

2.8.3 Σειριακή Είσοδος/Εξοδος (Z80 Serial I/O)

Το Z80 SIO είναι ένα ευέλικτο περιφερειακό για σειριακή επικοινωνία του Z80 με άλλα περιφερειακά ή άλλους μ/ε. Κατά την εκπομπή δεδομένων οι χαρακτήρες που θέλει να στείλει ο Z80 αποστέλλονται bit προς bit από μία φυσική γραμμή με σύγχρονο ή ασύγχρονο τρόπο. Κατά τη λήψη δεδομένων συμβαίνει η αντίστροφη διαδικασία δηλαδή τα δεδομένα λαμβάνονται bit προς bit από μία γραμμή και ομαδοποιούνται σε χαρακτήρες στο εσωτερικό του SIO ώστε ο Z80 να μπορεί να τα διαβάσει σε αυτή τη μορφή. Ο Z80 SIO διαθέτει 2 ανεξάρτητα κανάλια επικοινωνίας το A και το B.

Κατά το σύγχρονο τρόπο επικοινωνίας το Z80 SIO υποστηρίζει πρωτόκολλα IBM bisync (επιπέδου byte) και τα HDLC, IBM SDLC (επιπέδου bit). Είναι εντούτοις τόσο ευέλικτο ώστε μπορεί να υποστηρίζει και άλλα πρωτόκολλα σειριακής επικοινωνίας προγραμματίζοντάς το κατάλληλα. Για παράδειγμα διαθέτει γραμμές ελέγχου modem οι οποίες όταν δεν υπάρχει modem χρησιμοποιούνται ως γενικοί ακροδέκτες I/O.

Επίσης κατά το σύγχρονο τρόπο επικοινωνίας είναι δυνατή η δημιουργία ή ο έλεγχος κωδικού Cyclic Redundancy Check (CRC) ο οποίος χρησιμοποιείται για την επιβεβαίωση της ορθής (χωρίς λάθη) επικοινωνίας.

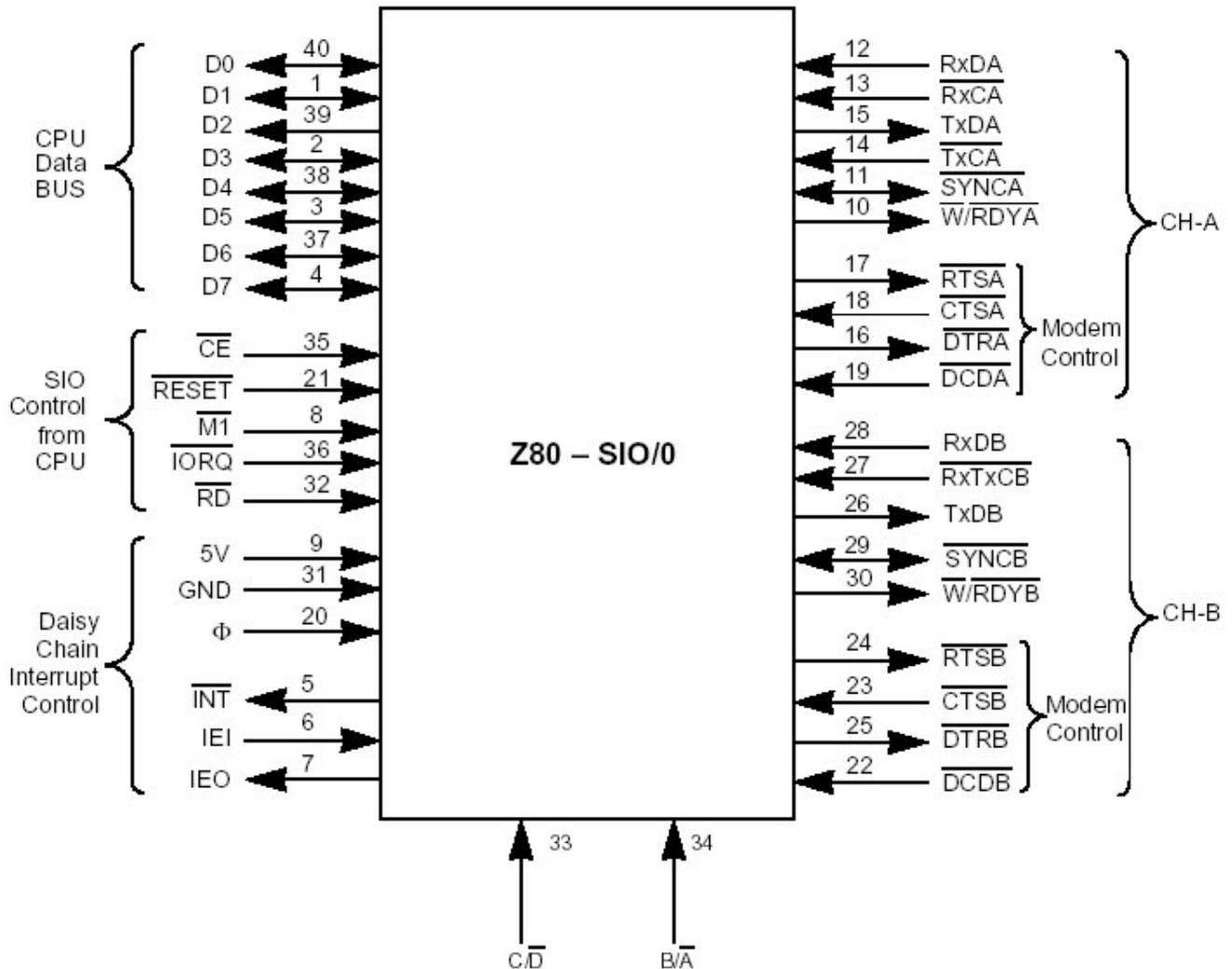
2.8.3.1 Ακροδέκτες του Z80 SIO

Το Z80 SIO αποτελείται από δύο ομάδες εξωτερικών ακροδεκτών, εκείνους που το διασυνδέουν απευθείας με τον Z80 χωρίς να χρειάζεται εξωτερική λογική και εκείνους που μεταφέρουν τα σειριακά σήματα και τα σχετικά με αυτά σήματα ελέγχου. Πιο συγκεκριμένα:

- D0-D7: Χρησιμοποιούνται για τη διασύνδεση με το δίαυλο δεδομένων του Z80.
- B/A~: Επιλογή του καναλιού που προσπελαύνει ο μ/ε.
- C/D~: Επιλογή ενός καταχωρητή ελέγχου ή δεδομένων. Τα B/A~ και C/D~ ουσιαστικά χρησιμοποιούνται για τη διευθυνσιοδότηση των εσωτερικών καταχωρητών του Z80 SIO και για το λόγο αυτό συχνά συνδέονται με τα A0 και A1 του διαύλου διευθύνσεων αντίστοιχα.
- CE~: Επιλογή του Z80 SIO ολοκληρωμένου. Χρησιμοποιείται για την αποκωδικοποίησή του, σε ορισμένη περιοχή διευθύνσεων όπως συμβαίνει με όλα τα chip enable σήματα.
- M1: Σε συνδυασμό με ενεργό σήμα RD δηλώνει τη χρονική στιγμή κατά την οποία προσκομίζεται μια εντολή από τη μνήμη προγράμματος. Το M1 σε συνδυασμό με ενεργό σήμα IORQ~ εκλαμβάνεται ως επιβεβαίωση διακοπής (απάντηση του μ/ε σε προηγούμενη αίτηση του Z80 SIO).
- IORQ~: Μπορεί να συνδεθεί με το αντίστοιχο σήμα του Z80 για την προσπέλαση των καταχωρητών του Z80 SIO ως IO ports και όχι ως θέσεις κύριας μνήμης.

- RD~: Όταν είναι ενεργό δηλώνει ότι πραγματοποιείται ανάγνωση από κάποιον καταχωρητή του Z80 SIO.
- RESET~: Επανεκκίνηση του Z80 SIO. Οι καταχωρητές θα πρέπει να αρχικοποιηθούν ξανά.
- IEI, IEO: Χρησιμοποιούνται για την ιεράρχηση των προτεραιοτήτων των διακοπών που προκαλούνται από διάφορα περιφερειακά όπως ο Z80 SIO. Το περιφερειακό που έχει τη μέγιστη προτεραιότητα έχει το IEI συνδεδεμένο στην τροφοδοσία και το IEO στο IEI του περιφερειακού με την αμέσως χαμηλότερη προτεραιότητα.
- INT~: Συνδέεται στο αντίστοιχο σήμα του Z80 και χρησιμοποιείται για την πρόκληση διακοπών.
- W/RDYA~, W/RDYB~: Σήματα εξόδου για τα κανάλια A και B που πολυπλέκουν δύο λειτουργίες. Αν χρησιμοποιηθούν ως Wait σήματα προσαρμόζουν την ταχύτητα προσπέλασης από τον μ/ε σε εκείνη που μπορεί να φτάσει ο Z80 SIO ενώ αν προγραμματιστούν ως Ready σήματα χρησιμοποιούνται για τη σύνδεση με DMA controller και τη μεταφορά δεδομένων από/προς την μνήμη χωρίς τη μεσολάβηση του μ/ε.
- CSTA~, CSTB~: Μπορούν να χρησιμοποιηθούν είτε ως Clear To Send σήματα είτε ως γενικής χρήσης ακροδέκτες εισόδου. Στη λειτουργία Clear To Send όταν ενεργοποιηθούν από κάποια εξωτερική συσκευή ειδοποιούν τον εκπομπό του αντίστοιχου καναλιού του Z80 SIO να αρχίσει μετάδοση.
- DCDA~, DCDB~: Παρόμοια με τα CTS σήματα για τη λήψη δεδομένων.
- RxDA, RxDB: Γραμμές λήψης δεδομένων.
- TxDA, TxDB: Γραμμές αποστολής δεδομένων.
- TxCA, TxCB: Ρολόι για την εκπομπή δεδομένων. Μπορεί να συνδεθεί απ'ευθείας με τις αντίστοιχες εξόδους του περιφερειακού Z80 CTC.
- RxCA, RxCB: Ρολόι για τη λήψη δεδομένων. Μπορεί να συνδεθεί απ' ευθείας με τις αντίστοιχες εξόδους του περιφερειακού Z80 CTC.
- RTSA~, RTSB~: Request To Send σήματα. Σε ασύγχρονη επικοινωνία ενεργοποιούνται όταν ο εκπομπός είναι έτοιμος να δεχτεί νέα δεδομένα προς αποστολή. Μπορούν να χρησιμοποιηθούν σαν έξοδοι γενικής χρήσης.
- DTRA~, DTRB~: Data Terminal Ready έξοδοι που απεικονίζουν την κατάσταση του DTR bit. Μπορούν να χρησιμοποιηθούν σαν γενικής χρήσης έξοδοι.
- SYNCA~, SYNCB~: Σε σύγχρονη επικοινωνία με εξωτερικό συγχρονισμό (external synchronization mode) χρησιμοποιούνται ως είσοδοι που ενεργοποιούνται όταν επιτευχθεί ο συγχρονισμός. Σε σύγχρονη επικοινωνία με εσωτερικό συγχρονισμό (internal synchronization mode) χρησιμοποιούνται ως έξοδοι που ενεργοποιούνται όταν αναγνωρίζονται SYNC χαρακτήρες.

Σε Dual In Package ολοκληρωμένα των 40 ακροδεκτών τα παραπάνω σήματα δεν μπορούν να βγουν όλα σε εξωτερικούς ακροδέκτες και είναι αναγκαστική κάποια επιπλέον πολυπλεξία. Πχ, στο Σχήμα 2-34 τα TxCB~, RxCB~ μοιράζονται κοινό ακροδέκτη



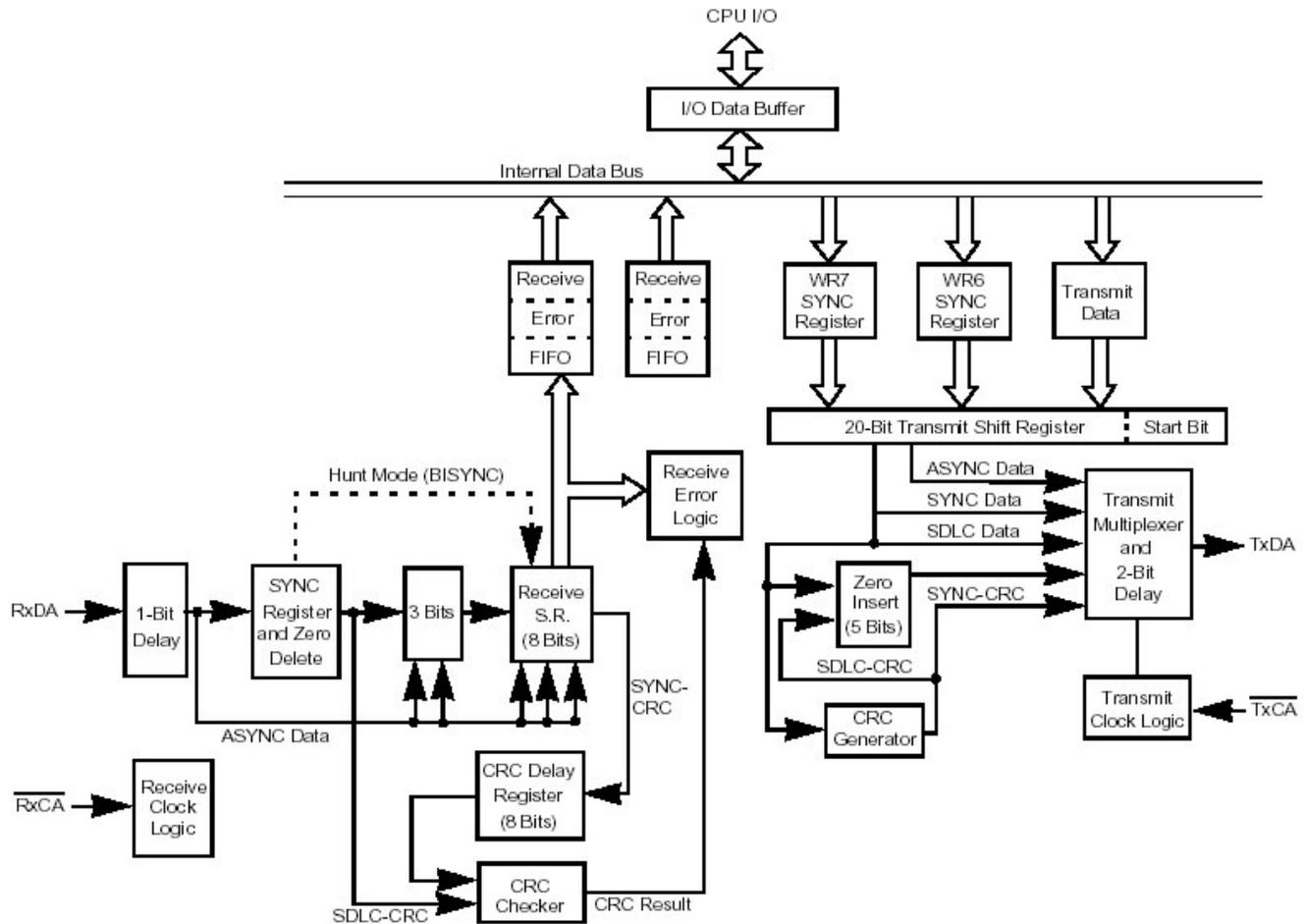
Σχήμα 2-34. Ακροδέκτες του Z80 SIO

2.8.3.2 Αρχιτεκτονική του Z80 SIO

Η γενική αρχιτεκτονική του Z80 SIO παρουσιάζεται στο Σχήμα 2-35. Στην πάνω πλευρά του σχήματος υπάρχουν τα τμήματα που είναι υπεύθυνα για την επικοινωνία με τον μ/ε, όπως είναι οι καταχωρητές ανάγνωσης/εγγραφής. Στην καρδιά του συστήματος υπάρχουν 2 καταχωρητές ολίσθησης: ένας 20-bit στην πλευρά του εκπομπού και ένας στην πλευρά του δέκτη με δύο τμήματα των 3-bit και των 8-bit. Το πως αυτά συνδυάζονται εξαρτάται από το πρωτόκολλο επικοινωνίας.

Στην πλευρά του εκπομπού υπάρχει γεννήτρια CRC το οποίο υπολογίζεται για έναν αριθμό bytes και αποστέλλεται μαζί με τα δεδομένα αν η δυνατότητα αυτή είναι ενεργοποιημένη. Στην πλευρά του δέκτη υπάρχει τμήμα που ελέγχει τα δεδομένα που φτάνουν κατά πακέτα (frames), υπολογίζει εκ νέου το CRC και το ελέγχει με αυτό

που έχει σταλεί. Αν είναι ίσα σημαίνει ότι η μεταφορά πραγματοποιήθηκε χωρίς σφάλματα.



Σχήμα 2-35. Αρχιτεκτονική του Z80 SIO

2.8.3.3 Πρωτόκολλα Επικοινωνίας

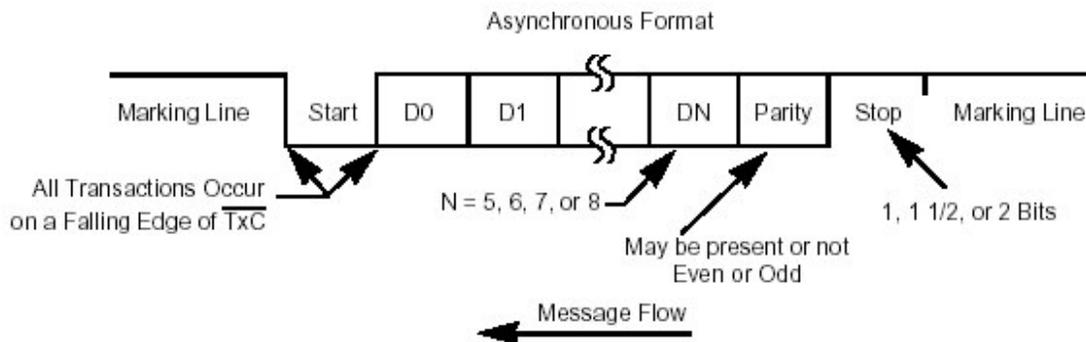
2.8.3.3.1 Ασύγχρονη Επικοινωνία (Asynchronous Mode)

Κατά την αρχικοποίηση σε ασύγχρονη επικοινωνία πρέπει να καθορισθεί το μήκος ενός χαρακτήρα, η ταχύτητα επικοινωνίας, τα start/stop bits, αν θα χρησιμοποιηθεί άρτια ή περιττή ισοτιμία για τον έλεγχο λαθών και οι πηγές διακοπών (interrupts). Όταν οριστούν όλα αυτά πρέπει να ενεργοποιηθεί ο εκπομπός και ο αποδέκτης.

Κατά την αποστολή δεδομένων προστίθενται αυτόματα τα start/stop bits καθώς και το ψηφίο ισοτιμίας. Η αποστολή γίνεται με ρυθμό 1/1 ή 1/16 ή 1/32 ή 1/64 σε σχέση με το ρολόι της γραμμής TxC και κατά την πίπτουσα ακμή του. Υπάρχει ένα bit με το όνομα Auto Enables το οποίο καθορίζει αν θα ελέγχεται η γραμμή CTS~ πριν την αποστολή δεδομένων. Αν το bit αυτό τεθεί στο 0 τότε το CTS~ καθώς και τα

DCD~, SYNC μπορούν να χρησιμοποιηθούν ως ακροδέκτες γενικού σκοπού. Ο μ/ε μπορεί να ειδοποιηθεί για την αποστολή νέου byte δεδομένων με διακοπή. Επίσης διακοπή μπορεί να προκληθεί όπως θα δούμε και από άλλες αιτίες όπως η αλλαγή σε κάποια από τα σήματα DCD~, CTS~, SYNC.

Στη λήψη δεδομένων ανιχνεύεται start bit όταν η γραμμή RxD παραμένει σε χαμηλό επίπεδο για κάποιο διάστημα. Ανάλογα με το πλήθος ψηφίων ανά χαρακτήρα που έχει οριστεί, δειγματοληπτείται η γραμμή RxC και συμπληρώνεται με άσσους αν το μήκος χαρακτήρα δεν είναι 8. Αν επιβάλλεται έλεγχος ψηφίου ισοτιμίας, αυτό υπολογίζεται χωρίς να προσμετρώνται οι συμπληρωματικοί άσσοι. Το νέο ψηφίο ισοτιμίας που προκύπτει συγκρίνεται με το parity bit που έχει αποσταλεί μαζί με τα δεδομένα. Αν δεν είναι ίδια τότε ενεργοποιείται το Parity Error bit στον καταχωρητή RR1 το οποίο παραμένει ενεργό μέχρι να διαβαστεί αυτή η κατάσταση από τον μ/ε. Αν δεν ανιχνευτεί stop bit προκύπτει Frame Error που απεικονίζεται στον ίδιο καταχωρητή (RR1) αλλά η τιμή του αντίστοιχου flag μεταβάλλεται σε κάθε νέο χαρακτήρα που λαμβάνεται ακόμα κι αν δεν διαβαστεί η κατάσταση του RR1. Επίσης, αν φτάσουν πάνω από 3 χαρακτήρες στον αποδέκτη χωρίς να τους διαβάσει ο μ/ε, προκαλείται Overrun Error (δες RR1). Γενικά ο μ/ε θα πρέπει να ελέγξει την κατάσταση του RR1 προτού διαβάσει τα δεδομένα από τον data register αλλιώς υπάρχει πιθανότητα στο χρόνο που θα μεσολαβήσει να χαθούν ενδείξεις λάθους ή χαρακτήρες δεδομένων.



Σχήμα 2-36. Ασύγχρονη Επικοινωνία

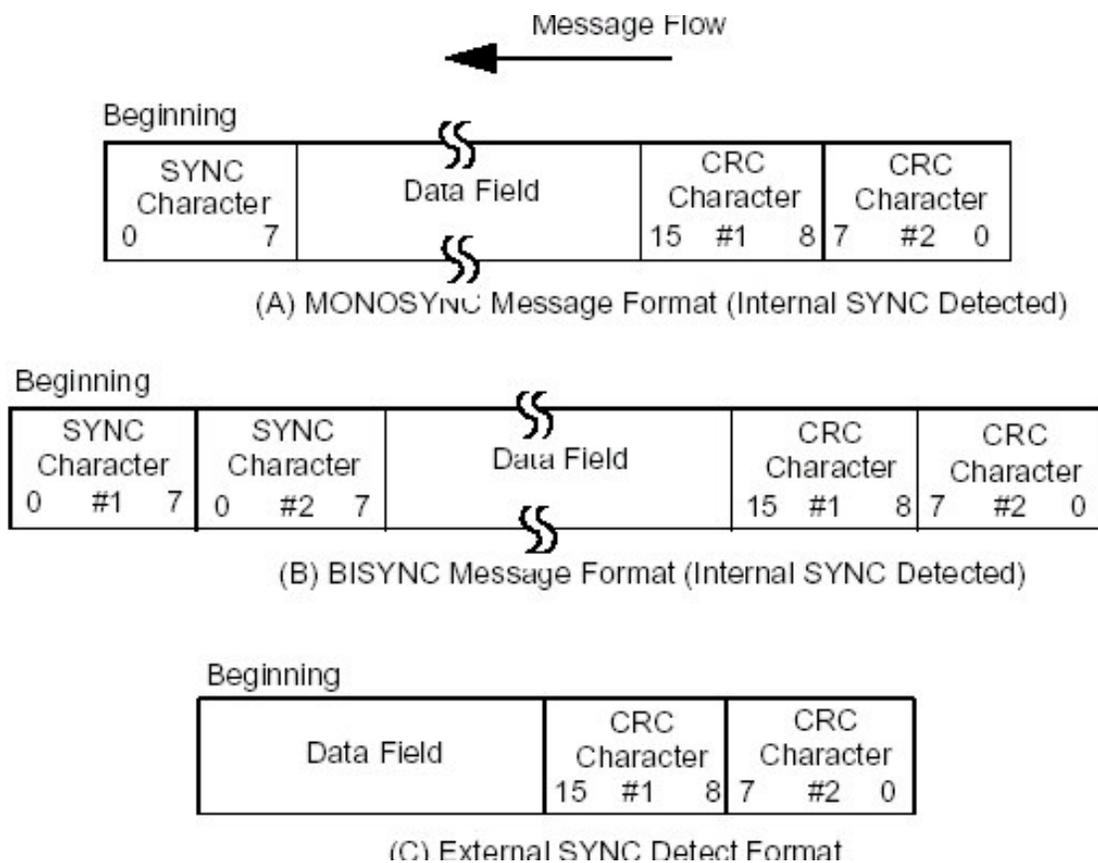
2.8.3.3.2 Σύγχρονη Επικοινωνία επιπέδου byte

Τρία είδη σύγχρονης επικοινωνίας υποστηρίζονται από το Z80 SIO τα οποία διαφέρουν στον τρόπο με τον οποίο επιτυγχάνεται ο αρχικός συγχρονισμός (Σχήμα 2-37):

- **Monosync.** Η επικοινωνία αρχίζει με έναν χαρακτήρα SYNC που στέλνει ο εκπομπός και η οποία είναι αποθηκευμένη στον WR6. Όταν ο δέκτης αντιληφθεί ότι ο χαρακτήρας που έλαβε ταιριάζει με τον αποθηκευμένο χαρακτήρα SYNC (στον καταχωρητής WR7) έχει επιτευχθεί συγχρονισμός. Ακολουθούν οι χαρακτήρες των δεδομένων και το πακέτο κλείνει με 2 χαρακτήρες CRC για τον έλεγχο λαθών όπως έχει ήδη περιγραφεί.

- Bisync. Όπως και στο Monosync με τη διαφορά ότι ο δέκτης συντονίζεται αν 2 συνεχόμενοι χαρακτήρες συγχρονισμού στην αρχή του πακέτου ανιχνευθούν και συγκριθούν με αυτούς που ο δέκτης έχει αποθηκευμένους στους καταχωρητές WR6, WR7.
- External Sync. Στην κατάσταση αυτή ο συγχρονισμός ελέγχεται από κάποια εξωτερική λογική η οποία ενεργοποιεί το σήμα SYNC~ του Z80 SIO όταν ανιχνευθεί συγχρονισμός.

Σε κάθε περίπτωση μετά την αρχικοποίηση ο δέκτης εισέρχεται σε Hunt Mode (Σχήμα 2-35) όπου προσπαθεί να ανιχνεύσει χαρακτήρα συγχρονισμού. Αφού επιτευχθεί συγχρονισμός εξέρχεται από Hunt Mode και η φάση λήψης δεδομένων και CRC είναι όμοια και στις 3 περιπτώσεις.



Σχήμα 2-37. Μέθοδοι σύγχρονης επικοινωνίας

Κατά τη σύγχρονη επικοινωνία ο εκπομπός μπορεί να ειδοποιηθεί τον μ/ε να γράψει τον επόμενο προς αποστολή χαρακτήρα είτε με τη βοήθεια διακοπής είτε με τη χρήση της γραμμής WAIT~/READY~. Επειδή ο εκπομπός θα πρέπει να αποστέλλει συνέχεια χαρακτήρες ακόμα κι αν ο μ/ε δεν έχει άμεσα κάτι να στείλει υπάρχει η επιλογή να στέλνονται ενδιάμεσα χαρακτήρες SYNC ή το CRC των χαρακτήρων δεδομένων που στάλθηκαν μέχρι εκείνη τη στιγμή ακολουθούμενο από SYNC χαρακτήρες. Όσον αφορά το ξεκίνημα υπολογισμού μιας νέας τιμής CRC

αυτό μπορεί να γίνει ενεργοποιώντας το Transmit CRC Enable bit. Το bit αυτό μπορεί να απενεργοποιηθεί αν δεν επιθυμούμε να ληφθούν υπόψη κάποιες τιμές στον υπολογισμό του CRC.

Ο οριστικός τερματισμός της εκπομπής γίνεται με την απενεργοποίηση του RTS bit και των σχετικών με τον εκπομπό διακοπών.

Στον αποδέκτη αφού γίνει αρχικοποίηση ανάλογη με εκείνη του εκπομπού υπάρχουν οι εξής επιλογές για να ειδοποιηθεί ο μ/ε σχετικά με την ολοκλήρωση λήψης ενός χαρακτήρα:

- Ο μ/ε ελέγχει με δική του ευθύνη το Z80 SIO (polled mode).
- Ο μ/ε ειδοποιείται με διακοπή για την άφιξη του πρώτου χαρακτήρα και στη συνέχεια διαβάζει τους επόμενους ελέγχοντας το σήμα WAIT~/READY~. Κάποιες συνθήκες όπως Overrun error και End of Frame μπορούν να προκαλέσουν διακοπές.
- Ο μ/ε ειδοποιείται με διακοπή για κάθε έναν χαρακτήρα που φτάνει. Αν έχει συμβεί σφάλμα η διακοπή συνοδεύεται από διαφορετική διεύθυνση ρουτίνας εξυπηρέτησης.

Για τον υπολογισμό και έλεγχο του CRC κατά τη λήψη το bit Receive CRC enable πρέπει να ενεργοποιηθεί την κατάλληλη στιγμή για να περιληφθούν στον υπολογισμό όλες οι απαραίτητες τιμές.

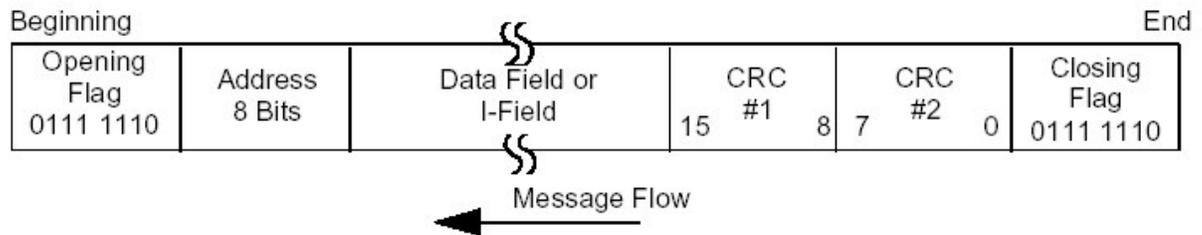
2.8.3.3.3 Σύγχρονη επικοινωνία επιπέδου bit (SDLC, HDLC)

Στην παράγραφο αυτή εξετάζονται άλλα δύο παρόμοια μεταξύ τους πρωτόκολλα σύγχρονης σειριακής επικοινωνίας όπου όμως ο συγχρονισμός επιτυγχάνεται σε επίπεδο bit. Αυτά είναι το IBM Synchronous Data Link Control (SDLC) και το High level synchronous Data Link Control (HDLC). Δεδομένης της ομοιότητας τους θα δοθούν λεπτομέρειες που αφορούν το SDLC και ισχύουν και στο HDLC.

Επειδή το SDLC είναι δομημένο σε επίπεδο bit υπάρχει μεγαλύτερη ευελιξία στη μορφή του πακέτου (frame) από το Bisync. Όπως στο Bisync το πακέτο ξεκινάει με συγκεκριμένους χαρακτήρες που ονομάζονται SYNC, έτσι και στο SDLC υπάρχει μια συγκεκριμένη ακολουθία bits (flag) τα οποία «σημαδεύουν» την αρχή ενός frame. Πχ, στο Σχήμα 2-38 η ακολουθία αυτή είναι το 0111 1110. Με την ίδια ακολουθία δυαδικών ψηφίων κλείνει επίσης το frame.

Το δεύτερο πεδίο που ακολουθεί στην αρχή του πακέτου είναι η διεύθυνση του παραλήπτη του μηνύματος. Ο αποδέκτης Z80 SIO έχει τη δυνατότητα να απομονώνει το πεδίο της διεύθυνσης και να το συγκρίνει με την δική του. Με αυτό τον τρόπο μπορεί να αποδεχτεί ή να απορρίψει ένα πακέτο αν δεν απευθύνεται σε αυτόν. Κάποιος θα μπορούσε να αναρωτηθεί τότε θα χρειαζόταν αυτό δεδομένου ότι στα προηγούμενα πρωτόκολλα έχουμε έναν αποστολέα και έναν αποδέκτη (η γραμμή TxD του ενός Z80 SIO συνδέεται με τη γραμμή RxD του δεύτερου και αντίστροφα). Χρησιμοποιώντας όμως το SDLC στο Z80 SIO και συνδέοντας modem ή γενικά ένα κατάλληλο φυσικό μέσο (physical module) στην έξοδο του το μικροϋπολογιστικό

σύστημα μπορεί να συνδεθεί σε κάποιο δίκτυο. Μετά τη διεύθυνση ακολουθεί το πεδίο των δεδομένων (που μπορεί να περιλαμβάνει και επιπλέον πληροφορία ελέγχου) και το πεδίο κλείνει με CRC για τον έλεγχο των λαθών.



Σχήμα 2-38. Μορφή ενός SDLC frame

Κατά την αρχικοποίηση του συστήματος θα πρέπει να ενεργοποιηθεί το SDLC, τα σήματα Request To Send και Data Terminal Ready, να επιλεγεί το SDLC CRC πολυώνυμο, να οριστεί το μήκος χαρακτήρα, να ενεργοποιηθούν οι κατάλληλες διακοπές κλπ. Ο μ/ε ειδοποιείται με διακοπή όταν ο καταχωρητής που γράφονται τα δεδομένα προς αποστολή είναι άδειος και μπορεί να δεχτεί νέο χαρακτήρα. Αν παρόλ' αυτά ο μ/ε δεν γράψει νέο χαρακτήρα θα προκύψει underrun κατάσταση και η αποστολή του τρέχοντος πακέτου θα τερματιστεί αφού προστεθεί το πεδίο CRC. Με τον τρόπο αυτό δεν χρειάζονται ειδικές χρονοβόρες διαδικασίες για να δηλωθεί στο Z80 SIO ο τερματισμός πακέτου. Μετά την αποστολή του CRC και του flag ο μ/ε είναι δυνατόν να ειδοποιηθεί ότι η αποστολή ενός καινούργιου πακέτου μπορεί να ξεκινήσει.

Ο αποδέκτης μετά την αρχικοποίηση εισέρχεται σε Hunt mode όπου ψάχνει να βρει flag στη γραμμή RxD. Το flag πρέπει να έχει τη μορφή που είναι αποθηκευμένη στα WR7, ενώ η διεύθυνση του πακέτου πρέπει να ταιριάζει με εκείνη που είναι αποθηκευμένη στο WR6. Αν ενεργοποιηθεί ένα κατάλληλο bit μπορεί να εισέρθει ο αποδέκτης σε κατάσταση ελέγχου διεύθυνσης οπότε θα προχωρήσει στο στάδιο της μεταφοράς χαρακτήρων δεδομένων στην μνήμη μόνο αφού διαπιστώσει ότι αυτά απευθύνονται σ' αυτόν. Αν δεν ταιριάζουν οι διευθύνσεις πακέτου με αποδέκτη απορρίπτεται η μεταφορά και ο αποδέκτης ξαναμπαίνει σε κατάσταση Hunt ψάχνοντας για το επόμενο flag αρχής πακέτου. Αν δεν ενεργοποιηθεί καθόλου ο έλεγχος διεύθυνσης τότε πιθανή διεύθυνση μέσα στο πακέτο θα αντιμετωπισθεί ως απλός χαρακτήρας δεδομένων.

Ο μ/ε μπορεί να ειδοποιηθεί για την άφιξη πακέτου με διακοπή κατά τον πρώτο χαρακτήρα ώστε να ξεκινήσει μπλοκ μεταφορά δεδομένων. Σε μια τέτοια μεταφορά μπορεί επίσης να ειδοποιηθεί με διακοπή αν φτάσει το τέλος του πακέτου ή αν προκύψει σφάλμα κατά τη μεταφορά. Υπάρχει επίσης η δυνατότητα να ενημερώνεται ο μ/ε σε κάθε άφιξη χαρακτήρα με ξεχωριστή διακοπή. Τέλος, ο μ/ε μπορεί να μην περιμένει καθόλου διακοπές αλλά να ελέγχει με δική του ευθύνη αν έχει φτάσει κάποιο καινούργιο πακέτο ή κάποιος νέος χαρακτήρας (polled mode). Όταν ληφθεί το flag αρχής ξεκινάει ο υπολογισμός του CRC ο οποίος ολοκληρώνεται στο τέλος του πακέτου το οποίο περιλαμβάνει και το αρχικό CRC που έστειλε ο αποστολέας. Κατά συνέπεια στο αποτέλεσμα της πράξης του CRC συνυπολογίζονται οι τιμές των χαρακτήρων δεδομένων και του αρχικού CRC. Μετά την ολοκλήρωση

της πράξης στο πεδίο του CRC θα πρέπει να υπάρχει η τιμή 0001 1101 0000 1111. Οι δύο αυτοί χαρακτήρες του CRC θα πρέπει να διαβαστούν από τον μ/ε για να καθαρίσει ο buffer του Z80 SIO.

2.8.3.4 Καταχωρητές του Z80 SIO

Η διευθυνσιοδότηση ενός καταχωρητή του Z80 SIO καθορίζεται πρωταρχικά από τις γραμμές B/A~ και C/D~.

C/D~	B/A~	Λειτουργία
0	0	Καταχωρητής δεδομένων του καναλιού A
0	1	Καταχωρητής δεδομένων του καναλιού B
1	0	Καταχωρητής ελέγχου/κατάστασης A
1	1	Καταχωρητής ελέγχου/κατάστασης B

Οι καταχωρητές ελέγχου είναι εγγράψιμοι καταχωρητές (WR0-WR7) ενώ οι καταχωρητές κατάστασης είναι μόνο ανάγνωσης (RR0-RR1).

2.8.3.4.1 WR0

Τα πεδία D0-D2 του WR0 καθορίζουν ποιος καταχωρητής θα αναγνωστεί ή θα εγγραφεί στην αμέσως επόμενη προσπέλαση του Z80 SIO. Τα πεδία D3-D5 ορίζουν μία από τις 7 πιθανές εντολές που απεικονίζονται στο Σχήμα 2-39. Η Send Abort χρησιμοποιείται από το SDLC για την αποστολή 8 ή 13 άσων που αναγνωρίζονται από τον αποδέκτη ως εντολή απόρριψης του τρέχοντος πακέτου. Η Reset Exit/Status Interrupts επιτρέπει ξανά διακοπές που προέρχονται από γραμμές ελέγχου όπως η DCD. Η Channel Reset αρχικοποιεί ξανά το τρέχον κανάλι το οποίο καθορίζεται από τη γραμμή B/A~. Η Enable Int on Next Char χρησιμοποιείται αν έχει επιλεγεί διακοπή μόνο στον πρώτο χαρακτήρα ενός πακέτου. Με αυτή την επιλογή μπορεί να προκληθεί διακοπή και κατά τη λήψη του επόμενου χαρακτήρα.

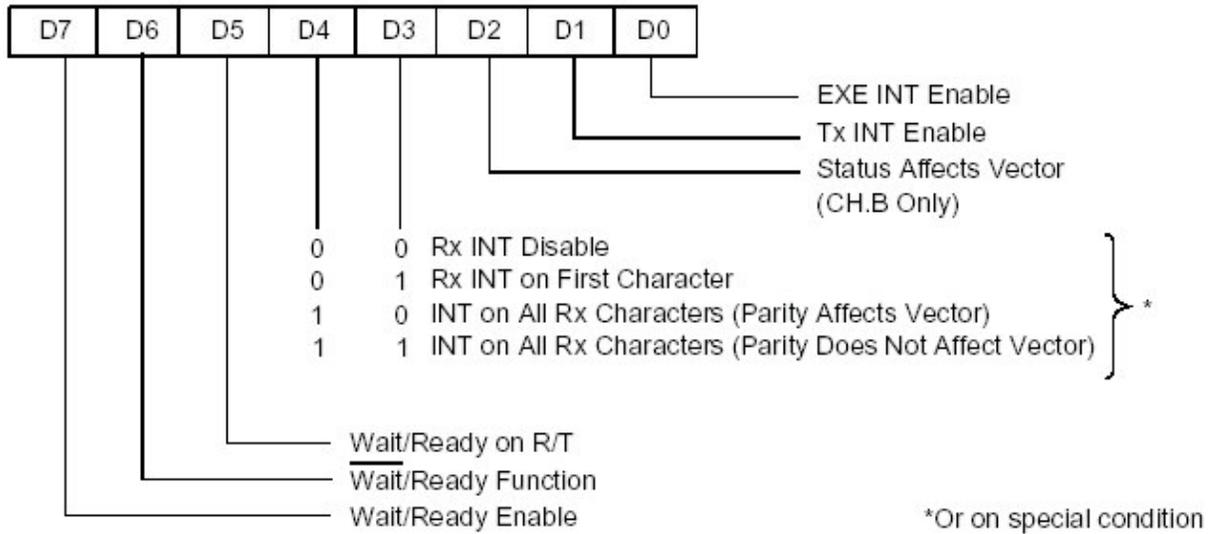
Η Reset Tx Interrupt Pending χρησιμοποιείται στο τέλος ενός πακέτου για να μην προκληθεί άλλη διακοπή όταν ο transmit data buffer μείνει άδειος μέχρι να ξεκινήσει ένα νέο πακέτο. Η Error Reset εντολή χρησιμοποιείται για να επαναφέρει τα flags του καταχωρητή ανάγνωσης που δείχνουν ότι έχει συμβεί Overrun ή Parity σφάλμα. Η Return From Interrupt πρέπει να εκτελείται κατά το τέλος της εξυπηρέτησης διακοπής του καναλιού A κατ' αντιστοιχία με την εντολή RETI. Τα bit 6 και 7 αφορούν αρχικοποίηση του CRC generator στον εκπομπό και του CRC checker στον αποδέκτη λειτουργία που απαιτείται να γίνεται όταν ξεκινάει η αποστολή ή λήψη ενός νέου πακέτου.

D7	D6	D5	D4	D3	D2	D1	D0	
					0	0	0	Register 0
					0	0	1	Register 1
					0	1	0	Register 2
					0	1	1	Register 3
					1	0	0	Register 4
					1	0	1	Register 5
					1	1	0	Register 6
					1	1	1	Register 7
		0	0	0				Null Code
		0	0	1				Send Abort (SDLC)
		0	1	0				Reset Ext/Status Interrupts
		0	1	1				Channel Reset
		1	0	0				Enable INT on Next Ax Character
		1	0	1				Reset TxINT Pending
		1	1	0				Error Reset
		1	1	1				Return from INT (CH-A Only)
0	0							Null Code
0	1							Reset Rx CRC Checker
1	0							Reset Tx CRC Generator
1	1							Reset Tx Underrun/EOM Latch

Σχήμα 2-39. Τα πεδία του WR0

2.8.3.4.2 WR1-WR2

Το D0 επιτρέπει διακοπές που προέρχονται από αλλαγές στις γραμμές DCD~,CTS~, SYNC~ ή κατά την ανίχνευση Abort ακολουθίας, ή κατά την αρχή SYNC ή CRC χαρακτήρα. Το Tx Int Enable (D1) επιτρέπει να προκληθούν διακοπές όταν ο transmit buffer είναι άδειος. Το Status Affects Vector αφορά το κανάλι B και όταν είναι 1 επιτρέπει να μεταβάλλεται το διάνυσμα διακοπής που είναι αποθηκευμένο στο WR2 ανάλογα με το ποιά αιτία προκάλεσε διακοπή (μεταβάλλονται τα bits V1-V3 του WR2, δεξ Σχήμα 2-41). Αν το Status Affects Vector είναι 0 τότε για οποιαδήποτε διακοπή καλείται η ίδια ρουτίνα εξυπηρέτησης διακοπής που διευθυνσιοδοτείται από την τιμή που είναι αποθηκευμένη στον **WR2**. Τα πεδία D3-D4 καθορίζουν πότε θα προκληθεί διακοπή κατά τη λήψη ενός πακέτου. Τέλος, τα D7-D5 ρυθμίζουν το αν το σήμα WAIT~/READY~ θα χρησιμοποιηθεί ως WAIT ή ως READY.



Σχήμα 2-40. Τα πεδία του WR1

	V3	V2	V1	Result
Ch B	0	0	0	Ch B Transmit Buffer Empty
	0	0	1	Ch B External/Status Change
	0	1	0	Ch B Receive Character Available
	0	1	1	Ch B Special Receive Condition*
Ch A	1	0	0	Ch A Transmit Buffer Empty
	1	0	1	Ch A External/Status Change
	1	1	0	Ch A Receive Character Available
	1	1	1	Ch A Special Receive Condition*
Note: *Special Receive Conditions: Parity Error, Rx Overrun Error, Framing Error, End-of-Frame (SDLC).				

Σχήμα 2-41. Πως μεταβάλλονται τρία bit του διανύσματος διακοπής στον WR2 όταν το Status Affects Vector του WR1 είναι ενεργό

2.8.3.4.3 WR3

Το πεδίο D0 του καταχωρητή αυτού είναι η γενική ενεργοποίηση του καναλιού λήψης. Το D1 προκαλεί την παράβλεψη των SYNC χαρακτήρων (μη φόρτωσή τους στον Rx Data buffer). Το πεδίο D2 ενεργοποιεί τον έλεγχο διεύθυνσης πακέτου. Αν η διεύθυνσή του δεν ταιριάζει με εκείνη που είναι αποθηκευμένη στον WR6 ή δεν είναι broadcast διεύθυνση (1111 1111) τότε ο μ/ε δεν ειδοποιείται με διακοπή να παραλάβει τους υπόλοιπους χαρακτήρες δεδομένων του πακέτου. Το D3

αρχικοποιεί ξανά τον υπολογισμό του CRC στον αποδέκτη ξεκινώντας από τον επόμενο χαρακτήρα που θα μεταφερθεί από τον καταχωρητή ολίσθησης. Στο πεδίο D4 αν γραφτεί 1 ο αποδέκτης εισέρχεται σε Hunt Mode όπου ψάχνει να βρει χαρακτήρα αρχής νέου πακέτου, πράγμα πολύ χρήσιμο στην περίπτωση που χαθεί συγχρονισμός. Το bit Auto Enables όταν γίνει 1 δίνει τη δυνατότητα στα σήματα DCD~ και CTS~ να ενεργοποιούν ή να απενεργοποιούν τον εκπομπό και αποδέκτη, αλλιώς τα σήματα αυτά μπορούν να χρησιμοποιηθούν σαν γενικής χρήσης είσοδοι που η στάθμη τους απεικονίζεται στον καταχωρητή RR0. Τέλος τα πεδία D6, D7 ορίζουν το πλήθος των bits από τα οποία θα αποτελείται κάθε χαρακτήρας.

Ας σταθούμε λίγο στο σημείο αυτό που θα μας βοηθήσει να καταλάβουμε την έννοια του bit oriented protocol την οποία έχει το SDLC. Ας υποθέσουμε ότι φτάνει στο πεδίο των δεδομένων από δεξιά προς αριστερά η ακολουθία

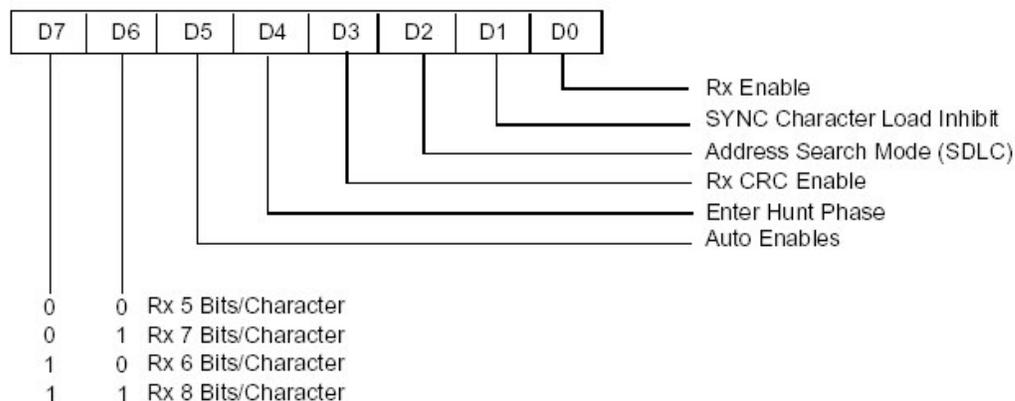
000100111010110100010101

Αν το πλήθος bit ανά χαρακτήρα έχει οριστεί 6 τότε θα προκύψουν οι εξής χαρακτήρες με την μορφή που θα τους διαβάσει ο μ/ε.

00-101010
00-001011
00-010111
00-001000

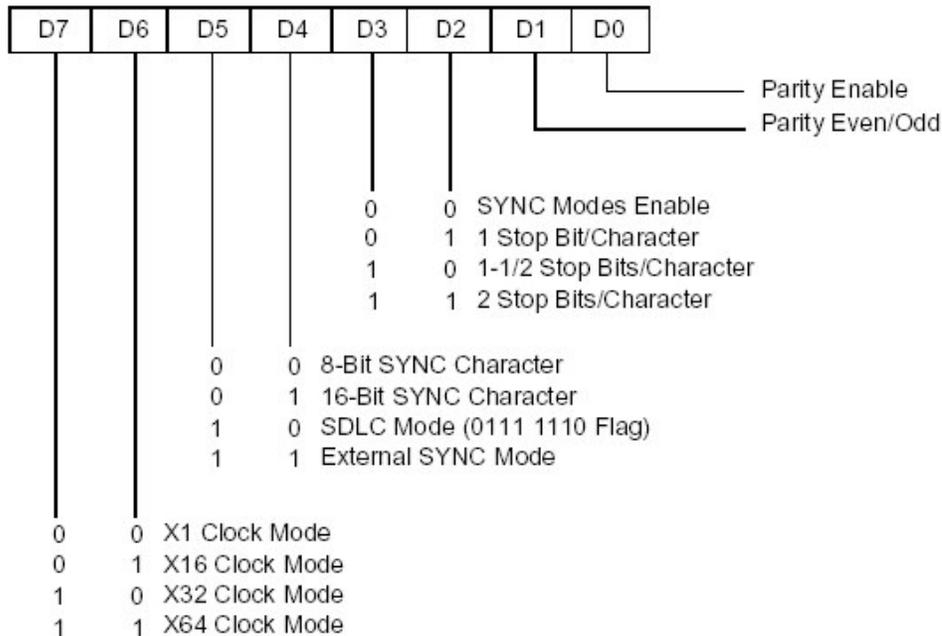
που προέκυψαν ομαδοποιώντας τα bits κατά 6 από δεξιά προς αριστερά και προσθέτοντας 2 μηδενικά στην αρχή. Αντίθετα, αν είχαμε ένα byte oriented πρωτόκολλο όπως το bisync οι χαρακτήρες που θα είχαν αναγνωριστεί θα ήταν

10101000
10110101
11001000



Σχήμα 2-42. Τα πεδία του WR3

2.8.3.4.4 WR4



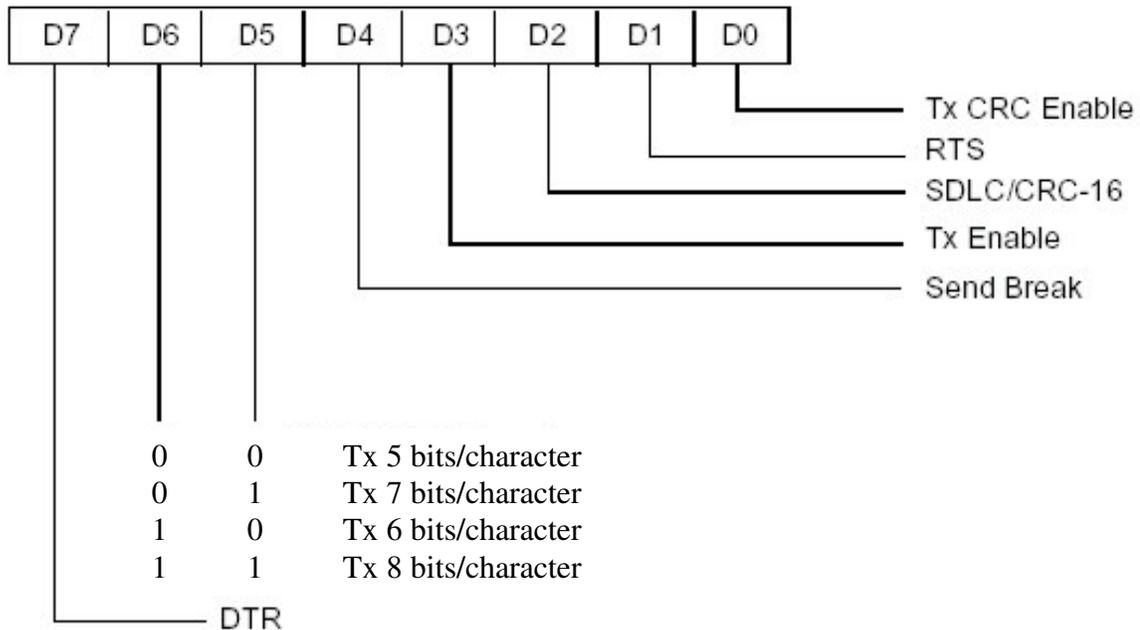
Σχήμα 2-43. Τα πεδία του WR 4

Το πεδίο D0 του WR4 δίνει τη δυνατότητα να προστεθεί σε κάθε χαρακτήρα ένα ψηφίο ισοτιμίας. Αν το μήκος του χαρακτήρα είναι <8 τότε το ψηφίο αυτό ενσωματώνεται στο χαρακτήρα. Το δεύτερο πεδίο D1 καθορίζει αν η ισοτιμία θα είναι άρτια ή περιττή. Τα D2-D3 ορίζουν το πλήθος των stop bits στην περίπτωση Ασύγχρονης επικοινωνίας. Στην περίπτωση Σύγχρονης επικοινωνίας θα πρέπει να επιλεγεί η τιμή 00 (SYNC). Τα D4-D5 ορίζουν το μήκος SYNC χαρακτήρα ή το πρωτόκολλο SDLC ή εξωτερικό συγχρονισμό ο οποίος δηλώνεται στη γραμμή SYNC. Τα πεδία D6-D7 καθορίζουν την ταχύτητα μετάδοσης δεδομένων σε σχέση με την ταχύτητα ρολογιού (RxC ή TxC). Σε σύγχρονη επικοινωνία πρέπει να επιλεγεί X1.

2.8.3.4.5 WR5

Όταν γραφτεί 1 στο πεδίο D0 ξεκινάει ο υπολογισμός του CRC για τους χαρακτήρες που θα αποσταλούν από το σημείο αυτό μέχρι να προκύψει Tx buffer underrun στο τέλος της αποστολής του πακέτου. Το D1 οδηγεί το σήμα RTS~ στη σύγχρονη επικοινωνία. Το D2 επιλέγει αν το πολώνυμο που θα χρησιμοποιηθεί στον υπολογισμό του CRC τόσο από τον εκπομπό όσο και από τον αποδέκτη θα είναι το CRC-16 (D2=1) ή το SDLC (D2=0). Το επόμενο bit (D3) είναι η γενική επίτρηση του αποστολέα. Τα δεδομένα δεν μεταδίδονται αν δεν γίνει 1 το πεδίο αυτό. Το D4 όσο είναι 1 επιβάλλει στον αποστολέα να στέλνει break χαρακτήρες. Τα D6, D5 καθορίζουν το πλήθος των bit ανά χαρακτήρα στην αποστολή. Ουσιαστικά ο μ/ε

γράφει ένα ολόκληρο byte στον Tx data buffer αλλά ο Z80 SIO αποστέλλει τα 5,6,7, ή 8 δεξιά bits του byte ανάλογα με τις τιμές των D6, D5. Τέλος, το D7 οδηγεί το σήμα DTR~.



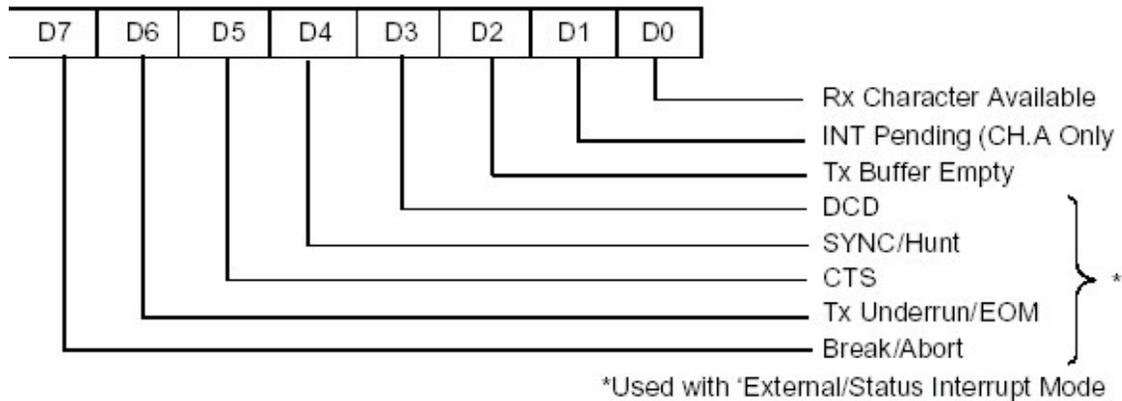
Σχήμα 2-44. Τα πεδία του WR5

2.8.3.4.6 WR6 και WR7

Ο καταχωρητής WR6 αποθηκεύει τον SYNC χαρακτήρα σε Monosync ή σε External Sync mode. Σε Bisync mode αποθηκεύει το λιγότερο σημαντικό byte του 16-bit SYNC. Σε SDLC mode αποθηκεύει τη διεύθυνση με του Z80 SIO με την οποία θα συγκρίνονται οι διευθύνσεις των πακέτων που λαμβάνονται.

Ο WR7 αποθηκεύει τον receive SYNC χαρακτήρα σε Monosync mode ή το πιο σημαντικό byte του 16-bit SYNC χαρακτήρα σε Bisync mode ή το flag (0111 1110) σε SDLC.

2.8.3.4.7 RR0

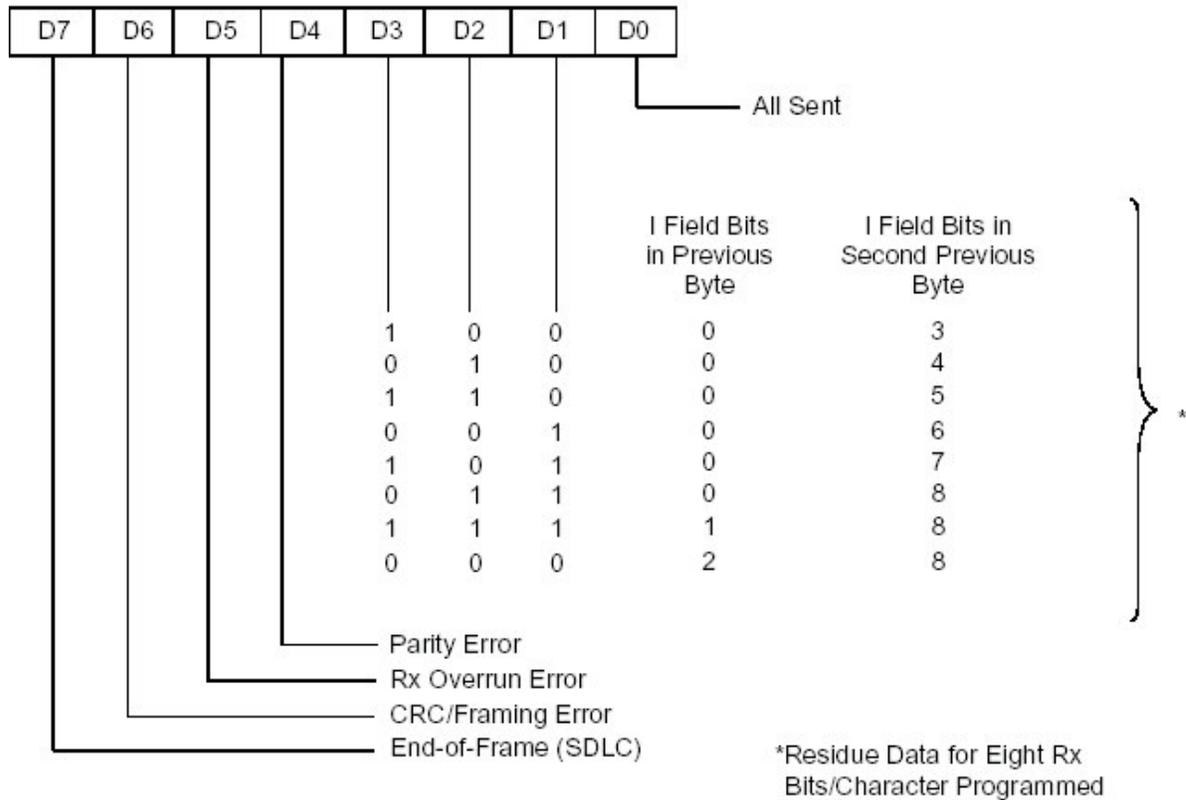


Σχήμα 2-45. Τα πεδία του RR0

Το πεδίο D0 γίνεται 1 όταν ολοκληρωθεί η λήψη ενός νέου χαρακτήρα στον αποδέκτη. Το D1 δείχνει αν υπάρχει κάποια διακοπή που δεν έχει εξυπηρετηθεί. Χρησιμοποιείται στην περίπτωση που δεν υποστηρίζεται αυτόματη διευθυνσιοδότηση της ρουτίνας εξυπηρέτησης από τον WR2. Όταν ο μ/ε διαβάσει το D2 ίσο με 1 τότε Tx data buffer είναι άδειος και έτοιμος να δεχθεί τον επόμενο χαρακτήρα που πρόκειται να αποσταλεί. Το D3 απεικονίζει τη στάθμη της γραμμής DCD~. Μεταβολή της στάθμης στη γραμμή αυτή είναι δυνατόν να προκαλέσει διακοπή οπότε ο μ/ε μπορεί να δει την τρέχουσα κατάσταση του σήματος από το bit αυτό. Το SYNC/Hunt δείχνει τη στάθμη της γραμμής SYNC σε Ασύγχρονη επικοινωνία. Στην περίπτωση External Synchronization Mode το πεδίο αυτό γίνεται 1 στη μετάβαση της γραμμής SYNC από υψηλό σε χαμηλό δυναμικό. Σε Monosync, Bisync mode το bit αυτό είναι αρχικά 1 μέχρι να επιτευχθεί συγχρονισμός οπότε γίνεται 0. Το CTS δείχνει την αντίστροφη κατάσταση της γραμμής CTS~. Τέλος, το bit Break/Abort ενεργοποιείται όταν ανιχνευθεί Break στη γραμμή RxD. Με την κατάλληλη διακοπή θα πρέπει να γίνει Reset ο αποδέκτης ώστε να ακυρωθεί η τρέχουσα λήψη πακέτου.

2.8.3.4.8 RR1

Το πεδίο D0 δείχνει όταν είναι 1 ότι σε Ασύγχρονη μετάδοση όλοι οι χαρακτήρες στον Tx buffer και στην ουρά του αποστολέα έχουν μεταδοθεί. Σε σύγχρονη μετάδοση είναι συνεχώς 1. Τα D3 έως D1 χρησιμοποιούνται για την οριοθέτηση του πεδίου I-field σε SDLC mode. Το D4 γίνεται 1 όταν κατά τη λήψη προκύψει σφάλμα ισοτιμίας και το D5 δηλώνει σφάλμα Overrun κατά τη λήψη. Το D6 δηλώνει Framing error σε Ασύγχρονη επικοινωνία και CRC σφάλμα σε Σύγχρονη. Το D7 δηλώνει ότι έχει ανιχνευθεί τέλος ενός SDLC frame. Δεν έχει χρήση σε άλλα πρωτόκολλα.



Σχήμα 2-46. Τα πεδία του RR1

2.8.3.4.9 RR2 (Κανάλι B)

Όπως έχει ήδη αναφερθεί, το διάνυσμα διακοπής που είναι αποθηκευμένο στον WR2 μπορεί να μεταβληθεί ανάλογα με την πηγή της διακοπής. Διαβάζοντας τον RR2 (μόνο από το κανάλι B) ο μ/ε μπορεί να δει ποια είναι η τρέχουσα τιμή του διανύσματος ανάλογα με την πηγή διακοπής που έχει τη μεγαλύτερη προτεραιότητα. Αν δεν υπάρχει διακοπή που δεν έχει εξυπηρετηθεί τότε αυτό που διαβάζεται από τον RR2 είναι η τιμή του WR2 με 0,1,1 στα bits D3, D2, D1 αντίστοιχα.

2.8.3.5 Παράδειγμα Προγραμματισμού του Z80 SIO σε SDLC mode

Σαν παράδειγμα προγραμματισμού του Z80 SIO θα παρουσιαστεί το στάδιο αρχικοποίησης του εκπομπού του καναλιού B σε SDLC mode, καθώς και η μορφή της ρουτίνας εξυπηρέτησης διακοπής για την εγγραφή του επόμενου byte. Με παρόμοιο τρόπο πρέπει να γίνει και η αρχικοποίηση στον αποδέκτη, ενώ θα πρέπει να γραφούν αντίστοιχες ρουτίνες εξυπηρέτησης για σφάλματα μετάδοσης. Ιδιαίτερη προσοχή θα πρέπει να δοθεί στο πως προκύπτει μια διεύθυνση ρουτίνας διακοπής με

τη χρήση του πεδίου Status Affects Vector και το διάνυσμα που αποθηκεύεται στο WR2.

Υποθέτουμε ότι το Z80 SIO αποκωδικοποιείται στο port address 0x80 και η γραμμή B/A~ συνδέεται στο A0 ενώ η C/D~ στο A1. Κατά συνέπεια στο 0x80 βρίσκεται ο data buffer του καναλιού A (συμβολικό όνομα DA), στο 0x81 ο data buffer του καναλιού B (DB), στο 0x82 οι καταχωρητές ελέγχου/κατάστασης του A (CA) και στο 0x83 του B (CB). Οι τιμές των καταχωρητών WR δίνονται σε δυαδική τιμή τα ορίσματα των εντολών για ευκολία στην ανάγνωση.

```
; Initialization of Interrupts on Z80 CPU
IM 2          ; Interrupt Mode 2 with vectored interrupts
LD A,F0       ; Load Accumulator with the high order byte of
              ; the interrupt vector
LD IV,A       ; Load Interrupt Vector High Order Byte with F0
EI            ; Ensure global interrupts are allowed
; Initialization of Z80 SIO
LD A,10000000 ; Reset Tx CRC
OUT CB,A      ; Store to WR0
LD A,00000010 ; Prepare WR2
OUT CB,A      ; Store to WR0
LD A,00100000 ; Base Interrupt Vector=0x20
OUT CB,A      ; Store to WR2
LD A,00000011 ; Prepare WR3
OUT CB,A      ; Store to WR0
LD A,00000000 ; Don't use AutoEnables
OUT CB,A      ; Store to WR3
LD A,00010100 ; Prepare WR4 and Reset Ext. Status Int
OUT CB,A      ; Store to WR0
LD A,00100000 ; Clk x1, SDLC, SYNC, No Parity
OUT CB,A      ; Store to WR4
LD A,00000001 ; Prepare WR1
OUT CB,A      ; Store to WR0
LD A,10000111 ; Status Affects Vector, TxInt Enable,
              ; Wait function selected to notify CPU
              ; for a next byte transfer
OUT CB,A      ; Store to WR1
LD A,00000001 ; Prepare WR5
OUT CB,A      ; Store to WR0
LD A,00101111 ; Tx Enable, CRC Enable, RTS, SDLC
OUT CB,A      ; Store to WR5
...
; Concerning that base low order byte of interrupt vector
; is 0x20, the resulting low order byte of interrupt vector
; for channel B Tx buffer empty is 0x30. The high order
; byte is stored in IV register.
;
0xF030:      ; ISR for Tx buffer empty
EX AF,AF'
EXX          ; Save current contents of registers
LD A,data    ; data:the character to be transmitted
OUT DB,A     ; Store to data buffer of channel B
LD A,11000000 ; Reset Tx Underrun
OUT CB,A     ; Store to WR0
;
```

```
; if this is character contains the last I-field bit
; the current frame must be closed properly by the Z80 SIO
; (CRC and flag must be added). This is done if the
; following command is issued instead of the previous
; 4 commands
;
LD A,00101000          ; Reset Interrupt Pending
OUT CB,A
EX AF,AF'
EXX                    ; Save current contents of registers
```

2.8.4 Απευθείας Προσπέλαση Μνήμης (Z80 DMA Controller)

Τα συνηθισμένα περιφερειακά που επιτελούν Απευθείας Προσπέλαση στη Μνήμη (Direct Memory Transfer Controllers-DMAC) έχουν την ικανότητα να μεταφέρουν μεγάλο πλήθος δεδομένων από περιφερειακές συσκευές στην κύρια μνήμη ή το αντίστροφο με μεγάλη ταχύτητα αφού η μεταφορά γίνεται χωρίς τη μεσολάβηση της ΚΜΕ. Ουσιαστικά ένας DMAC οδηγεί το δίαυλο διευθύνσεων (Address bus) με τις κατάλληλες διευθύνσεις καθώς και τις γραμμές ελέγχου με τέτοιο τρόπο ώστε να εκτελεσθεί μια μεταφορά. Το αντίστοιχο περιφερειακό του Z80 μπορεί να επιτελέσει μερικές επιπλέον λειτουργίες που κανονικά θα πραγματοποιούνταν με ευθύνη της ΚΜΕ:

- Μεταφορά δεδομένων από μια θέση μνήμης σε άλλη,
- από I/O χώρο στη μνήμη και αντίστροφα, και
- αναζήτηση μιας συγκεκριμένης τιμής είτε ανεξάρτητα είτε κατά τη διάρκεια ταυτόχρονης μεταφοράς.

Σε πολλά συστήματα με επεξεργαστές διαφορετικούς από τον Z80 η μεταφορά των δεδομένων μεταξύ περιφερειακών και κύριας μνήμης είναι σημαντικά ταχύτερη όταν χρησιμοποιείται DMAC από την περίπτωση που η μεταφορά γίνεται με τη μεσολάβηση της ΚΜΕ. Όμως δεδομένου ότι ο Z80 διαθέτει εντολές για αυτόματη μεταφορά ολοκλήρου μπλοκ δεδομένων η χρήση DMAC είναι συχνά περιττή. Η χρήση του Z80 DMAC έχει εντούτοις άλλα πλεονεκτήματα που σχετίζονται με το χρόνο αντίδρασης σε μια αίτηση μεταφοράς δεδομένων μεταξύ περιφερειακού και κύριας μνήμης. Σε περίπτωση που πολλές συσκευές έχουν ανάγκη για επικοινωνία με τη μνήμη μέσω DMA περισσότεροι του ενός DMAC μπορούν να χρησιμοποιηθούν όπως θα φανεί στη συνέχεια.

Σε ένα σύστημα που βασίζεται στον Z80 αλλά δεν έχει DMAC ο Z80 έχει τους εξής τρόπους να αντιδράσει σε μία αίτηση περιφερειακού για μεταφορά δεδομένων:

- Το περιφερειακό προκαλεί διακοπή (interrupt) στον Z80 και αυτός δρομολογεί μια μπλοκ μεταφορά δεδομένων στη ρουτίνα εξυπηρέτησης της διακοπής. Αυτή η μέθοδος έχει χρόνο αντίδρασης 5-10 μs σε σύστημα που λειτουργεί με ρολόι 4MHz. Ο χρόνος αυτός θεωρείται αρκετά μεγάλος.
- Ο Z80 ελέγχει συνεχώς την περιφερειακή συσκευή ακόμα κι αν αυτή δεν είναι έτοιμη και όταν διαπιστωθεί ανάγκη μεταφοράς αυτή ξεκινάει με άλμα σε εντολή μπλοκ μεταφοράς δεδομένων. Αυτή η μέθοδος απαιτεί λιγότερα από 5 μs χρόνο αντίδρασης αλλά ο μ/ε δεσμεύεται ελέγχοντας συνεχώς την περιφερειακή συσκευή (polling)
- Ο Z80 ξεκινάει μια μπλοκ μεταφορά δεδομένων χωρίς να ελέγξει την περιφερειακή συσκευή. Η συσκευή χρησιμοποιεί το σήμα WAIT για να αναβάλλει τη μεταφορά όσο χρειάζεται. Αυτή η μέθοδος έχει χρόνο

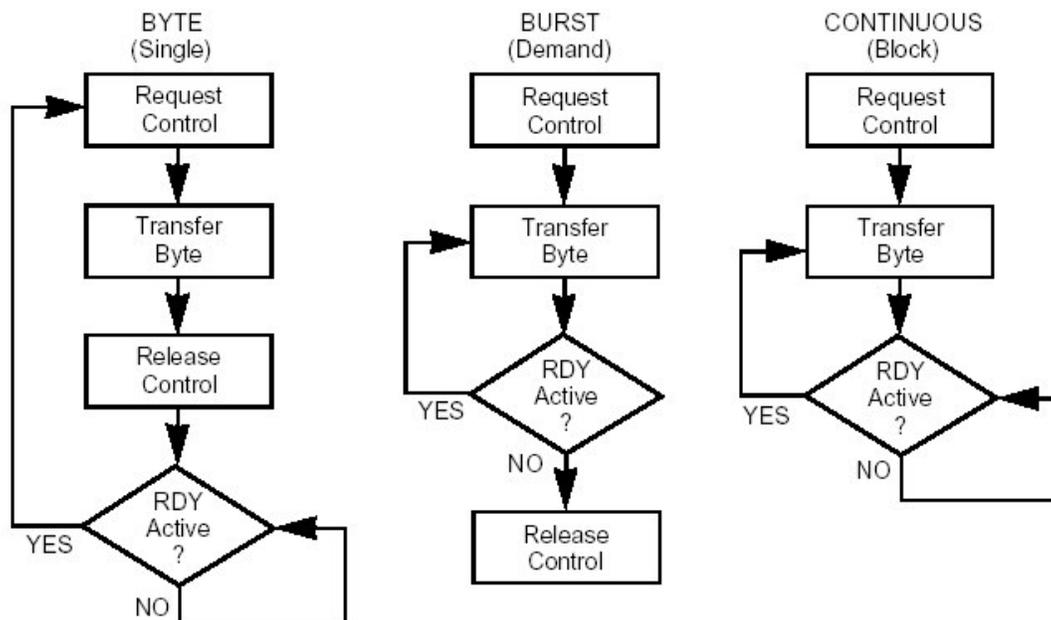
αντίδρασης της τάξης των 250ns αλλά έχει επίσης το σημαντικό μειονέκτημα ότι αδρανοποιεί τους διαύλους του συστήματος όσο η γραμμή WAIT είναι ενεργή.

Όταν χρησιμοποιείται DMAC οι χρόνοι αντίδρασης και μεταφοράς δεδομένων είναι οι βέλτιστοι χωρίς να υπάρχουν μειονεκτήματα όπως αυτά που αναφέρθηκαν παραπάνω. Η διαδικασία της μεταφοράς δεδομένων περιλαμβάνει αρχικά την εγγραφή από τον Z80 στο DMAC της διεύθυνσης αρχής του μπλοκ και του μεγέθους των δεδομένων. Η αρχικοποίηση αυτή είναι σημαντικό να τονιστεί ότι γίνεται σε **μη κρίσιμο χρόνο**. Η μεταφορά ξεκινάει αμέσως όταν η συσκευή δηλώσει στο DMAC ότι είναι έτοιμη μέσω του σήματος READY.

Συνοπτικά η DMAC του Z80 χρησιμοποιείται σε συστήματα όπου ισχύει ένα από τα ακόλουθα:

- Υπάρχει ανάγκη για συνεχείς I/O μεταφορές και η ΚΜΕ δεν μπορεί να ασχολείται μόνο με αυτές.
- Η ταχύτητα I/O μεταφοράς δεδομένων που μπορεί να επιτύχει η ΚΜΕ δεν είναι επαρκής.
- Ο χρόνος αντίδρασης σε μια αίτηση I/O μεταφοράς είναι μικρότερος από αυτόν που μπορεί να επιτύχει η ΚΜΕ.

2.8.4.1 Καταστάσεις Λειτουργίας του Z80 DMAC

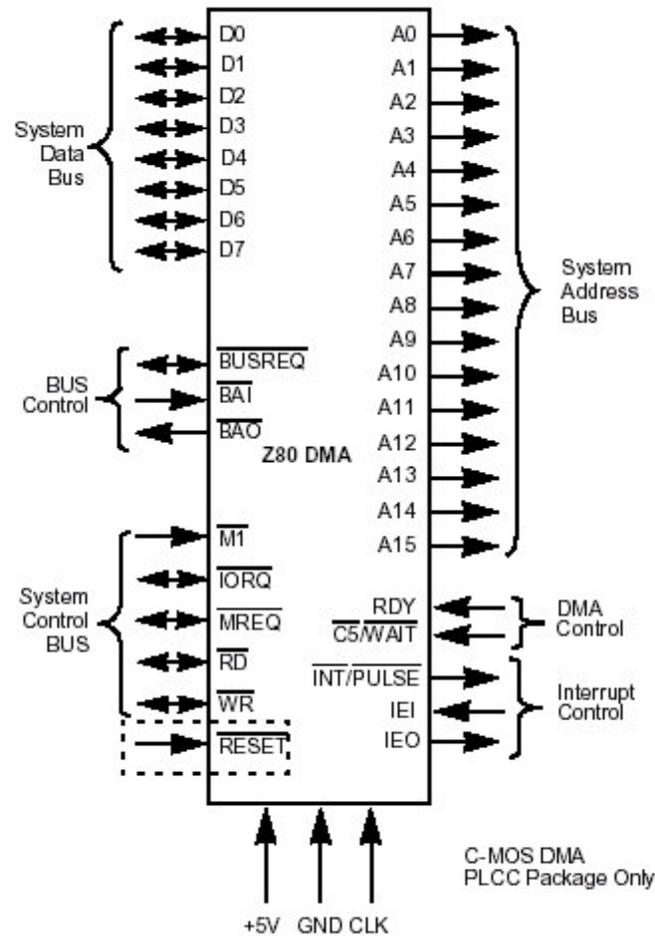


Σχήμα 2-47. Καταστάσεις Λειτουργίας του Z80 DMAC

Όπως οι περισσότερες περιφερειακές συσκευές έτσι και ο Z80 DMAC είναι προγραμματιζόμενος, δηλαδή διαθέτει κάποιους καταχωρητές διαμόρφωσης (configuration registers) οι οποίοι πρέπει να προγραμματιστούν από την ΚΜΕ. Η ΚΜΕ πρέπει να προγραμματίσει τουλάχιστον τη διεύθυνση του μπλοκ δεδομένων και το μέγεθος του. Σε κάθε μεταφορά πρέπει να είναι η γνωστή η διεύθυνση από την οποία θα ληφθούν τα δεδομένα και η διεύθυνση προορισμού τους (source and destination address). Η πηγή και ο προορισμός μιας DMA μεταφοράς γίνονται μέσω δύο θυρών (ports) που διαθέτει ο DMAC: το Port A και το Port B. Για κάθε byte που μεταφέρεται, οι διευθύνσεις πηγής και προορισμού ενημερώνονται (πχ, αυξάνονται κατά 1), όπως επίσης και η τιμή ενός μετρητή ώστε να είναι έτοιμες για τη μεταφορά του επόμενου byte. Η διαδικασία τερματίζεται όταν η τιμή του μετρητή γίνει ίση με το μέγεθος που έχει ορίσει προηγουμένως η ΚΜΕ.

Υπάρχουν 3 καταστάσεις (modes) Λειτουργίας: Byte, Burst και Continuous mode. Κατά τη μεταφορά σε Byte mode ουσιαστικά γίνεται επικάλυση λειτουργιών προσπέλασης της κύριας μνήμης από την ΚΜΕ και το DMAC. Σε Burst mode ο DMAC χειρίζεται κατ' αποκλειστικότητα τους διαύλους συστήματος μέχρις ότου μεταφερθεί ένα πλήρες μπλοκ. Όταν η μεταφορά αυτή ολοκληρωθεί, οι διάυλοι ελευθερώνονται για χρήση από την ΚΜΕ εκτός κι αν η περιφερειακή συσκευή οδηγήσει τη γραμμή READY του DMAC κατάλληλα ώστε να αιτηθεί μια νέα μεταφορά. Τέλος η περίπτωση του Continuous mode μοιάζει με το Burst mode όμως οι διάυλοι δεν ελευθερώνονται μετά τη μεταφορά ενός μπλοκ αλλά συνεχίζουν να ελέγχονται από τον DMAC.

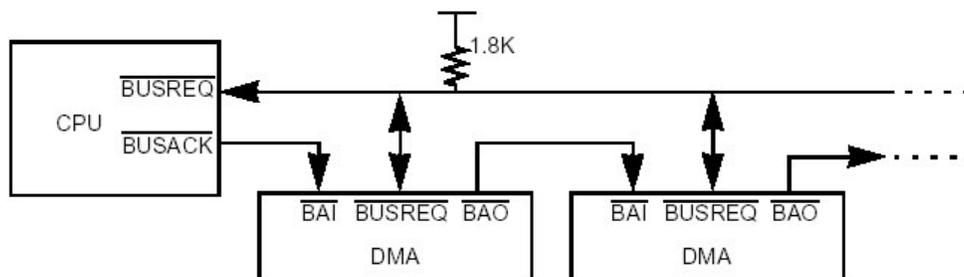
2.8.4.2 Ακροδέκτες του Z80 DMAC



Σχήμα 2-48. Ακροδέκτες του Z80 DMAC

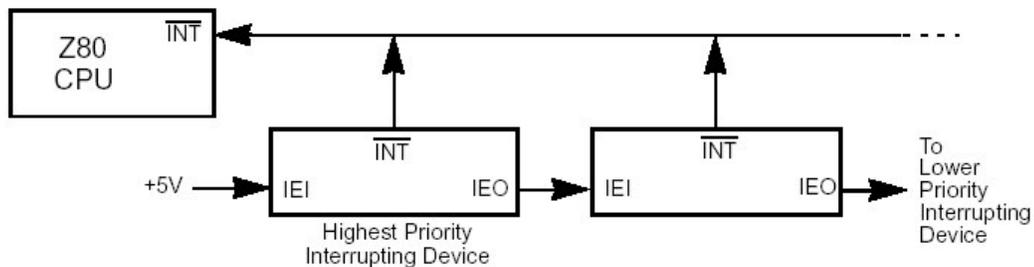
Στο Σχήμα 2-48 φαίνονται οι ακροδέκτες του ολοκληρωμένου κυκλώματος ομαδοποιημένοι κατά λειτουργία.

- CLK, 5V, GND: ρολόι, τροφοδοσία και γείωση συστήματος.
- Οι γραμμές A0-A15 και D0-D7 συνδέονται στο διάλυο διευθύνσεων και δεδομένων του συστήματος αντίστοιχα.



Σχήμα 2-49. Σύνδεση BAI~ και BAO~ σε αλυσίδα DMAC

- Η γραμμή BAI~ (Bus Acknowledge In) οδηγείται από τον Z80 για να δηλώσει στον DMAC ότι οι δίαυλοι συστήματος είναι στη διάθεσή του.
- Η γραμμή BAO~ (Bus Acknowledge Out) χρησιμοποιείται για την περίπτωση που υπάρχουν στο σύστημα περισσότεροι του ενός DMAC. Στην περίπτωση αυτή τα σήματα BAI~ και BAO~ συνδέουν σε αλυσίδα τους DMAC ορίζοντας την προτεραιότητα κάθε ενός ανάλογα με τη θέση του στην αλυσίδα (Σχήμα 2-49). Αντίθετα με την περίπτωση των διακοπών όπου μια διακοπή μεγαλύτερης προτεραιότητας έχει δικαίωμα να διακόψει μια άλλη μικρότερης εδώ μια DMA μεταφορά δεν διακόπτεται πριν ολοκληρωθεί.
- Το BUSREQ~ είναι διπλής κατεύθυνσης σήμα που είτε κάνει αίτηση προς τον Z80 να παραχωρήσει τους διαύλους πριν από μια DMA μεταφορά είτε δηλώνει στους υπόλοιπους DMAC μιας αλυσίδας ότι κάποιος άλλος έχει ήδη ζητήσει εξυπηρέτηση.
- Το CE~/WAIT~ μπορεί να χρησιμοποιηθεί είτε σαν chip enable σήμα για την επιλογή του DMAC κατά την εγγραφή ενός control ή την ανάγνωση ενός status byte, είτε για την εισαγωγή Wait cycles κατά τη διαδικασία του DMA έτσι ώστε να μπορεί να χρησιμοποιηθεί αργή μνήμη σε με DMA διαδικασία.



Σχήμα 2-50. Σύνδεση IEI και IEO σε αλυσίδα DMAC

- Τα σήματα IE~, IO~ κατ' αντιστοιχία με τα BAI~ και BAO~ συνδέουν σε αλυσίδα περισσότερα του ενός DMAC και χρησιμοποιούνται για να δηλώσουν ότι κάποιο DMAC έχει κάνει αίτηση διακοπής (Σχήμα 2-50). Οι αιτήσεις ιεραρχούνται κατά προτεραιότητα και η εξυπηρέτηση interrupt μικρής προτεραιότητας μπορεί να διακοπεί από μια μεγαλύτερης.
- Το σήμα INT~ συνδέεται στο αντίστοιχο του Z80 και χρησιμοποιείται για την αίτηση διακοπής όταν ο DMAC δεν έχει υπό τον έλεγχό του το δίαυλο συστήματος (δεν είναι δηλαδή bus master).
- Το IORQ~ είναι είσοδος όταν τον έλεγχο έχει ο Z80 και δηλώνει ότι οι γραμμές A0-A7 του διαύλου διεύθυνσεων περιέχουν έγκυρη διεύθυνση I/O χώρου (και όχι κύριας μνήμης). Στην περίπτωση αυτή το IORQ~ σε συνδυασμό με τα CE~ και RD~ σήματα μπορεί να δηλώνει ότι έχει επιλεγεί ένας configuration register του DMAC για ανάγνωση ή εγγραφή. Αν το DMAC έχει αποκτήσει τον έλεγχο των διαύλων του συστήματος τότε το IORQ~ ως έξοδος μπορεί να επιλέξει άλλες περιφερειακές συσκευές για συμμετοχή σε DMA μεταφορές.

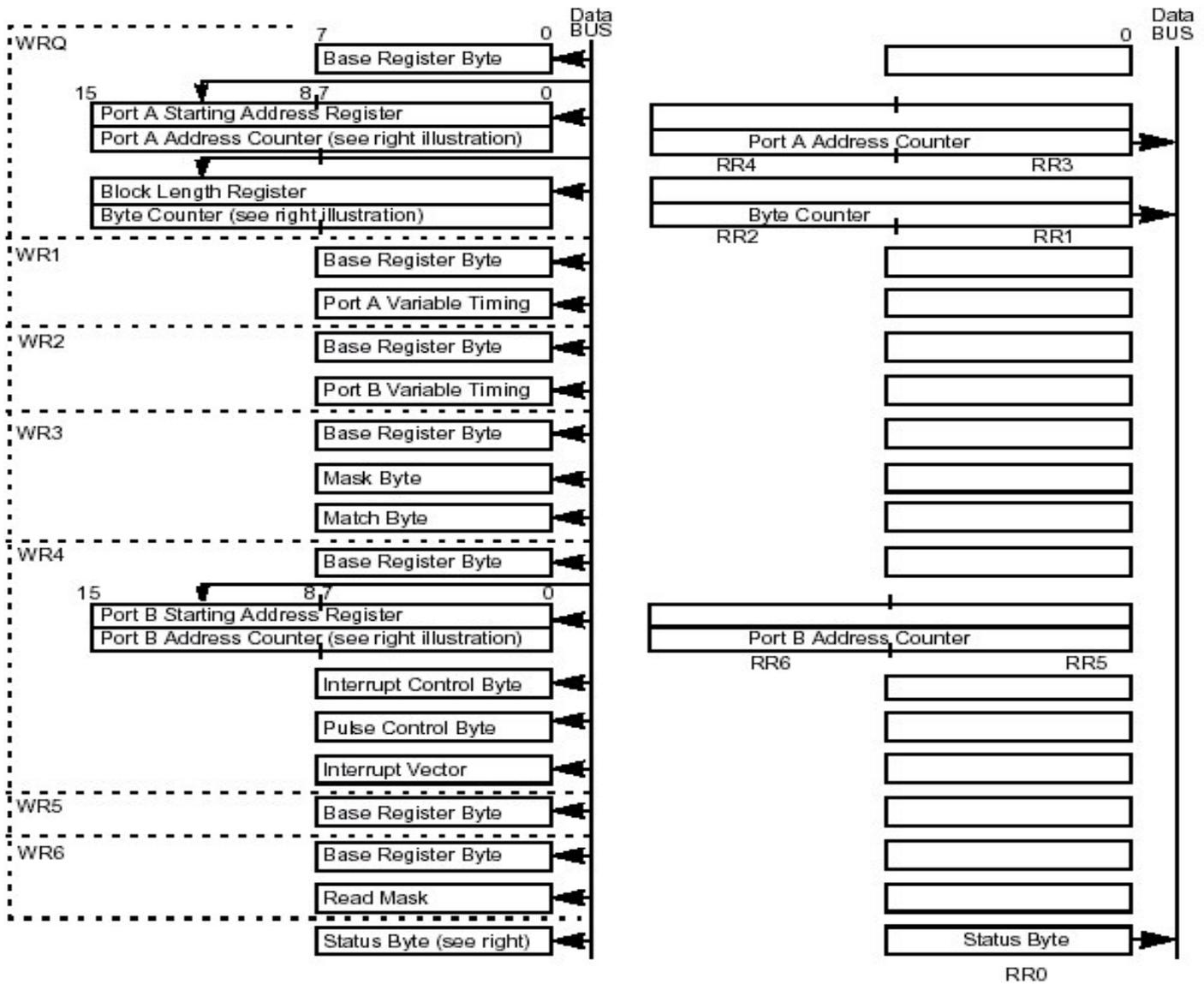
- Το M1~ σήμα μπορεί να υποδείξει ότι βρισκόμαστε στον κύκλο προσκόμισης της εντολής, ή ότι έχουμε interrupt acknowledge (σε συνδυασμό με το IORQ~), ή για εσωτερικό reset.
- Το MREQ~ είναι έξοδος και όταν το DMAC αποκτήσει τον έλεγχο δείχνει ότι έχουμε μεταφορά δεδομένων από ή προς την κύρια μνήμη.
- Το σήμα RD~ όταν είναι είσοδος οδηγείται από τον Z80 για την ανάγνωση της τιμής κάποιου configuration register του DMAC. Όταν το DMAC έχει τον έλεγχο το σήμα RD~ γίνεται έξοδος και χρησιμοποιείται κατά την ανάγνωση δεδομένων από κάποιο άλλο περιφερειακό ή τη μνήμη.
- Το RDY~ οδηγείται από ένα περιφερειακό συνδεδεμένο με το DMAC και χρησιμοποιείται να ξεκινήσει η μεταφορά δεδομένων. Για παράδειγμα, αν το περιφερειακό θέλει να μεταφέρει δεδομένα στην κύρια μνήμη, τότε ενεργοποιεί το RDY~ για να ειδοποιήσει τον DMAC να αναλάβει τον έλεγχο (να γίνει bus master) και να ξεκινήσει η μεταφορά.
- Το WR~ είναι ανάλογο του RD~ αλλά για εγγραφή.

2.8.4.3 Προγραμματισμός του DMAC

Ο DMAC αποκωδικοποιείται σαν μια μόνο διεύθυνση για ανάγνωση ή εγγραφή στον I/O χώρο μνήμης παρότι υπάρχει ένα πλήθος εσωτερικών καταχωρητών ανάγνωσης ή εγγραφής που αυτός διαθέτει. Ο διαχωρισμός του καταχωρητή στον οποίο απευθύνεται μια προσπέλαση του Z80 γίνεται από κάποια συγκεκριμένα bits της ίδιας της τιμής την οποία γράφει ο Z80 στη μοναδική διεύθυνση του DMAC καθώς και από τη σειρά με την οποία προσπελούνται οι καταχωρητές

Για τον προγραμματισμό του DMAC υπάρχουν 7 εγγράψιμοι **βασικοί** καταχωρητές (base registers WR0-WR6) και μια σειρά **συνδεδεμένων** (associated) καταχωρητών. Ένας βασικός μπορεί να έχει έναν αριθμό συνδεδεμένων καταχωρητών (base register group) οι οποίοι προσπελούνται με συγκεκριμένη ακολουθία τιμών και όχι απ' ευθείας με αποκωδικοποίηση του address bus. Με άλλα λόγια κάθε ένας από τους βασικούς ή συνδεδεμένους καταχωρητές δεν έχει δική του διεύθυνση στον χώρο μνήμης ή I/O. Κάποια bit fields ενός καταχωρητή ρυθμίζουν αυτόνομες λειτουργίες και κάποια άλλα είναι δείκτες (pointers) για την προσπέλαση άλλων καταχωρητών σε επόμενη προσπέλαση του DMAC.

Εκτός από τους εγγράψιμους υπάρχουν και άλλοι 7 καταχωρητές κατάστασης (status registers RR0-RR6) οι οποίοι μπορούν να προσπελαστούν διαβάζοντας από τη θέση μνήμης που αποκωδικοποιείται το DMAC. Το ποιος από τους RR1-RR6 θα προσπελαστεί με μια ανάγνωση καθορίζεται από τις τιμές που έχουν γραφτεί στο WR6 group προηγουμένως. Ο Status Byte (RR0) προσπελώνεται με διαφορετικό τρόπο από τους υπόλοιπους καταχωρητές ανάγνωσης όπως θα δούμε παρακάτω. Η δομή των καταχωρητών που μόλις περιγράφηκε απεικονίζεται στο Σχήμα 2-51.

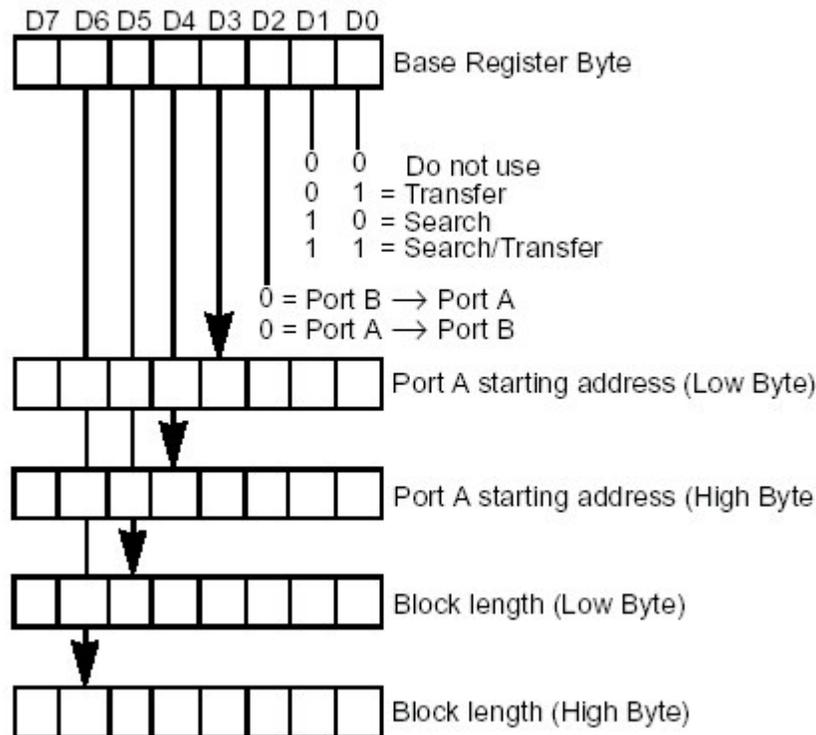


Σχήμα 2-51. Η δομή των καταχωρητών ανάγνωσης και εγγραφής

Πριν περιγράψουμε ξεχωριστά πως μπορεί να προσπελαστούν οι καταχωρητές ενός register group και τι πεδία διαθέτει κάθε ένας από αυτούς είναι χρήσιμο να αναφέρουμε τις καταστάσεις στις οποίες μπορεί να βρεθεί ο DMAC. Κατά την εκκίνηση του συστήματος ο DMAC βρίσκεται σε Disabled Mode όπου μπορούν να προσπελαστούν οι καταχωρητές διαμόρφωσης και κατάστασης για να αρχικοποιηθεί το σύστημα. Όταν ο DMAC είναι Enabled (εκτελώντας μια «ENABLE DMA ενέργεια-εντολή») υπάρχει η περίπτωση να είναι ενεργός (active) ή ανενεργός (inactive ή stopped). Στην δεύτερη περίπτωση μπορούν και πάλι να προσπελαστούν οι καταχωρητές αρκεί να έχει προηγηθεί μια «DISABLE DMA εντολή». Σε ανενεργή κατάσταση φτάνει όταν έχει ολοκληρωθεί για παράδειγμα μία μεταφορά ή μια αναζήτηση. Ενεργή κατάσταση έχουμε όταν εκτελείται μια μεταφορά ή μια αναζήτηση και μια τέτοια λειτουργία μπορεί να ανασταλεί προσωρινά (suspended) αν

βρισκόμαστε σε Continuous mode μεταφορά και απενεργοποιηθεί η γραμμή RDY. Κατά την Ενεργή κατάσταση δεν είναι δυνατή η προσπέλαση καταχωρητών.

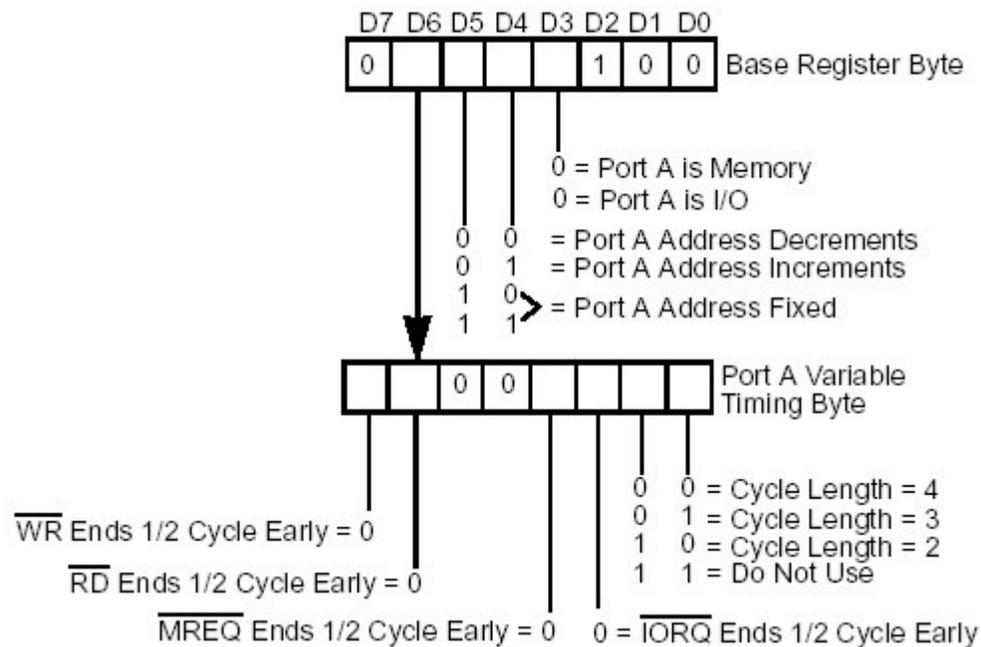
2.8.4.3.1 WR0 group registers



Σχήμα 2-52. Προσπέλαση WR0 group registers

Για να προσπελάσει τον WR0 ο Z80 πρέπει να γράψει στη διεύθυνση που αποκωδικοποιείται ο DMAC μια τιμή που στο bit 7 έχει 0 και στα bit 1 και 0 μια τιμή διάφορη του 00. Όπως φαίνεται στο Σχήμα 2-52 με τα bit 1 και 0 μπορεί να καθορισθεί η επιθυμητή ενέργεια που πρέπει να ενεργοποιηθεί (μεταφορά, αναζήτηση ή συνδυασμός των δύο). Το bit 2 καθορίζει την κατεύθυνση της ενέργειας (από ποια θύρα-port θα ληφθούν και σε ποιά θα κατευθυνθούν τα δεδομένα). Όταν κάποια από τα bit 3-6 γίνουν 1 αυτό σημαίνει ότι με επόμενη εγγραφή θα προσπελαστεί κάποιος από τους συνδεδεμένους καταχωρητές. Πχ, αν σε εγγραφή του WR0 γίνουν 1 τα bit 4 και 6 στις επόμενες 2 εγγραφές θα προσπελαστούν το port A starting address (low) και το block length (high) με αυτή τη σειρά. Οι associated registers του WR0 όπως είναι φανερό και από τα ονόματα τους καθορίζουν την 16-bit διεύθυνση της θύρας A (port A) και το 16-bit πλήθος των δεδομένων που θα μεταφερθούν.

2.8.4.3.2 WR1 και WR2 group registers



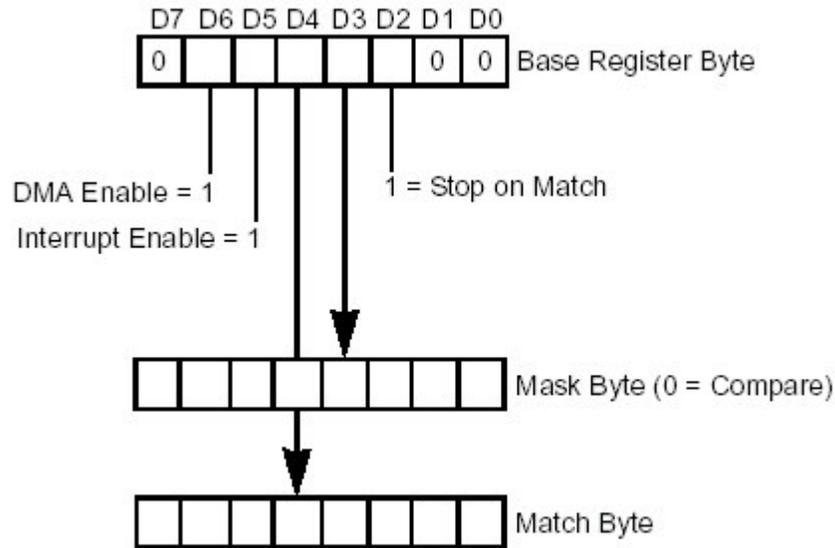
Σχήμα 2-53. WR1 group registers

Ο WR1 διευθύνσιδοτεείται όταν τα bit 7, 2, 1, 0 της τιμής που γράφει ο Z80 στον DMAC είναι 0,1,0,0 αντίστοιχα. Ο WR1 έχει έναν συνδεδεμένο καταχωρητή που προσπελαύνεται αφού γίνει 1 το bit 6. Από τα υπόλοιπα πεδία του WR1 μπορεί να καθοριστεί αν ο διάυλος A είναι μνήμη ή I/O και ο τρόπος με τον οποίο μεταβάλλεται η διεύθυνση του κατά την μεταφορά δεδομένων. Ο συνδεδεμένος καταχωρητής έχει πεδία που καθορίζουν το χρονοισμό των σημάτων \overline{WR} ~, \overline{RD} ~, \overline{MREQ} ~, \overline{IORQ} ~

Το WR2 group έχει ακριβώς την ίδια δομή με το WR1 με τη διαφορά ότι αναφέρεται στην θύρα B. Ο βασικός καταχωρητής καθορίζεται όταν τα bit 7, 2, 1, 0 της τιμής που γράφει ο Z80 είναι όλα 0. Στην περίπτωση ενεργειών που δεν χρειάζεται μεταφορά όπως είναι η αναζήτηση, οι καταχωρητές της θύρας B δεν προγραμματίζονται.

2.8.4.3.3 WR3 group registers

Οι καταχωρητές που σχετίζονται με τον WR3 καθορίζουν τις συνθήκες τερματισμού μιας λειτουργίας αναζήτησης (Σχήμα 2-54)

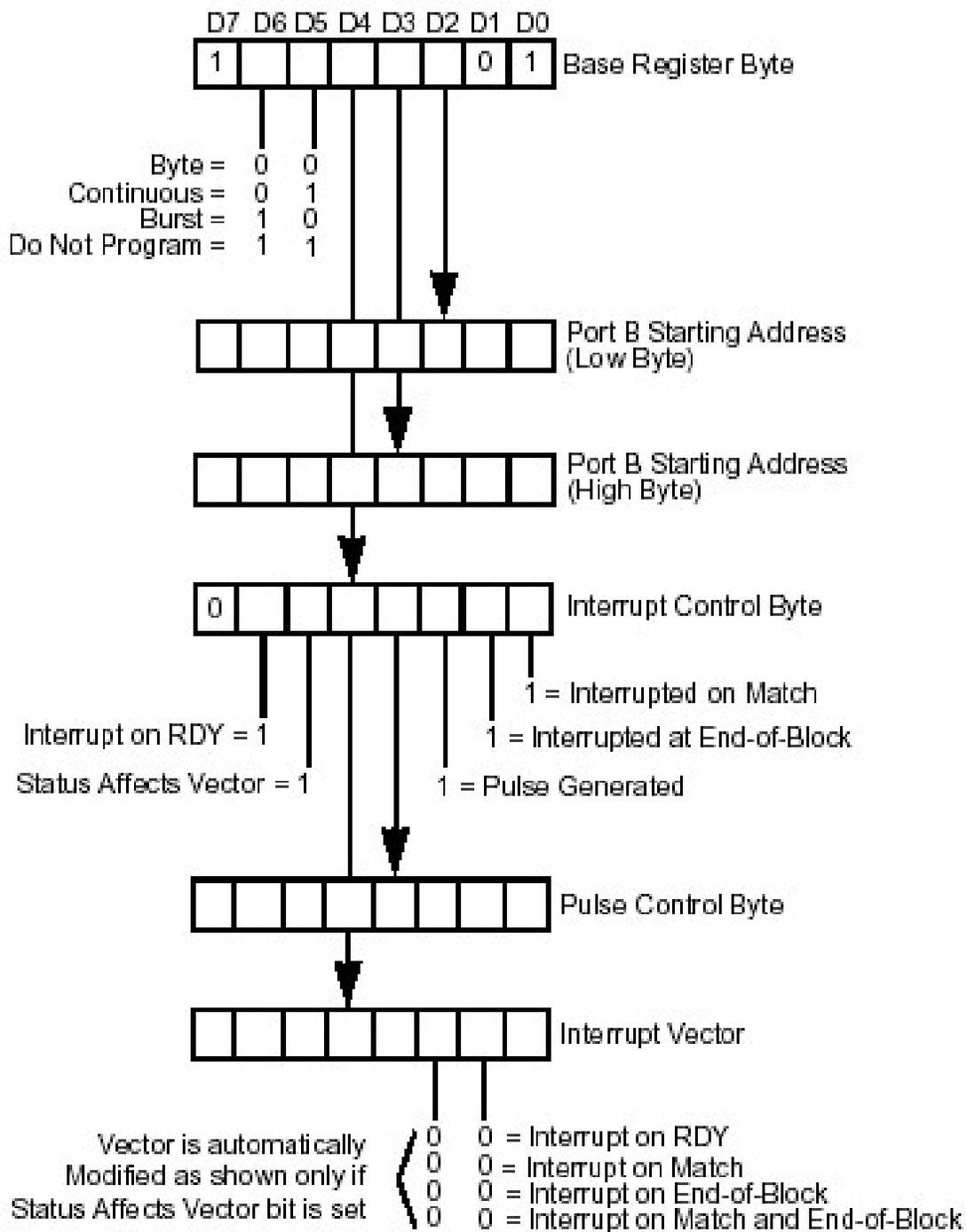


Σχήμα 2-54. WR3 group registers

Ο βασικός καταχωρητής WR3 προσπελαύνεται όταν στα bit 7, 1 και 0 υπάρχει 0. Αν στο bit 2 υπάρχει η τιμή 1 τότε αν είμαστε σε διαδικασία αναζήτησης (έχει ήδη καθοριστεί από το WR0 register group) αυτή θα τερματιστεί αν βρεθεί η τιμή που ορίζεται από τους δύο συνδεδεμένους καταχωρητές (mask/match byte). Συγκεκριμένα ο mask καθορίζει ποια πεδία του match θα συγκριθούν. Αν ο mask έχει δυαδική τιμή 11110000 τότε αναζητείται μια τιμή που το πιο σημαντικό nibble της πρέπει να ταιριάζει με το πιο σημαντικό nibble της τιμής του match byte. Τα DMA, Interrupt Enable bits επιτελούν μια ENABLE Interrupt ή μια ENABLE DMA εντολή αντίστοιχη με εκείνη που προβλέπεται στο WR6.

2.8.4.3.4 WR4 group register

Με τους καταχωρητές του group αυτού ορίζονται παράμετροι για τον τρόπο DMA μεταφοράς δεδομένων. Ο βασικός καταχωρητής διευθυνσιοδοτείται όταν τα bit 7, 1, 0 της τιμής που γράφει ο Z80 είναι 1, 0, 1 αντίστοιχα. Τα bit 4, 3, 2 χρησιμοποιούνται για την προσπέλαση τριών ακόμα συνδεδεμένων καταχωρητών. Υπάρχουν εντούτοις επιπλέον 2 συνδεδεμένοι οι οποίοι δεν δεικτοδοτούνται απ' ευθείας από τον βασικό αλλά έμμεσα από το Interrupt Control Byte που είναι και το ίδιο ένας συνδεδεμένος καταχωρητής (Σχήμα 2-55).



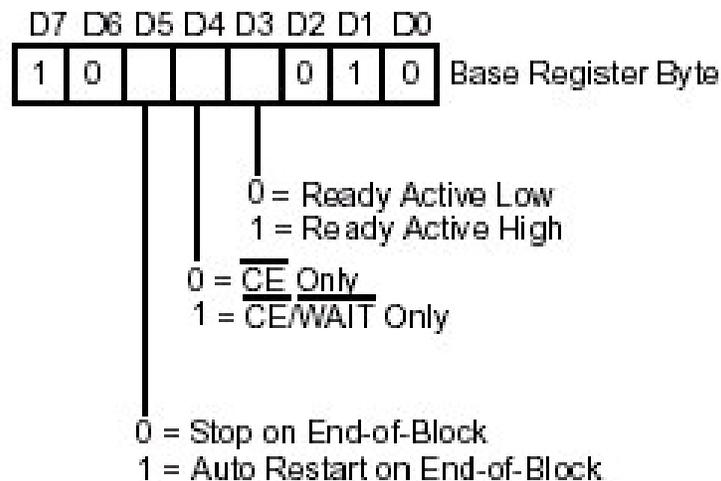
Σχήμα 2-55. WR4 group registers

Τα υπόλοιπα πεδία που υπάρχουν στους καταχωρητές του group αυτού καθορίζουν το αν θα έχουμε Continuous, Burst ή Byte mode, την αρχική διεύθυνση της θύρας B, την αιτία που μπορεί να προκαλέσει διακοπή (ολοκλήρωση μεταφοράς μπλοκ, εύρεση αναζητούμενης τιμής, ενεργοποίηση σήματος READY, δημιουργία

παλμού). Η δημιουργία παλμών στο σήμα INT~ είναι μια δυνατότητα που ενεργοποιείται όταν τα bits 2, 3 του Interrupt Control καταχωρητή τεθούν στην τιμή 1. Κατόπιν προσπελαύνεται ο Pulse Control Byte για να του δοθεί μια τιμή η οποία στη συνέχεια συγκρίνεται με το λιγότερο σημαντικό byte του Byte Counter καταχωρητή. Όταν οι δύο αυτές τιμές συμπέσουν δημιουργείται ένας παλμός. Ουσιαστικά θα δημιουργείται ένας παλμός για κάθε 256 bytes που μεταφέρονται σε ένα μπλοκ με μέγεθος μεγαλύτερο του 256.

Ο καταχωρητής Interrupt Vector χρησιμοποιείται για τον σχηματισμό της διεύθυνσης που θα σταλεί στον Z80 σε απάντηση ενός interrupt acknowledge για να επιλεγεί η κατάλληλη ρουτίνα εξυπηρέτησης. Η τιμή της διεύθυνσης αυτής μπορεί να τροποποιείται αυτόματα (μέσω των bit 1 και 2 του Interrupt Vector) ανάλογα με την αιτία της διακοπής όταν έχει αυτό επιτραπεί μέσω του Status Affects Vector bit του Interrupt Control Byte.

2.8.4.3.5 WR5 group register

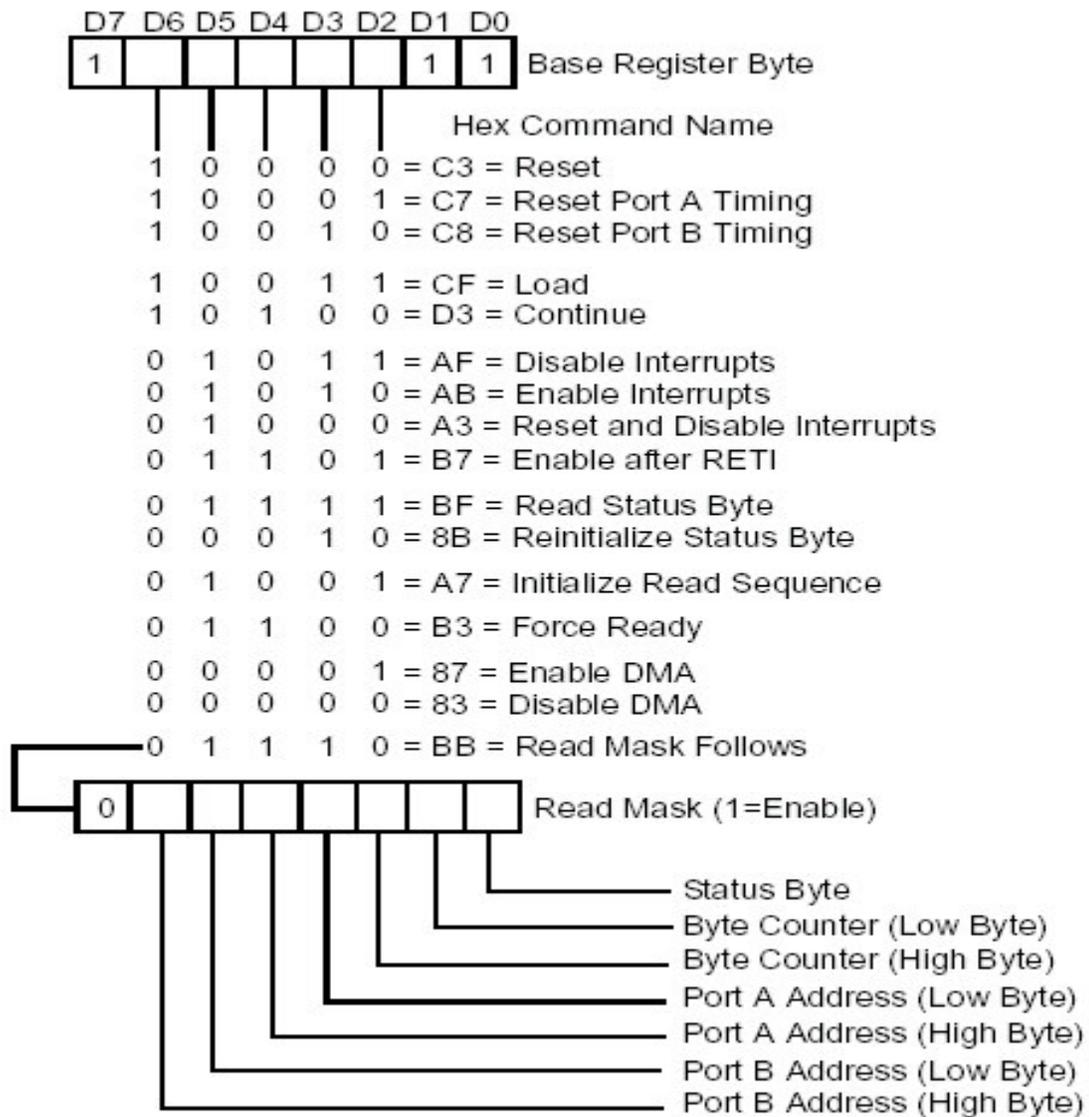


Σχήμα 2-56. WR5 group register

Στο WR5 υπάρχει μόνο ο βασικός καταχωρητής που διευθυνσιοδοτείται όταν τα bits 7,6,2,1,0 γίνουν αντίστοιχα 1,0,0,1,0. Μέσω του καταχωρητή αυτού μπορεί να καθοριστεί αν το σήμα READY θα είναι active low ή high και αν ο ακροδέκτης CE~/WAIT~ θα λειτουργεί μόνο ως CE~ ή με πολυπλεγμένες και τις δύο λειτουργίες του. Η σπουδαιότερη όμως ενέργεια που μπορεί να καθοριστεί είναι αν θα ελευθερωθούν οι δίαυλοι συστήματος με την λήξη της τρέχουσας μεταφοράς ή αν θα ακολουθήσει επόμενη διαδικασία DMA.

2.8.4.3.6 WR6 group registers

Το group αυτό περιλαμβάνει μια σειρά από εντολές που αντιστοιχούν σε κάποια τιμή του βασικού καταχωρητή. Ο ίδιος ο βασικός καταχωρητής διευθύνσιοδοτείται όταν τα bit 7, 1, 0 είναι όλα 1.



Σχήμα 2-57. WR6 group registers

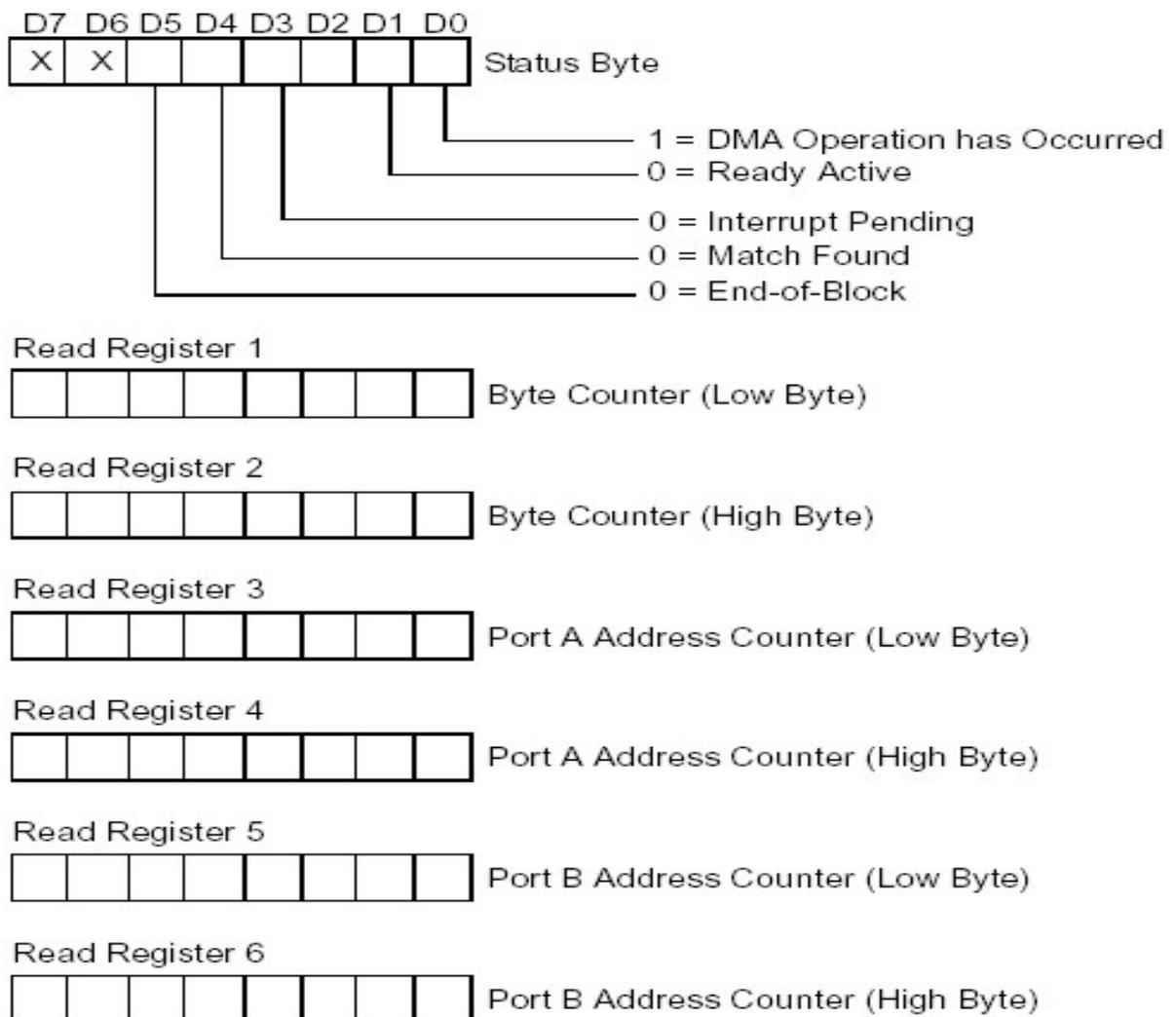
Ακολουθεί μια σύντομη περιγραφή της κάθε μιας εντολής

- Reset, Reset port A, Reset port B timing: Γενική αρχικοποίηση ή αρχικοποίηση των χρόνων που μεταβλήθηκαν με τα WR1/WR2.
- Load: Φόρτωση των νέων τιμών αρχικής διεύθυνσης μπλοκ δεδομένων και πλήθους δεδομένων για να ξεκινήσει μια νέα DMA μεταφορά από διαφορετικές διευθύνσεις.
- Continue: Μηδενίζει τον Byte Counter αλλά αφήνει τις τρέχουσες τιμές των μετρητών ανέπαφες ώστε να συνεχιστεί η μεταφορά δεδομένων από την ίδια θέση αλλά για άλλο ένα συνεχόμενο μπλοκ ίδιου μεγέθους. Χρησιμοποιείται όταν χρειάζεται μεταφορά διαδοχικών μπλοκ οπότε στην περίπτωση αυτή παραλείπεται η χρήση της Load. Η Continue θα πρέπει να ενεργοποιείται μέσα από τη ρουτίνα εξυπηρέτησης του τέλους μεταφοράς ενός μπλοκ.
- Disable Interrupts και Enable Interrupts κάνουν ότι και τα αντίστοιχα πεδία στον WR4.
- Reset and Disable Interrupts. Είναι χρήσιμες για διασύνδεση με συστήματα που έχουν μ/ε 8080A ή 8085.
- Enable after RETI: Χρησιμοποιείται πάντα σε ρουτίνες χειρισμού διακοπών για την επανάκτηση των διαύλων από το DMAC σε Z80 συστήματα. Πρέπει να ακολουθείται από μια Enable DMA εντολή πριν το κλείσιμο της ρουτίνας με RETI.
- Read Status Byte: Η επόμενη ανάγνωση μετά από μια τέτοια εντολή διαβάζει την τιμή του Status Byte.
- Reinitialize Status Byte: Αρχικοποίηση των bit 4, 5 του Status Byte (Match not found, not end of block).
- Read Mask Follows: Μετά από αυτή την εντολή γράφεται μια τιμή στον διασυνδεδεμένο καταχωρητή η οποία καθορίζει τη σειρά με την οποία θα πρέπει να προσπελαστούν καταχωρητές ανάγνωσης με επόμενα διαδοχικά διαβάσματα. Πχ, αν δοθεί στον Read Mask η τιμή 01100110 στα επόμενα τρία διαβάσματα θα αναγνωσθούν οι byte counter low, high, port B address low, high με αυτή την σειρά.
- Initialize Read Sequence. Αν δεν θέλουμε να ολοκληρώσουμε την ανάγνωση που ορίστηκε από την Read Mask Follows, μπορούμε να χρησιμοποιήσουμε αυτή την εντολή.
- Force Ready. Προσομοίωση της ενεργοποίησης του σήματος RDY για περίπτωση μεταφοράς από μνήμη σε μνήμη όπου δεν έχει νόημα ένα τέτοιο σήμα.
- Enable DMA, Disable DMA: Επιτρέπουν και απαγορεύουν αντίστοιχα στο DMA να αποκτήσει τον έλεγχο των διαύλων συστήματος.

Όλες οι εντολές που απαριθμούνται στο Σχήμα 2-57 φέρνουν το DMAC σε DISABLE STATE, εκτός από την ENABLE DMA, κατά συνέπεια η εντολή αυτή θα πρέπει να εκτελείται τελευταία.

2.8.4.3.7 Καταχωρητές Ανάγνωσης

Οι καταχωρητές Ανάγνωσης παρουσιάζονται στο Σχήμα 2-58. Ο Status Byte ο οποίος όπως είδαμε διαβάζεται με διαφορετικό τρόπο από τους υπόλοιπους περιλαμβάνει πεδία που δηλώνουν αν έχει ολοκληρωθεί μια DMA λειτουργία, αν το σήμα READY είναι ενεργό τη στιγμή εκείνη, αν υπάρχει διακοπή που δεν έχει εξυπηρετηθεί, αν βρέθηκε κάποια τιμή που αναζητήθηκε και αν έχει φτάσει το τέλος της μεταφοράς ενός μπλοκ. Οι υπόλοιποι καταχωρητές ανάγνωσης έχουν ενιαία πεδία και από αυτούς ο μ/ε μπορεί να διαβάσει τις τρέχουσες 16-bit τιμές του Byte Counter, Address Counter για τις θύρες A και B. Θα πρέπει να τονιστεί ότι οι παραπάνω 16-bit καταχωρητές **δεν** έχουν τις αρχικές διευθύνσεις και το μέγεθος του μπλοκ που θα μεταφερθεί όπως αυτές ορίστηκαν στους καταχωρητές των WR0 και WR4 groups αλλά τις τρέχουσες τιμές των διευθύνσεων που προσπελούνται και το πλήθος των δεδομένων που έχουν ήδη μεταφερθεί.



Σχήμα 2-58. Οι καταχωρητές ανάγνωσης

2.8.4.3.8 Αρχικοποίηση Συστήματος

Ας δούμε τώρα τη σειρά με την οποία πρέπει να προγραμματιστούν οι καταχωρητές εγγραφής που αναφέραμε για την αρχικοποίηση του συστήματος. Ο Πίνακας 2-1, αναφέρει τα βήματα της αρχικοποίησης αλλά όχι τις τιμές που θα πάρουν οι καταχωρητές αφού αυτές εξαρτώνται από τη συγκεκριμένη εφαρμογή. Όπου υπάρχει η λέξη *εντολή* αυτή αναφέρεται σε εγγραφή στον WR6. Πολλά από τα βήματα απαιτούν την προσπέλαση περισσότερων του ενός καταχωρητών πχ, αν ένα group έχει 3 καταχωρητές, απαιτούνται ισάριθμες εγγραφές από τον Z80 στον DMAC. Επίσης για ορισμένες λειτουργίες κάποια από τα βήματα του Πίνακα 1 είναι περιττά όπως πχ, όταν αρχικοποιείται αναζήτηση τιμής όπου δεν χρειάζεται να αρχικοποιηθούν οι τιμές της θύρας B που δεν συμμετέχει σε μια τέτοια λειτουργία.

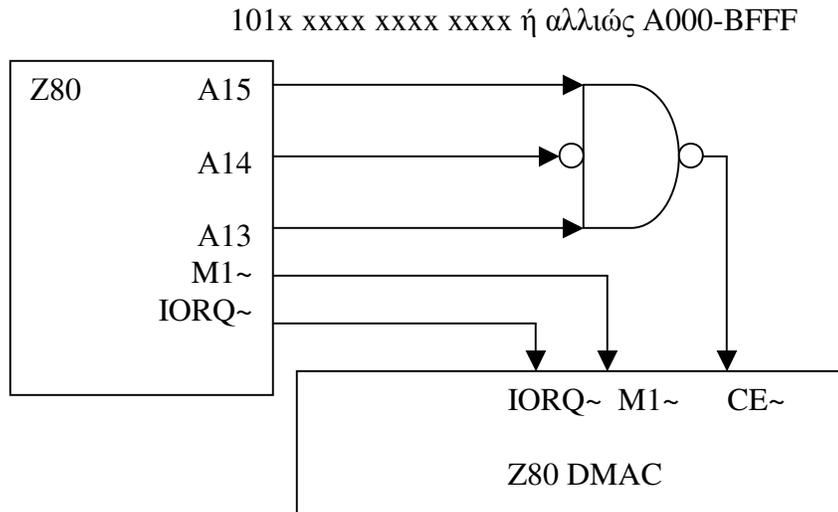
Disable DMA εντολή
Reset εντολές
WR0 group
WR1 group
WR2 group
WR3 group
WR4 group
WR5 group
Reset Port A timing
Reset Port B timing
Load εντολή
Reinitialize Status Byte εντολή
Read Mask Follows εντολή
Read Mask control byte
Initiate Read Sequence εντολή
Force Ready εντολή
Enable Interrupts εντολή
Enable DMA εντολή

Πίνακας 2-1. Σειρά Λειτουργιών Αρχικοποίησης

2.8.4.4 Παράδειγμα διασύνδεσης DMAC με Z80

Τα σήματα του DMAC που εξετάσαμε είναι συμβατά με εκείνα του Z80 έτσι ώστε να ελαχιστοποιείται η χρήση εξωτερικής λογικής για προσαρμογή σημάτων. Η αποκωδικοποίηση του DMAC σε μια διεύθυνση στην κύρια μνήμη ή στο χώρο I/O μπορεί να γίνει με πολλούς διαφορετικούς τρόπους ανάλογα με το αν υπάρχουν ένας ή περισσότεροι DMAC και με το αν αυτοί θα επιλέγονται σε μία μοναδική διεύθυνση (αποκωδικοποίηση όλων των γραμμών του διαύλου διευθύνσεων) ή σε ένα εύρος διευθύνσεων (συμμετοχή λιγότερων γραμμών στην αποκωδικοποίηση). Η

αποκωδικοποίηση περισσότερων του ενός DMAC μπορεί να γίνει και με τη χρήση κυκλωμάτων όπως πύλες NAND, μνήμη PROM κ.α. Ένα παράδειγμα αποκωδικοποίησης με χρήση πύλης NAND παρουσιάζεται στο Σχήμα 2-59. Το DMAC επιλέγεται όταν ο Z80 δώσει μια διεύθυνση που έχει στο A15, A14 και A13 τις τιμές 1, 0 και 1 αντίστοιχα. Κατά συνέπεια οι διευθύνσεις που επιλέγουν το DMAC είναι οι

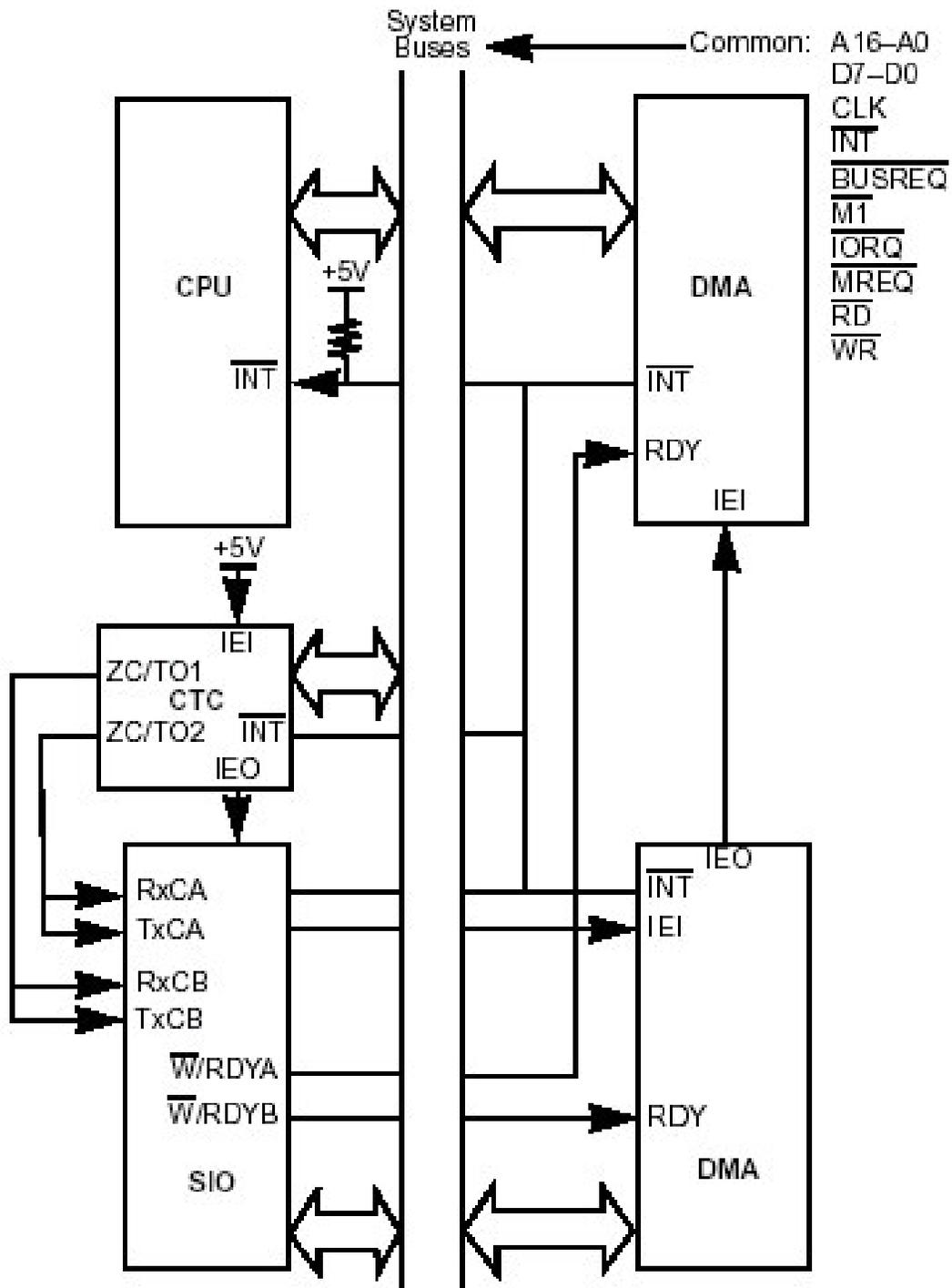


Σχήμα 2-59. Αποκωδικοποίηση με τη χρήση πύλης NAND

Ένα σύστημα με 2 DMAC που μεταφέρουν δεδομένα μεταξύ των 2 καναλιών της σειριακής θύρας Z80 SIO και της μνήμης (της οποίας η διασύνδεση παραλείπεται για λόγους απλότητας) παρουσιάζεται στο Σχήμα 2-59. Στις κοινές γραμμές που συνδέουν τα περιφερειακά με τον Z80 περιλαμβάνονται όλοι οι δίαυλοι και οι γραμμές ελέγχου που απαριθμούνται επάνω δεξιά στο Σχήμα 2-60. Οι γραμμές RDY των δύο καναλιών του SIO οδηγούν τις αντίστοιχες των 2 DMAC ώστε όταν φτάνει κάποιος καινούργιος χαρακτήρας, αυτός να μεταφέρεται στην κύρια μνήμη χωρίς τη μεσολάβηση του Z80. Αν δεν υπήρχε DMAC, ο SIO θα έπρεπε να δημιουργήσει μια διακοπή στον Z80 και εκείνος με τη σειρά του θα έπρεπε να διαβάσει από το SIO το χαρακτήρα που έφτασε και να τον αποθηκεύσει στην κατάλληλη θέση μνήμης. Ο χρόνος εξυπηρέτησης μιας τέτοιας διακοπής εν συγκρίσει με τον χρόνο που απαιτεί το DMA σαν ποσοστό της δραστηριότητας του Z80 φαίνεται στον Πίνακα 2-2.

	Z80 (2.5 MHz)	Z80A (4 MHz)
DMA sequential transfer	0.065%	0.041%
DMA sequential transfer/search		
Interrupt-driven transfer	0.340%	0.213%

Πίνακας 2-2. Σύγκριση χρόνων



Σχήμα 2-60. Μεταφορά δεδομένων σειριακής στη μνήμη μέσω DMA

2.9 Παράδειγμα ολοκληρωμένου μικροϋπολογιστικού συστήματος με Z80

Στα Σχήματα 2-61 και 2-62 παρουσιάζονται παραδείγματα σύνδεσης του μ/ε Z80 με μνήμες και περιφερειακά όπως θύρες I/O και Σειριακή θύρα. Τα δύο κυκλώματα θα μπορούσαν να συνδυαστούν σε ένα αλλά αυτό δεν έγινε για να μπορέσει να φανεί όσο το δυνατόν περισσότερη λεπτομέρεια από τα κυκλώματα στις εκτυπωμένες σελίδες.

Ας ξεκινήσουμε από το Σχήμα 2-61 όπου η καρδιά του κυκλώματος είναι φυσικά ο μ/ε Z80. Ο επεξεργαστής αυτός χρειάζεται μία και μόνη τροφοδοσία 5V στον ακροδέκτη 11 και τη γείωση στον ακροδέκτη 29. Θεωρούμε ότι έχουμε διαθέσιμη μια τέτοια τάση από κάποιο έτοιμο τροφοδοτικό. Το ρολόι του Z80 (CLK) είναι συνδεδεμένο με έναν εξωτερικό ταλαντωτή (πχ, resonator) τον OG1 συχνότητας 4MHZ. Τέτοιοι ταλαντωτές είναι διαθέσιμοι σε εξαρτήματα των 3 ακροδεκτών εκ των οποίων οι δύο χρειάζονται για την τροφοδοσία και ο τρίτος σαν έξοδος τετραγωνικού παλμού στην ονομαστική συχνότητα του ταλαντωτή.

Η είσοδος RESET συνδέεται με ένα απλό κύκλωμα RC (R4, C1) που κατά την εκκίνηση του συστήματος φροντίζει να παραμείνει σε χαμηλό δυναμικό η είσοδος RESET για κάποιο χρονικό διάστημα και με τον τρόπο αυτό να γίνει ομαλά η αρχικοποίηση του συστήματος. Ο διακόπτης S2 μπορεί να προκαλέσει εκφόρτιση του C1 σε οποιαδήποτε χρονική στιγμή της λειτουργίας του συστήματος και ουσιαστικά έναν παλμό RESET με ευθύνη του χρήστη. Οι δύο Schmitt Trigger αναστροφείς που συνδέονται πριν τον ακροδέκτη RESET χρησιμοποιούνται για να καλύψουν την περίπτωση που η άνοδος της τάσης στα άκρα του C1 δεν είναι μονότονη πχ, αν ο S2 ανοιγοκλείσει κάποιες φορές πριν σταθεροποιηθεί. Περιφερειακά που χρειάζονται θετικό παλμό RESET θα μπορούσαν να τροφοδοτηθούν από την έξοδο του πρώτου αντιστροφέα (IC6A).

Υπάρχουν δύο ολοκληρωμένα κυκλώματα που αποτελούν τη μνήμη του συστήματος. Το ένα είναι μια RAM τύπου 55257 η οποία μπορεί να αποθηκεύσει $2^{15}=32768$ bytes. Ο διάυλος δεδομένων του Z80 συνδέεται κατευθείαν στους αντίστοιχους ακροδέκτες του 55257, το ίδιο και οι γραμμές A0-A14 του διαύλου διευθύνσεων. Η γραμμή A15 συνδέεται μέσω ενός αντιστροφέα στην γραμμή CE~ του 55257 αποκωδικοποιώντας το ολοκληρωμένο αυτό στις θέσεις μνήμης 8000-FFFF όπου το A15 είναι 1.

Το άλλο ολοκληρωμένο κύκλωμα είναι μια EPROM 27C512 η οποία είναι έχει $2^{16}=65536$ θέσεις μνήμης του ενός byte. Όμως αυτό το πλήθος είναι το μέγιστο που μπορεί να καλύψει ο χώρος μνήμης του Z80 δεδομένου ότι ο διάυλος διευθύνσεων του είναι 16-bit. Τις μισές θέσεις τις έχει καταλάβει ήδη η RAM επομένως το πολύ 32768 θέσεις μπορεί να χρησιμοποιηθούν από EPROM. Οι υπόλοιπες θέσεις που προσφέρει το 27C512 θα μείνουν αχρησιμοποίητες. Για το λόγο αυτό το A15 του 27C512 συνδέεται σε σταθερή τάση πχ, στην γείωση για να χρησιμοποιηθούν οι χαμηλότερες 32768 θέσεις. Οι A0-A14 συνδέονται με τις αντίστοιχες γραμμές του Z80 ενώ το A15 του Z80 οδηγεί κατευθείαν το CE~ του 27C512 ώστε να αποκωδικοποιηθεί στις θέσεις μνήμης 0000-7FFF όπου A15=0.

Τόσο το \overline{CE} της RAM όσο και εκείνο της EPROM οδηγούνται μέσω πυλών OR στις οποίες συνδέεται το σήμα \overline{MREQ} του Z80. Το σήμα αυτό όταν ενεργοποιηθεί (=0) δηλώνει ότι προσπελάσσεται χώρος Κύριας Μνήμης σε αντιδιαστολή με το \overline{IORQ} που δηλώνει προσπέλαση σε χώρο I/O. Με το να λαμβάνεται υπόψη το \overline{MREQ} αποκλείεται η περίπτωση ο Z80 να προσπαθεί να προσπελάσει I/O θέση και η μνήμη να ενεργοποιείται εκλαμβάνοντας τη διεύθυνση στο δίαυλο διευθύνσεων ως αναφορά στο δικό της χώρο.

Το σήμα \overline{OE} και των δύο ολοκληρωμένων συνδέεται κατευθείαν στο \overline{RD} του Z80 γιατί θα πρέπει να ενεργοποιείται όταν ο Z80 προσπαθεί να διαβάσει από τη μνήμη. Αντίθετα το $\overline{R/W}$ της RAM συνδέεται με το \overline{WR} του Z80 αφού πρέπει να γίνει 0 μόνο όταν ο Z80 πραγματοποιεί εγγραφή. Αντίστοιχο σήμα δεν υπάρχει στην EPROM αφού αυτή η μνήμη χρησιμοποιείται μόνο για ανάγνωση στην κανονική λειτουργία του συστήματος. Το σήμα \overline{RFSH} του Z80 δεν χρησιμοποιείται αφού δεν έχει περιληφθεί στο κύκλωμα δυναμική μνήμη (η 55257 είναι στατική).

Εξετάζοντας το κύκλωμα του Σχήματος 2-62, η οδήγηση του RESET, CLK και της τροφοδοσίας είναι ίδια με εκείνη του Σχήματος 2-61. Τα περιφερειακά SIO και PIO συνδέονται στον χώρο I/O του Z80 δεδομένου ότι το σήμα \overline{IORQ} αυτών συνδέεται με το \overline{IORQ} του Z80. Ο χώρος I/O είναι 256 θέσεων (00-FF). Κάθε ένα από το δύο περιφερειακά καταλαμβάνει 4 θέσεις μνήμης στο χώρο αυτό, οι οποίες ορίζονται από τα σήματα $\overline{CONTSEL}$, $\overline{PORTSEL}$ (συνώνυμα των C/D και B/A). Για το λόγο αυτό τα συγκεκριμένα σήματα συνδέονται στα A1 και A0 του διαύλου διευθύνσεων. Το \overline{CE} των δύο αυτών ολοκληρωμένων οδηγείται από τη γραμμή A7. Οι γραμμές A8-A15 δεν συμμετέχουν στη διεύθυνση I/O χώρου. Οι γραμμές A2-A6 θα μπορούσαν να συμμετέχουν στην αποκωδικοποίηση (πχ, μέσω μιας πύλης NAND) όμως αυτό δεν έχει συμβεί στο συγκεκριμένο κύκλωμα. Έτσι το PIO ενεργοποιείται με μία I/O διεύθυνση της δυαδικής μορφής 0xxxx00 έως 0xxxx11, όπου το x μπορεί να είναι 0 ή 1. Έτσι το PIO προσπελάσσεται από τις διευθύνσεις 00-03 αλλά και από τις 60-63 ή τις 1C-1F. Ομοίως το SIO προσπελάσσεται από διευθύνσεις I/O της δυαδικής μορφής 1xxxx00 έως 1xxxx11 (πχ, 80-83 ή F0-F3 ή 8C-8F κλπ).

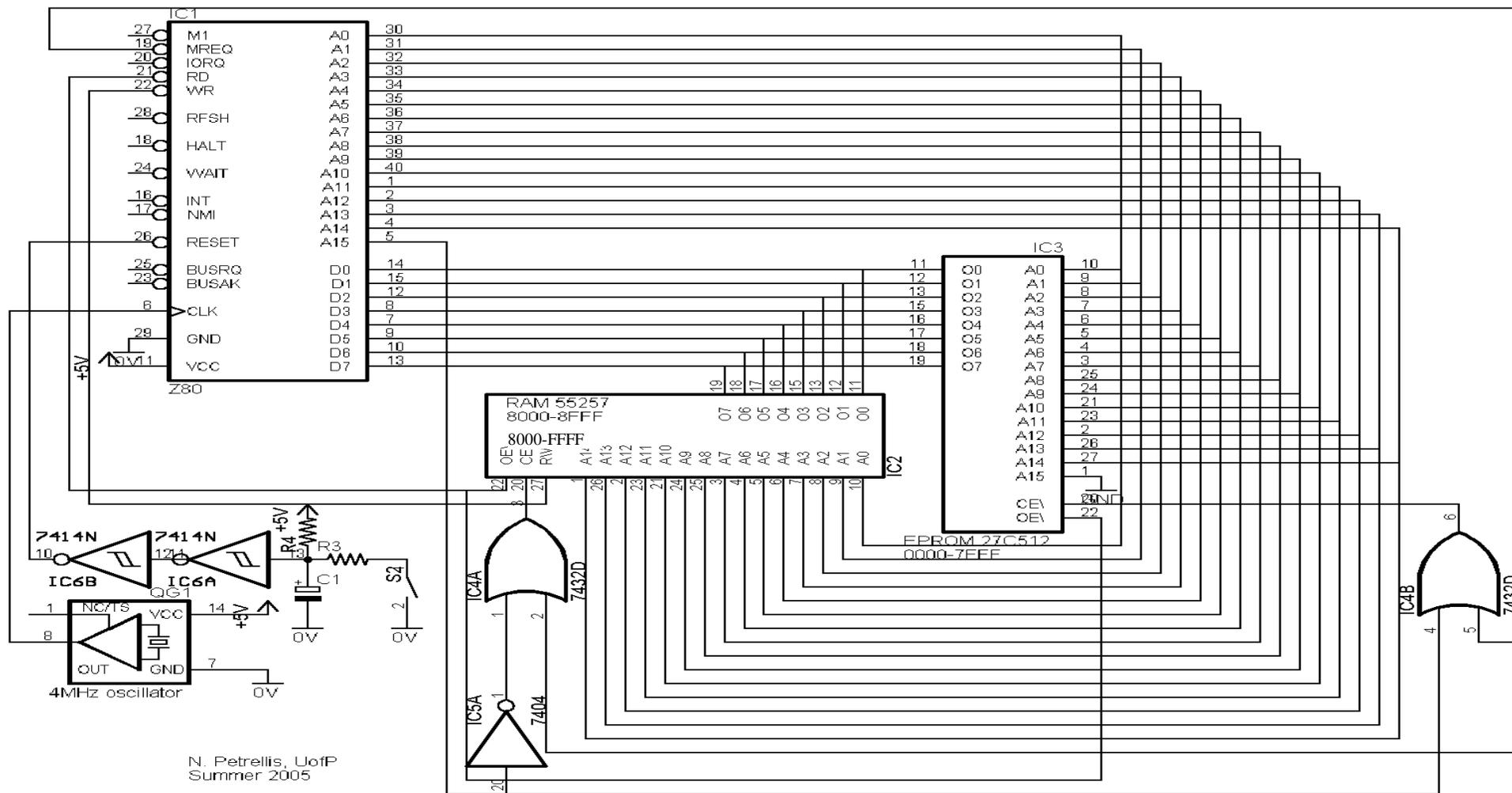
Οι γραμμές D0-D7 της SIO και της PIO συνδέονται απευθείας στο δίαυλο δεδομένων του Z80 για να μπορεί αυτός να προσπελάσει τις διευθύνσεις των καταχωρητών ελέγχου και δεδομένων των δύο περιφερειακών. Οι γραμμές RD και M1 των περιφερειακών συνδέονται απευθείας στις αντίστοιχες του Z80. Επίσης τα δύο αυτά περιφερειακά θα πρέπει να τροφοδοτηθούν με την κοινή τάση των 5V του συστήματος καθώς και το ρολόι.

Και τα δύο περιφερειακά μπορούν να προκαλέσουν διακοπές και έχουν τη δυνατότητα να «δείξουν» στον Z80 τη διεύθυνση της κατάλληλης ρουτίνας εξυπηρέτησης μέσω του διαύλου δεδομένων. Όμως για να καθοριστεί αν οι διακοπές του SIO ή του PIO θα έχουν μεγαλύτερη προτεραιότητα πρέπει να συνδεθούν κατάλληλα οι γραμμές INT, IEI, IEO. Στο συγκεκριμένο σύστημα έχει δοθεί υψηλότερη προτεραιότητα στο SIO και για το λόγο αυτό η γραμμή IEI του περιφερειακού αυτού συνδέεται με τα +5V. Η έξοδος IEO συνδέεται με την IEI του PIO και η IEO του PIO δεν συνδέεται πουθενά εφόσον δεν υπάρχει τρίτο ολοκληρωμένο με διακοπές στο σύστημα. Οι γραμμές INT των περιφερειακών συνδέονται απευθείας με την αντίστοιχη γραμμή του Z80 και αποκλείεται να ενεργοποιηθούν ταυτόχρονα δεδομένης της σύνδεσης των IEI, IEO που περιγράφηκε.

Όσον αφορά τη χρησιμότητα του ΡΙΟ, δύο ακροδέκτες από τις θύρες του έχουν συνδεθεί σε έναν διακόπτη (είσοδος) και ένα ρελέ (έξοδος). Ο διακόπτης όταν είναι ανοικτός προκαλεί τη σύνδεση 5V στο Β7 ενώ όταν κλείσει τη σύνδεση 0V στον ίδιο ακροδέκτη. Περισσότεροι τέτοιοι διακόπτες θα μπορούσαν να αποτελούν το πληκτρολόγιο (keypad) μιας εφαρμογής. Το ρελέ Κ1 από την πλευρά του θα μπορούσε να ελέγχει κάποιο μοτέρ, μια βάννα, κάποιο φωτισμό κλπ. Η ανάστροφα πολωμένη δίοδος D1 χρησιμοποιείται για τη βραχυκύκλωση της ανάστροφης τάσης που μένει στο πηνίο του ρελέ όταν αυτό απενεργοποιηθεί. Ένα ρελέ χρειάζεται αρκετό ρεύμα για να λειτουργήσει (ανάλογα και με το ρεύμα του διακόπτη που ελέγχει) και για το λόγο αυτό συνδέεται απευθείας στο συλλέκτη ενός τρανζίστορ (χωρίς τη μεσολάβηση μεγάλης αντίστασης). Το δε τρανζίστορ θα πρέπει να είναι αρκετής ισχύος για να αντέξει το ρεύμα που απαιτεί το ρελέ και είναι πιθανόν να έχει την ανάγκη ψύκτρας για την καλή απαγωγή της θερμότητας που αναπτύσσεται σε αυτό. Η τάση του πηνίου του ρελέ πρέπει να είναι συνήθως 12 ή 24V για τον έλεγχο συσκευών ισχύος, άρα στην περίπτωση αυτή το σύστημα χρειάζεται και δεύτερη τροφοδοσία πέραν των +5V που χρησιμοποιούν τα ψηφιακά μέρη.

Η SIO στο συγκεκριμένο σύστημα χρησιμοποιείται για να συνδεθεί με μία σειριακή ασύγχρονη θύρα RS232, όπου δεν χρησιμοποιούνται τα handshaking σήματα (RTS, CTS κλπ) παρά μόνο οι γραμμές αποστολής και λήψης δεδομένων (TxD, RxD) της Β θύρας. Τα επίπεδα των γραμμών αυτών είναι 0 ή 5 V τα οποία δεν είναι συμβατά με τις αρνητικές τάσεις που απαιτεί το πρωτόκολλο RS232. Για το λόγο αυτό μεσολαβεί το αναλογικό ολοκληρωμένο κύκλωμα MAX232 που κάνει αυτή ακριβώς τη μετατροπή των επιπέδων τάσης πριν τη σύνδεση με τον κονέκτορα. Το MAX232 διαθέτει και άλλες εισόδους/εξόδους με δυνατότητες αντίστοιχης μετατροπής επιπέδων τάσης στις οποίες θα μπορούσαν να συνδεθούν τα handshaking σήματα αν αυτά χρησιμοποιούνταν.

Για τον καθορισμό της ταχύτητας (baud rate) εκπομπής ή λήψης δεδομένων από την SIO θα πρέπει να συνδεθεί ένα ρολόι (πχ, ένας ταλαντωτής, κάποια έξοδος του Z80 CTC κλπ) στους ακροδέκτες RxD, TxD.



Σχήμα 2-61. Σύνδεση Z80 με μνήμη RAM και EPROM

