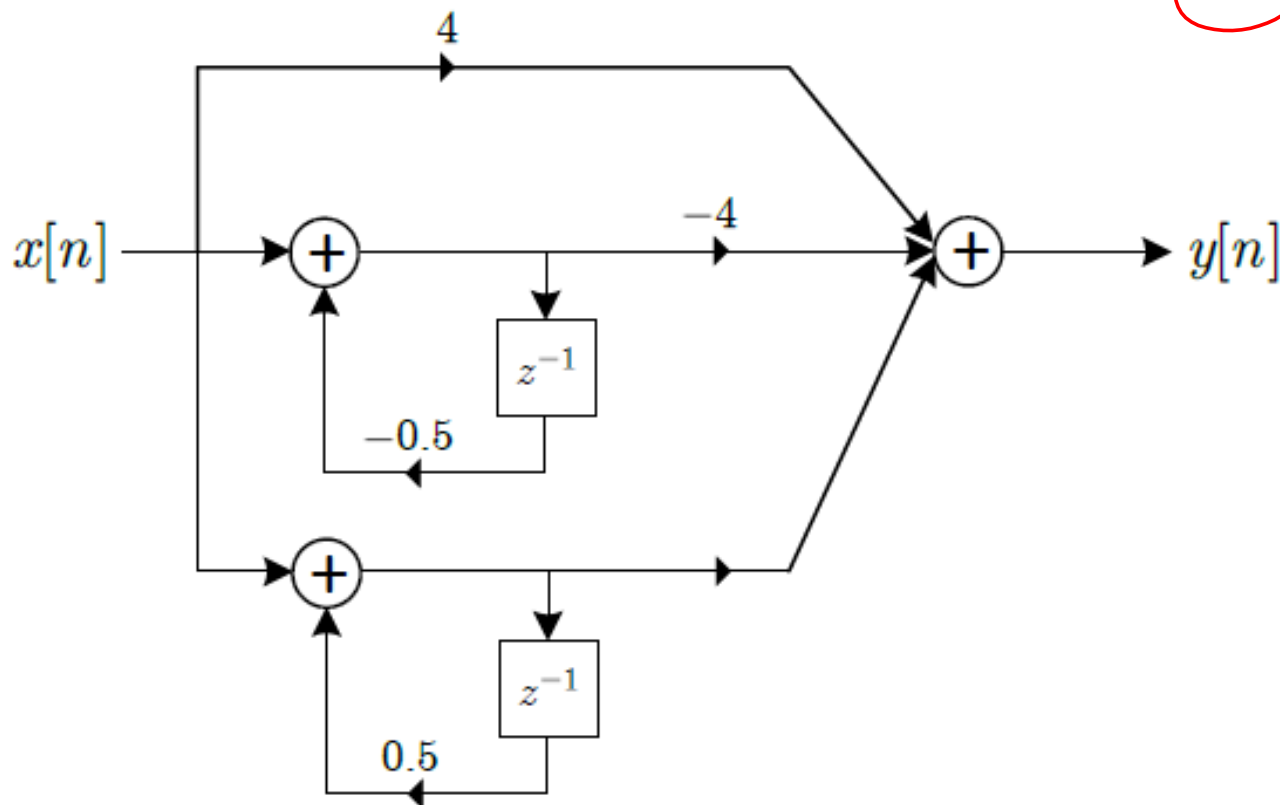
- Υλοποίηση IIR συστημάτων διακριτού χρόνου

- Παράδειγμα:

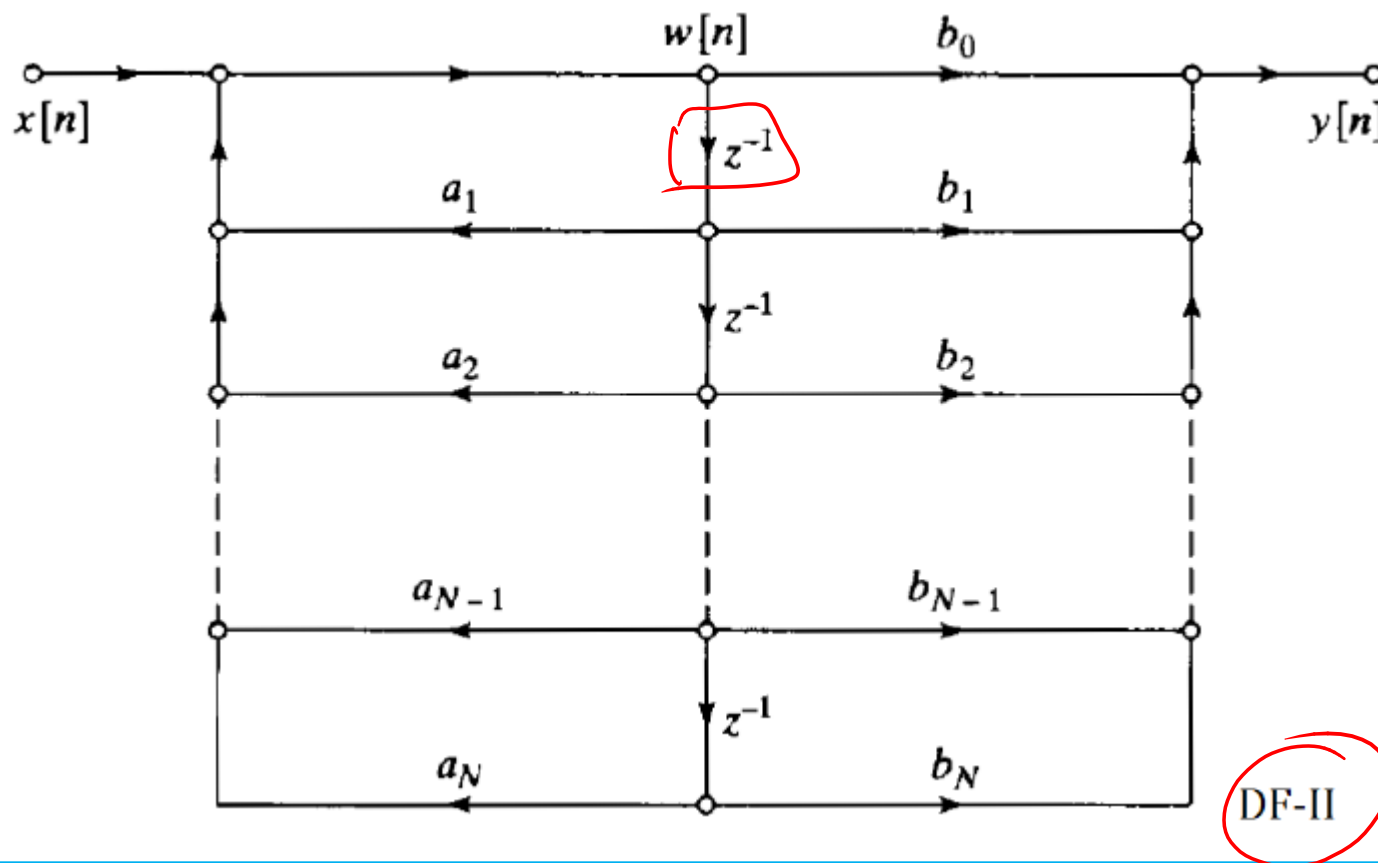
○ Έστω το ΓΧΑ σύστημα που δίνεται από τη συνάρτηση μεταφοράς

$$H(z) = \frac{1 + 2.5z^{-1} - z^{-2}}{1 - 0.25z^{-2}}$$

Σχεδιάστε την υλοποίηση σε παράλληλη μορφή με υποσυστήματα 1^{ης} και 2^{ης} τάξης



- Υλοποίηση IIR συστημάτων διακριτού χρόνου
- Ανάστροφη μορφή (transposed form)
- Τα διαγράμματα που σχεδιάζουμε ως τώρα μπορούν εναλλακτικά να αναπαρασταθούν ως γράφοι
- Για παράδειγμα, ο γράφος ενός γενικού IIR συστήματος σε Direct Form II



- **Υλοποίηση IIR συστημάτων διακριτού χρόνου**

- Ανάστροφη μορφή (transposed form)

- Η θεωρία γράφων μας προσφέρει μια διαδικασία που ονομάζεται αναστροφή (reversal) που μας δίνει μερικές εναλλακτικές μορφές δομών, με ενδιαφέρουσες ιδιότητες

- **Σχεδίαση Ανάστροφης Μορφής**



1. Αναστρέφουμε τη φορά όλων των κλάδων του διαγράμματος

Κάθε αθροιστής μετατρέπεται σε διακλάδωση, και το αντίστροφο

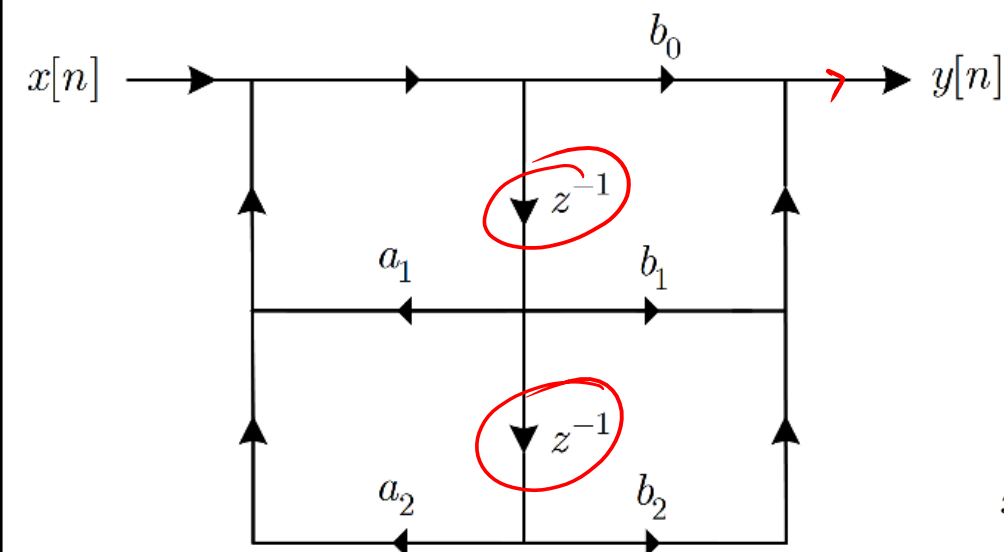
2. Αλλάζουμε θέση μεταξύ εισόδου και εξόδου



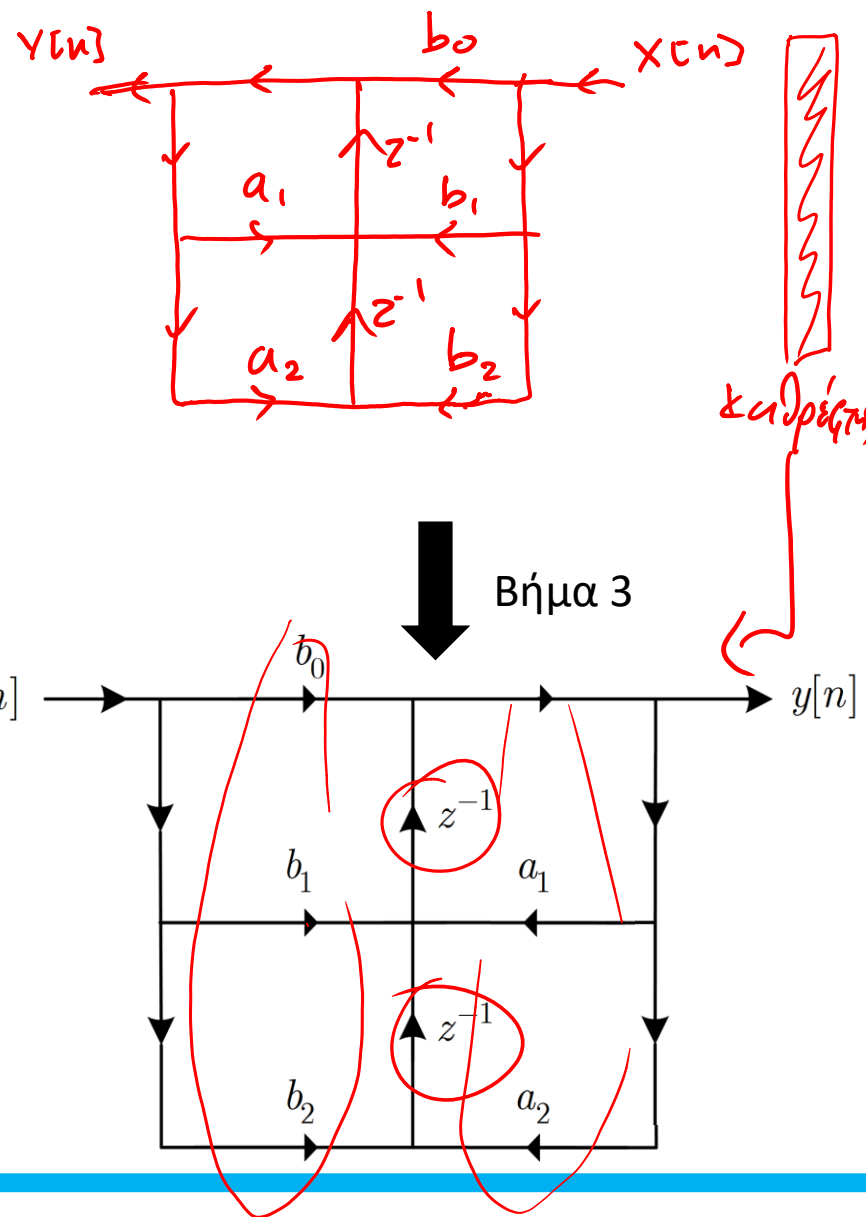
3. Επανασχεδιάζουμε το διάγραμμα, αναστρέφοντάς το ώστε να παρουσιάζεται η είσοδος στα αριστερά και η έξοδος στα δεξιά

- Ας το δούμε στην πράξη

- Υλοποίηση IIR συστημάτων διακριτού χρόνου
- Ανάστροφη μορφή (transposed form)
- Έστω ότι μας δίνεται ο Direct Form II γράφος όπως παρακάτω



- Αν εφαρμόσουμε τα βήματα που περιγράφει πριν, παίρνουμε το νέο διάγραμμα που φαίνεται στα δεξιά
- Προσέξτε ότι καταφέραμε χωρίς προσθήκη μνήμης να υλοποιήσουμε πρώτα τα μηδενικά και μετά τους πόλους!! 😊



ΤΕΛΟΣ ΔΙΑΛΕΞΗΣ

