

## Διαφάνεια 3: IIC – Inter Integrated Circuit

1) Οι δύο αγωγοί που χρησιμοποιούνται έχουν τα ονόματα SDA και SCL, όπως θα δούμε και στην συνέχεια. Σε αυτούς τους δύο αγωγούς συνδέονται όλες οι συσκευές και μέσω του πρωτοκόλλου μπορούν να επικοινωνήσουν μεταξύ τους.

2) 100kbps είναι η τυπική ταχύτητα επικοινωνίας, τα 400Kbps είναι η γρήγορη ταχύτητα επικοινωνίας και τα 3.4Mbps είναι η υψηλή ταχύτητα επικοινωνίας. Για την αύξηση της ταχύτητας χρειάστηκαν να γίνουν διάφορες αλλαγές στο πρωτόκολλο IIC, αλλά και στο hardware των συσκευών που το υποστηρίζουν.

Στο πρωτόκολλο αυτό υπάρχει trade-off μεταξύ ταχύτητας και αγωγών. Χρησιμοποιούνται λίγοι αγωγοί, πράγμα που εξοικονομεί χώρο, αλλά υπάρχει επιβράδυνση στην ταχύτητα, πράγμα που δυσχεραίνει την επικοινωνία.

3) Ο αριθμός των συσκευών που μπορούν να συνδεθούν είναι 112 εάν χρησιμοποιηθούν διευθύνσεις των 7bit. Κάθε συσκευή για να επικοινωνήσει με μία άλλη θα πρέπει να ξέρει την διεύθυνσή της.

4) Όταν λέμε peer-to-peer εννοούμε ότι δύο συσκευές επικοινωνούν απ' ευθείας μεταξύ τους. Επίσης, για όσο διαρκεί η επικοινωνία των δύο συσκευών, οι άλλες δεν επιτρέπεται να παρεμβαίνουν στον δίαυλο. Σήμερα, οι περισσότερες συσκευές έχουν ειδικό hardware για να ανιχνεύουν τότε ο δίαυλος είναι απασχολημένος (BUSY).

5) -

6) Συνήθως το πρωτόκολλο αυτό χρησιμοποιείται σε πολύ μικρές αποστάσεις μέσα στο ίδιο κύκλωμα. Το επιτρεπτό μήκος των αγωγών που χρησιμοποιούνται για την επικοινωνία των συσκευών είναι συνήθως έως 4 μέτρα (9 έως 12 πόδια). Κι αυτό γιατί όταν χρησιμοποιούνται αγωγοί με μεγάλο μήκος αφενός αυξάνουν την χωρητικότητα του καλωδίου (φαινόμενο ανάκλασης του σήματος στο καλώδιο) και αφετέρου αυξάνεται ο θόρυβος του διαύλου τόσο πολύ που τα μεταδιδόμενα σήματα δεν είναι πλέον αναγνώσιμα από τις διάφορες συσκευές.

Μπορούμε να αυξήσουμε το μήκος των αγωγών επικοινωνίας με τους εξής τρόπους:

α) Να μειώσουμε το ρυθμό μετάδοσης της πληροφορίας (για 500bps φτάνει και τα 100μ)

β) Να χρησιμοποιήσουμε συστραμμένα ή/και μονωμένα καλώδια (twisted pair / shielded cables)

γ) Να χρησιμοποιήσουμε ειδικά κυκλώματα επανάληψης σήματος (repeaters) από την Philips.

## Διαφάνεια 4: Που χρησιμοποιείται

**EEPROM:** Electrically Erasable and Programmable Read Only Memory. Εάν δεν μας φτάνει η μνήμη που διαθέτει ο μικροελεγκτής, υπάρχει η δυνατότητα τοποθέτησης εξωτερικής μνήμης η οποία θα επικοινωνεί μέσω IIC με τον μικροελεγκτή.

**Sonar Rangers:** Πρόκειται για αισθητήρια υπερήχων, τα οποία έχουν την δυνατότητα να ανιληφθούν την απόσταση ανάμεσα στο αισθητήριο και κάποιο εμπόδιο. Εκπέμπουν ηχητικά κύματα σε υψηλές συχνότητες, μη ακουστές από τον άνθρωπο, και μέσω των ανακλάσεων που λαμβάνουν υπολογίζουν την απόσταση.

**CCD Camera:** Το ακρωνύμιο CCD προέρχεται από το Charged Coupled Device. Πρόκειται για έναν αισθητήρα εικόνας που αποτελείται από ένα ολοκληρωμένο κύκλωμα με μία συστοιχία συνδεδεμένων πυκνωτών ευαίσθητων στο φως. Χρησιμοποιούνται για ψηφιακές φωτογραφίες, στην φωτομετρία και στην αστρονομία. Για περισσότερες πληροφορίες μπορείτε να απευθυνθείτε στην online εγκυκλοπαίδεια [www.wikipedia.org](http://www.wikipedia.org) που έχει ένα πολύ ωραίο άρθρο.

**Real-Time Clock:** Είναι ένα ψηφιακό ρολόι που δείχνει την τρέχουσα ώρα και υπάρχει σε όλους τους σύγχρονους Η/Υ. Χαρακτηριστικό του είναι ότι δεν επηρεάζεται από την διακοπή της τροφοδοσίας, αφού περιέχει δικιά του αυτόνομη μπαταρία. Υπάρχουν και ρολόγια που δεν είναι Real-time και κάθε φορά που διακόπτεται η τροφοδοσία χάνουν την ένδειξή τους.

## Διαφάνεια 5: Ιστορική Αναδρομή

Η ανακάλυψη του πρωτοκόλλου IIC από την Phillips έγινε κατά την κατασκευή ενός νέου μοντέλου τηλεόρασης. Συγκεκριμένα οι μηχανικοί της εταιρίας έψαχναν έναν απλό και αποτελεσματικό τρόπο επικοινωνίας μεταξύ της κεντρικής υπολογιστικής μονάδας και των υπολοίπων ηλεκτρονικών τμημάτων της τηλεόρασης (πχ τμήμα συντονισμού δέκτη – tuner, κυκλώματα προβολής κινούμενης εικόνας – video, κύκλωμα ήχου κοκ). Μέχρι τότε οι περιφερειακές συσκευές συνδεόταν με την MCU σαν επιπλέον διευθύνσεις μνήμης I/O, πράγμα που οδηγούσε στην χρήση των Data Bus και Address Bus. Κάτι τέτοιο όμως σήμαινε **πολλές καλωδιώσεις**, κίνδυνος **ηλεκτρομαγνητικών παρεμβολών** και χρήση **Address Decoders**, πράγμα που δεν είναι αποδεκτό για μαζική παραγωγή συσκευών. Έτσι κατέληξαν στην ανακάλυψη του πρωτοκόλλου IIC, το οποίο σήμερα είναι πολύ δημοφιλές και ευρέως διαδεδομένο.

Φυσικά για λόγους ανταγωνισμού έσπευσαν και οι άλλες εταιρίες να κατασκευάσουν παρόμοια πρωτόκολλα. Κύριο παράδειγμα αποτελεί η Intel με την κατασκευή του SMBUS το 1995, το οποίο κατά ένα μεγάλο μέρος του στηρίχθηκε στο ήδη υπάρχον IIC.

Τελικά η βιομηχανία συγχώνευσε κατά κάποιον τρόπο τις προσπάθειες αυτές και ενσωμάτωσε το πρωτόκολλο IIC στους μικροελεγκτές. Στους σύγχρονους μικροελεγκτές υπάρχουν ειδικές μονάδες επικοινωνίας με χρήση του πρωτοκόλλου IIC, οι οποίες έχουν ειδικά κατασκευασμένο hardware για τον σκοπό αυτό. Πρωτοπόρος στην υιοθέτηση του IIC αποτελεί η εταιρία κατασκευής ολοκληρωμένων κυκλωμάτων Microchip ([www.microchip.com](http://www.microchip.com)) και στην συνέχεια ακολούθησαν πολλές άλλες (Texas Instruments, Analog Devices, Siemens, Atmel κα. Οι γνωστοί μικροελεγκτές AVR έχουν ενσωματώσει το πρωτόκολλο IIC στο TWI τμήμα τους.

## Διαφάνεια 6: Συνδεσμολογία

Αυτή είναι η τυπική συνδεσμολογία με μία συσκευή Master (μικροελεγκτής) και πολλούς Slave (αισθητήρια, άλλοι μικροελεγκτές κοκ). Οι γραμμές SDA και SCL μπορούν να χρησιμοποιηθούν για επικοινωνία και προς τις δύο κατευθύνσεις.

Για την ορθή λειτουργία του διαύλου απαιτείται η σύνδεση των δύο αγωγών με την τάση τροφοδοσίας μέσω δύο pull-up αντιστάσεων. Έτσι επιτυγχάνεται μείωση του ηλεκτρομαγνητικού θορύβου.

Κάθε συνδεδεμένη συσκευή έχει την δική της μοναδική διεύθυνση και μπορεί να στέλνει ή να παίρνει πληροφορία ανάλογα με τον τύπο της και τον τρόπο χρήσης. Για παράδειγμα μία LCD οθόνη θα παίρνει πληροφορίας ενώ μία πυξίδα θα στέλνει πληροφορία.

Υπάρχει η δυνατότητα παραπάνω από μία συσκευές να είναι Master, αλλά όχι ταυτόχρονα. Όταν μία συσκευή Master ξεκινήσει την επικοινωνία, τότε όλες οι άλλες συσκευές αυτόματα βρίσκονται σε κατάσταση Slave. Συνήθως μικροελεγκτές παίζουν τον ρόλο Master.

Για τις γραμμές αυτές ισχύει το εξής: Active = LOW, passive = HIGH. Δηλαδή επικρατεί HIGH όταν είναι σε κατάσταση IDLE και μόλις μία συσκευή θέλει να επικοινωνήσει ρίχνει την γραμμή σε LOW.

## Διαφάνεια 7: Οι έννοιες Master - Slave

Παρόλο που το πρωτόκολλο παρέχει υποστήριξη για πολλαπλές συσκευές Master, οι περισσότερες διατάξεις χρησιμοποιούν μονάχα μία συσκευή Master και όλες οι υπόλοιπες συσκευές θεωρούνται Slaves.

## Διαφάνεια 8: Σήματα Ελέγχου (1)

Για το σήμα START έχουμε μετάβαση της γραμμής SDA από 1 σε 0 ενώ η γραμμή SCL βρίσκεται σε κατάσταση 1. Για το σήμα STOP έχουμε αντίστοιχα μετάβαση της γραμμής SDA από 0 σε 1 όταν στην γραμμή SCL επικρατεί 1.

Στην περίπτωση που μεταδοθεί ένα σήμα STOP στην μέση ενός μηνύματος, τότε αμέσως η επικοινωνία διακόπτεται. Η καλή πρακτική επιβάλλει στην περίπτωση αυτή να απορριφθεί η πληροφορία του μηνύματος και να συνεχιστεί η κατάσταση LISTEN.

Η λειτουργία Repeated Start χρησιμοποιείται όταν ένας Master έχει τελειώσει την επικοινωνία με μία συσκευή και θέλει να απευθυνθεί σε κάποια άλλη. Στην περίπτωση αυτή μπορεί αντί να στείλει σήμα STOP, να στείλει ένα νέο σήμα START για την έναρξη της επικοινωνίας με την άλλη συσκευή.

Από την στιγμή που θα εμφανιστεί το σήμα START έως ότου μεταδοθεί το σήμα STOP, τότε ο δίαυλος θεωρείται κατειλημμένος (BUSY).

## Διαφάνεια 9: Σήματα Ελέγχου (2)

Σε κάθε byte που μεταδίδεται, μετά την λήψη του 8ου bit θα πρέπει ο δέκτης να στείλει σήμα ACK, δηλαδή να ρίξει σε επίπεδο LOW την γραμμή SDA για διάστημα ενός χτύπου του ρολογιού.

Εάν μετά την λήψη του 8ου bit δεν σταλθεί σήμα ACK από τον δέκτη προς τον πομπό, τότε λέμε ότι έχουμε κατάσταση NACK.

1) Μπορεί η διεύθυνση με την οποία θέλει να επικοινωνήσει ο πομπός είτε να μην αντιστοιχεί σε κάποια συσκευή από αυτές που είναι συνδεδεμένες στον δίαυλο είτε να αντιστοιχεί αλλά η συσκευή αυτή να υπολειτουργεί ή να μην λειτουργεί καθόλου.

2) Όπως έχουμε ήδη αναφέρει, η συσκευή Master καθορίζει την συχνότητα του ρολογιού επικοινωνίας SCL, δηλαδή τον ρυθμό μετάδοσης της πληροφορίας. Εάν το clock χτυπάει με μεγαλύτερη συχνότητα από εκείνη που μπορεί να παρακολουθήσει η συσκευή Slave, τότε εμφανίζεται το φαινόμενο απώλειας συγχρονισμού. Για τον λόγο αυτό θα πρέπει πάντα να προσέχουμε το εξής: Το ρολόι της συσκευής Master να είναι αρκετά αργό έτσι ώστε να μπορεί να το παρακολουθήσει και η πιο αργή συσκευή Slave που βρίσκεται συνδεδεμένη στον δίαυλο.

3) Όταν ένας από τους δύο αγωγούς (SCL, SDA) είναι σε κατάσταση LOW μονίμως, τότε λέμε ότι ο δίαυλος έχει μπλοκάρει.

Στις περιπτώσεις αυτές, η συσκευή Master θα πρέπει να παράγει σήμα STOP για ακύρωση και εν συνεχεία επανάληψη της επικοινωνίας.

## Διαφάνεια 10: Διευθύνσεις

Το bit R/W είναι ως προς την συσκευή Master. Δηλαδή όταν είναι 1, η συσκευή Master γράφει δεδομένα στον δίαυλο άρα γίνεται πομπός. Αντίθετα εάν είναι 0 τότε ο Master διαβάζει δεδομένα από τον δίαυλο, άρα γίνεται δέκτης.

Για τον καθορισμό της διεύθυνσης κάθε συσκευής υπάρχουν συγκεκριμένα pins πάνω στο τσιπ τα οποία μπορούμε να συνδέσουμε με την γη ή την τροφοδοσία για να καθορίσουμε την διεύθυνση.

Για λόγους οικονομίας pins, από πολλούς κατασκευαστές ακολουθείται η εξής πρακτική: Για κάθε τύπο συσκευής είναι καθορισμένα τα πρώτα 4 bit και τα υπόλοιπα 3 επιλέγονται ελεύθερα από εμάς. Πχ, οι μνήμες EEPROM έχουν τον κωδικό 1010.

Εναλλακτικά μπορούμε να προγραμματίσουμε την διεύθυνση κάποιας συσκευής, αλλά μόνο εάν αυτή έχει μικροελεγκτή.

Σε κάθε περίπτωση θα πρέπει να συμβουλευόμαστε το datasheet του κατασκευαστή κάθε συσκευής για τον καθορισμό της διεύθυνσής της στον δίαυλο.

### **General Call Address:**

Η General Call Address απευθύνεται σε όλες τις συσκευές του διαύλου. Όσες είναι κατάλληλα προγραμματισμένες για την διαχείρισή της, απαντούν με σήμα ACK. Σε αντίθετη περίπτωση την αγνοούν.

Χρησιμοποιείται όταν η συσκευή Master θέλει να μεταδώσει το ίδιο μήνυμα σε όλες τις συσκευές Slave του διαύλου.

Μπορεί να χρησιμοποιηθεί σε περιπτώσεις που θέλουμε να πάρουμε ταυτόχρονες μετρήσεις από πολλά αισθητήρια. Στην περίπτωση αυτή ο μικροελεγκτής μας θα μπορεί

να δώσει εντολή ταυτόχρονα σε όλα τα αισθητήρια για διεξαγωγή μέτρησης και στην συνέχεια να επικοινωνήσει με το κάθε ένα για να λάβει την μέτρηση.

Ανάλογα με το byte που μεταδίδεται μετά από την διεύθυνση αυτή (Command Byte) συμβαίνει και η αντίστοιχη λειτουργία:

**00000110:** Γίνεται Reset και ανακατανομή Διευθύνσεων σε όλες τις συσκευές που είναι συνδεδεμένες στον δίαυλο.

**00000100:** Απλή Ανακατανομή Διευθύνσεων. Κάθε συσκευή διαβάζει πάλι την διεύθυνσή της

**XXXXXXX1:** Η συσκευή Master με τον τρόπο αυτό δηλώνει ότι δεν ξέρει που πρέπει να στείλει τα δεδομένα. Είναι ουσιαστικά μία έκκληση για επίδειξη προσοχής από πλευράς των υπολοίπων συσκευών Master. Τα υπόλοιπα bits του byte εμπεριέχουν την διεύθυνση της συσκευής.

### **10 bit Addressing Mode:**

Τα δύο MSB βρίσκονται στο πρώτο byte και τα υπόλοιπα 8 LSB βρίσκονται στο δεύτερο byte.

Το ότι χρησιμοποιείται κρατημένη διεύθυνση εξασφαλίζει ότι οι διευθύνσεις 10 bit που μεταδίδονται δεν θα επηρεάζουν την λειτουργία όσων συσκευών δεν τις υποστηρίζουν. Με τον τρόπο αυτό μπορούμε να συνδέσουμε άφοβα στον ίδιο δίαυλο συσκευές που είναι παλιές και ενδεχομένως να μην υποστηρίζουν 10bit διευθύνσεις.

## Διαφάνεια 13: Τρόπος Λειτουργίας (3)

Τα bit δεδομένων δημιουργούνται από τον πομπό καθώς το ρολόι SCL χτυπάει. Στον 9ο παλμό του ρολογιού ο πομπός ελευθερώνει την γραμμή SDA και ο δέκτης πρέπει να την κατεβάσει σε επίπεδο LOW έτσι ώστε να δώσει σήμα ACK. Μετά το ack ο δέκτης πρέπει να απελευθερώσει την γραμμή SDA, έτσι ώστε ο πομπός να μπορεί να στείλει το επόμενο byte δεδομένων.

Ένα κενό μήνυμα που αποτελείται από ένα σήμα START και ένα σήμα STOP είναι απαγορευμένο από το πρωτόκολλο.

## Διαφάνεια 14: Παράδειγμα 1

Σύμφωνα με τα όσα έχουμε ήδη πει, αρχικά η συσκευή Master (δηλαδή ο μικροελεγκτής) θα πρέπει να εκπέμψει το σήμα START για να δηλώσει την έναρξη της επικοινωνίας.

Αμέσως μετά θα πρέπει να εκπεμφθεί η διεύθυνση της συσκευής EEPROM συνοδευόμενη από το Direction bit. Επειδή στην περίπτωση μας απαιτείται εγγραφή στην μνήμη, θα πρέπει να τοποθετηθεί το bit 0 που αντιστοιχεί στο Write.

Στην συνέχεια περιμένουμε έως ότου η συσκευή απαντήσει με ACK, πράγμα που σημαίνει αφενός ότι υπάρχει και αφετέρου ότι είναι έτοιμη να δεχτεί τα δεδομένα. Μόλις λάβουμε σήμα ACK, τότε θα πρέπει να στείλουμε το πρώτο byte.

Το πρώτο byte που λαμβάνει η συσκευή στις περισσότερες των περιπτώσεων αφορά

κάποια εντολή που της δίνεται. Στην συγκεκριμένη περίπτωση θα μπορούσε κάλλιστα να εμπεριέχει την θέση μνήμης όπου θέλουμε να γίνει η εγγραφή.

Σε κάθε περίπτωση θα πρέπει να διαβάσουμε προσεκτικά το φυλλάδιο του κατασκευαστή (datasheet) πριν χρησιμοποιήσουμε την οποιαδήποτε συσκευή, διότι οι λειτουργίες αυτές διαφέρουν από συσκευή σε συσκευή και από κατασκευαστή σε κατασκευαστή.

Μόλις λάβουμε ACK, προχωρούμε στην αποστολή του επόμενου byte κοκ σύμφωνα με όσα έχουν ήδη αναλυθεί. Στο τέλος για τον τερματισμό της επικοινωνίας στέλνουμε το σήμα STOP.

Πριν το S και μετά το P ο δίαυλος είναι ελεύθερος IDLE, ενώ για όσο διαρκεί η επικοινωνία είναι BUSY και καμία από τις άλλες συσκευές εκτός αυτών των δύο δεν δικαιούται να παρέμβει με καθ' οιονδήποτε τρόπο στην επικοινωνία.

**ΠΡΟΣΟΧΗ:** Θα πρέπει πριν ο Master αποστείλει το σήμα STOP να έχει δεχθεί ACK. Ειδάλλως ο δίαυλος θα μπλοκαριστεί από την συσκευή Slave, μέχρις ότου καταφέρει να στείλει το σήμα ACK. Δηλαδή η συσκευή Slave θα κρατά την γραμμή SDA σε κατάσταση LOW, έως ότου ο Master δώσει έναν παλμό ρολογιού στην γραμμή SCL.

Στο σχήμα ο Slave στέλνει στον Master μόνο τα σήματα ACK. Όλα τα υπόλοιπα στέλνονται από τον Master στον Slave.

## Διαφάνεια 15: Παράδειγμα 2

Η λειτουργία είναι παρόμοια με εκείνη της προηγούμενης σελίδας. Οι μόνες διαφορές που υπάρχουν είναι οι εξής:

1) Εφόσον θέλουμε ο μικροελεγκτής να διαβάσει πληροφορία από την μνήμη EEPROM, θα πρέπει αυτή τη φορά το Direction Bit να είναι 1, το οποίο αντιστοιχεί στο Read.

2) Όταν ο Master θελήσει να σταματήσει την ροή δεδομένων και να κλείσει την επικοινωνία, θα πρέπει στο τελευταίο byte να μην στείλει το σήμα ACK και στην συνέχεια το σήμα STOP.

Εάν η συσκευή Slave δεν δεχθεί σήμα ACK, τότε θα σταματήσει να στέλνει bytes προς την συσκευή Master, οπότε η δεύτερη μπορεί να τερματίσει την επικοινωνία.

Αυτό γίνεται διότι μερικές συσκευές εάν λάβουν ACK, ακόμη κι αν έχει τελειώσει η πληροφορία που θέλουν να μεταδώσουν, συνεχίζουν και πάλι από την αρχή. Άρα λοιπόν είναι απαραίτητο στο τελευταίο Byte να μην σταλθεί το σήμα ACK.

Στο σχήμα, τα START, Slave Address, R/W bit στέλνονται από τον Master στον Slave. Στην συνέχεια ο Slave απαντά με ACK και στέλνει το πρώτο byte. Από εκεί και πέρα, για κάθε byte που στέλνει ο Slave, ο Master απαντά με ACK, εκτός από το τελευταίο. Για τον τερματισμό της επικοινωνίας, ο Master μετά το NACK μεταδίδει στον δίαυλο ένα σήμα STOP.

## Διαφάνεια 16: Λεπτομέρειες

1) Το bit διαβάζεται από τον δέκτη μόνο καθ' όσον η γραμμή Clock παραμένει HIGH. Η κατάσταση της γραμμής SDA θα πρέπει να αλλάζει μονάχα όσο η γραμμή SCL είναι σε επίπεδο LOW.

3) Η συσκευή Master περιμένει έως ότου η γραμμή SCL ανέλθει σε επίπεδο HIGH.

Αυτό συμβαίνει ιδίως σε αργές συσκευές Slave, αργές από άποψη εσωτερικής λειτουργίας. Εάν στείλουμε σε αυτές πολλά byte πληροφορίας, ενδεχομένως να μην μπορέσουν να τα επεξεργαστούν σωστά και να οδηγηθούμε σε σφάλματα. Για παράδειγμα ένα Sonar μέχρι να πάρει μέτρηση ή μία μνήμη EEPROM μέχρι να διαβάσει ή να γράψει δεδομένα. Με τον τρόπο αυτό η συσκευή Slave ειδοποιεί τον Master ότι είναι απασχολημένη και δεν μπορεί να δεχθεί περαιτέρω πληροφορία.

Εάν για παράδειγμα η συσκευή Slave είναι μία μνήμη EEPROM που, ως γνωστόν, έχει αρκετά μεγάλο χρόνο εγγραφής. Στην περίπτωση που τα δεδομένα έρχονται με μεγαλύτερο ρυθμό από εκείνον που μπορεί να χειριστεί, τότε μπορεί να κάνει δύο πράγματα:

α) Εάν όσο διαρκεί η εγγραφή στην μνήμη, της έρθει καινούργιο byte προς εγγραφή, τότε μπορεί να στείλει σήμα NACK. Στην περίπτωση αυτή η συσκευή Master θα στείλει σήμα STOP απελευθερώνοντας τον δίαυλο και θα προσπαθήσει αργότερα.

β) Να καθυστερήσει τον Master έως ότου γίνει σωστά η εγγραφή. Στην περίπτωση αυτή έχουμε δέσμευση του διαύλου για μεγάλα χρονικά διαστήματα, ιδίως όταν τα δεδομένα προς εγγραφή είναι πολλά.

Παρόλο που το πρωτόκολλο δίνει αυτήν την δυνατότητα, λίγες είναι οι συσκευές Slave που την χρησιμοποιούν.

## Διαφάνεια 17: Πολλαπλές Συσκευές Master (1)

Κανονικά οι διαφορετικές συσκευές Master παρακολουθούν συνέχεια τον δίαυλο.

Κάθε φορά που μεταδίδεται το σήμα START από μία συσκευή Master, οι υπόλοιποι Master γίνονται Slave και περιμένουν να ακούσουν την διεύθυνσή τους για σύναψη επικοινωνίας. Εάν δεν την ακούσουν, τότε δεν παρεμβαίνουν στον δίαυλο μέχρι να μεταδοθεί το σήμα STOP. Άρα οι συσκευές Master που θέλουν να στείλουν δεδομένα, καταλαμβάνουν διαδοχικά τον δίαυλο.

Όταν υπάρχουν πολλές συσκευές Master στον ίδιο δίαυλο, προκύπτουν τα ακόλουθα προβλήματα:

α) Θα πρέπει να υλοποιηθεί ένας αλγόριθμος που θα επιτρέπει μονάχα σε μία συσκευή Master κάθε φορά να ολοκληρώνει την επικοινωνία. Όλες οι υπόλοιπες συσκευές θα πρέπει να σταματούν την επικοινωνία μόλις ανακαλύπτουν ότι έχουν χάσει στην διαδικασία επιλογής – διαιτησίας (arbitration). Το γεγονός ότι πολλοί Masters έχουν ξεκινήσει ταυτόχρονα την επικοινωνία δεν θα πρέπει να είναι ορατό από τις συσκευές Slave, ούτως ώστε να μην υπάρξει απώλεια δεδομένων.

β) Μπορεί διαφορετικές συσκευές Master να χρησιμοποιούν διαφορετικές συχνότητες

επικοινωνίας, αλλά πρέπει να βρεθεί ένας τρόπος συγχρονισμού των ρολογιών αυτών.

## Διαφάνεια 18: Πολλαπλές Συσσκευές Master (2)

Τι γίνεται όμως σε περιπτώσεις όπου δύο συσκευές Master μεταδίδουν ταυτόχρονα το σήμα START ή μία συσκευή Master μόλις έχει κάνει RESET και δεν έχει συλλάβει το σήμα START? Πώς μπορεί να αντιληφθεί ότι ο δίαυλος είναι απασχολημένος? Αυτό το πράγμα γίνεται μέσω της διαιτησίας.

Ο χαμένος είναι εκείνος που προσπαθεί να τραβήξει την γραμμή SDA σε επίπεδο HIGH, ενώ κάποια άλλη συσκευή Master την κρατάει σε επίπεδο LOW. Στην περίπτωση αυτή η γραμμή παραμένει LOW λόγω κυκλωματικής κατασκευής των ελεγκτών IIC (wired-AND: βοηθάει να σκεφτούμε ότι αυτό που τελικά επικρατεί στον δίαυλο προκύπτει εάν κάνουμε την λογική πράξη AND μεταξύ των διαφόρων σημάτων). Τότε η πρώτη συσκευή αντιλαμβάνεται ότι ο δίαυλος είναι κατειλημμένος από μία άλλη συσκευή και περιμένει έως ότου σταλθεί το σήμα STOP που δηλώνει ότι ο δίαυλος ελευθερώθηκε. Δηλαδή ουσιαστικά ο νικητής της “μονομαχίας” είναι εκείνος που θέλει να στείλει 0 ενώ ο άλλος θέλει να στείλει 1. Γι' αυτόν τον λόγο, όταν μία συσκευή Master στέλνει ένα bit στον δίαυλο, τότε ελέγχει εάν το επίπεδο τάσης της γραμμής SDA είναι το επιθυμητό.

Η διαιτησία μπορεί να συνεχίζεται για πολλά bits έως ότου καθοριστεί ο νικητής, δηλαδή απομείνει μονάχα μια συσκευή Master που θα ελέγχει τον δίαυλο. Αρχικά συγκρίνονται οι διευθύνσεις των συσκευών προορισμού (Slave) και στην περίπτωση που δύο ή περισσότεροι Masters απευθύνονται στην ίδια συσκευή, τότε η διαιτησία συνεχίζει και στα πακέτα δεδομένων. Ο χαμένος Master θα πρέπει αυτομάτως να μπει σε Slave Mode, να απελευθερώσει τον δίαυλο SDA και να προσπαθήσει ξανά αργότερα, όταν δηλαδή ο master που κέρδισε εκπέμψει σήμα STOP.

ΠΑΡΑΔΕΙΓΜΑ: Αρχικά και οι δύο συσκευές Master στέλνουν την ίδια διεύθυνση στον δίαυλο και ο Slave απαντά με σήμα ACK. Στο σημείο αυτό και οι δύο Master έχουν την εντύπωση ότι κατέχουν τον δίαυλο και συνεχίζουν με την αποστολή του πρώτου byte δεδομένων. Η διαιτησία λαμβάνει χώρα την στιγμή που το bit που στέλνει ο ένας Master διαφέρει από αυτό που στέλνει ο άλλος. Νικητής στέφεται εκείνος που θέλει να στείλει το bit 0, δηλαδή να κρατήσει την γραμμή SDA σε κατάσταση LOW. Ο χαμένος αποσύρεται (back off condition), συμπεριφέρεται πλέον ως συσκευή Slave και δεν πειράζει την κατάσταση των αγωγών SCL και SDA (κίτρινη ζώνη). Όταν δοθεί σήμα STOP, τότε ο χαμένος το αντιλαμβάνεται και προσπαθεί εκ νέου να στείλει τα δεδομένα του.

Απαγορεύεται η σύγκριση - διαιτησία μεταξύ: ενός σήματος REPEATED START κι ενός bit δεδομένων, ενός σήματος STOP κι ενός bit δεδομένων, ενός σήματος REPEATED START κι ενός σήματος STOP. Για να μην συμβούν οι συγκρίσεις αυτές, θα πρέπει κάθε επικοινωνία να περιέχει τον ίδιο αριθμό πακέτων δεδομένων, αλλιώς το αποτέλεσμα της διαιτησίας είναι απροσδιόριστο. Φυσικά για να γίνει κάτι τέτοιο υπεύθυνο είναι το πρόγραμμα που τρέχει στους Masters.

## Διαφάνεια 19: Συγχρονισμός Ρολογιών

Μόλις το ρολόι ενός Master οδηγήσει την γραμμή SCL από 1 σε 0, τότε γίνεται reset σε



όλα τα ρολόγια των Masters του διαύλου και αυτά ξεκινούν το μέτρημα της LOW Period. Όποιο ρολόι τελειώσει πρώτο, προσπαθεί να μεταβάλλει την κατάσταση της γραμμής SCL από 0 σε 1. Αυτό όμως δεν μπορεί να το καταφέρει εκ κατασκευής εάν υπάρχει κάποιο άλλο ρολόι το οποίο βρίσκεται σε κατάσταση LOW. Άρα λοιπόν η γραμμή SCL βρίσκεται σε κατάσταση LOW έως ότου τελειώσει η LOW Period του ρολογιού κάθε συσκευής Master.

Το αντίστροφο συμβαίνει για τον καθορισμό της HIGH περιόδου του SCL. Μόλις τελειώσει η Low περίοδος, τότε τα ρολόγια αρχίζουν να μετρούν την HIGH Period του. Αυτό που θα τελειώσει πρώτο οδηγεί την γραμμή SCL από 1 σε 0 και αναγκάζει τα υπόλοιπα ρολόγια να κάνουν reset όπως προηγούμενως.

Ο μηχανισμός αυτός λειτουργεί σαν χειραψία (Handshake) μεταξύ αργών και γρήγορων συσκευών Master.

Υπάρχουν οι εξής τρόποι υλοποίησης του πρωτοκόλλου IIC στο κύκλωμα που κατασκευάζουμε:

- 1) Να χρησιμοποιήσουμε ειδικό κύκλωμα IIC Controller, το οποίο τις περισσότερες φορές βρίσκεται ενσωματωμένο στις συσκευές.
- 2) Να υλοποιήσουμε το πρωτόκολλο αυτό μέσω software. Στην περίπτωση αυτή θα πρέπει να χρησιμοποιήσουμε 2 general purpose I/O pins σαν θύρες. Είναι εφικτό μόνο για κύκλωμα single Master.

## Διαφάνεια 20: IIC στον AVR (1)

Υπάρχουν 2 pins που λέγονται SDA και SCL (στον ATmega128 είναι τα pins 25 και 26), όπου συνδέονται τα δύο καλώδια του διαύλου. Υπάρχει και φίλτρο για Spikes μικρότερα των 50ns. Επίσης το εσωτερικό κύκλωμα του AVR διαθέτει pull-up αντιστάσεις συνδεδεμένες με αυτά τα pins, οι οποίες σε μερικές εφαρμογές μπορούν να αντικαταστήσουν τις εξωτερικές που απαιτούνται από το IIC.

Bit Rate Generator: Ελέγχει το ρολόι SCL όταν ο μικροελεγκτής λειτουργεί ως Master. Η συχνότητα του ρολογιού καθορίζεται από την τιμή του TWBR register και των prescaler bits του TWSR, σύμφωνα με τον τύπο που θα δοθεί σε επόμενη διαφάνεια. Εάν η συσκευή μας είναι συνδεδεμένη ως Slave, τότε θα πρέπει η ταχύτητα του SCL να είναι τουλάχιστον 16 φορές μικρότερη από αυτήν της CPU, για να υπάρχει αρκετό χρονικό περιθώριο επεξεργασίας των δεδομένων.

Bus Interface Unit: Ο TWDR περιέχει το byte που θα αποσταλεί ή που έχει ληφθεί. Επιπροσθέτως υπάρχει και το τμήμα ACK, όπου βρίσκεται αποθηκευμένο το bit ACK/NACK που θα σταλεί ή που έχει ληφθεί. Το τμήμα αυτό δεν είναι προσπελάσιμο από τον προγραμματιστή παρά μόνο μέσω των καταχωρητών TWCR και TWSR. Η μονάδα START/STOP ανιχνεύει τα σήματα που μεταδίδονται στον δίαυλο. Τέλος, η μονάδα εντοπισμού διαιτησίας (Arbitration Detection) παρακολουθεί συνεχώς τον δίαυλο όταν ο μικροελεγκτής στέλνει δεδομένα ως Master και προσπαθεί να εντοπίσει εάν συμβαίνει κάποια διαιτησία και εάν η συσκευή αυτή έχει χάσει. Στην περίπτωση αυτή ενημερώνει την Control Unit, για να ληφθούν τα αναγκαία μέτρα και να παραχθούν οι αντίστοιχοι κώδικες κατάστασης (TWSR).

Address Match Unit: Η μονάδα αυτή ελέγχει εάν η διεύθυνση που έχει μεταδοθεί στον δίαυλο ταυτίζεται με την διεύθυνση που έχει ο μικροελεγκτής και βρίσκεται αποθηκευμένη

στον καταχωρητή TWAR ή με την διεύθυνση General Call (0000000), εάν φυσικά ο AVR είναι ρυθμισμένος να απαντά σε General Calls. Εάν οι διευθύνσεις αυτές συμπίπτουν, τότε ενημερώνεται η Control Unit για να ληφθούν τα αναγκαία μέτρα. Στην περίπτωση που ο μικροελεγκτής βρίσκεται σε κατάσταση Sleep, τότε αμέσως ενεργοποιείται και διαχειρίζεται τα δεδομένα.

**Control Unit:** Η μονάδα αυτή περιέχει δύο καταχωρητές: τους TWCR και TWSR. Ο πρώτος χρησιμοποιείται από τους προγραμματιστές για τον καθορισμό της συμπεριφοράς της συσκευής και ο δεύτερος ανανεώνεται ανά τακτά χρονικά διαστήματα με συγκεκριμένους κώδικες που αντικατοπτρίζουν την κατάσταση του διαύλου. Εάν η κατάσταση αυτή απαιτεί την προσοχή του προγράμματος, τότε εκπέμπεται το σήμα διακοπής και η γραμμή SCL παίρνει την τιμή 0 έως ότου το πρόγραμμα τελειώσει την επεξεργασία των δεδομένων και η επικοινωνία μπορεί να συνεχιστεί.

## Διαφάνεια 21: IIC στον AVR (2)

**TWBR:** Η τιμή του θα πρέπει να είναι τουλάχιστον ίση με 10. Μαζί με τα bit TWPS1/0 καθορίζουν την συχνότητα του ρολογιού σύμφωνα με τον τύπο που παρατίθεται.

**TWCR: TWINT (TWI Interrupt Flag):** Όταν το hardware τελειώσει κάποια εργασία (πχ λήψη ενός byte δεδομένων), θέτει το bit TWINT ίσο με 1 έτσι ώστε αυτό να γίνει αντιληπτό από το πρόγραμμα. Όταν το πρόγραμμα “δει” την σημαία, θα πρέπει να την μηδενίσει γράφοντας λογικό 1 (δεν μηδενίζεται από το hardware). Μέχρι όμως να μηδενιστεί η σημαία, η γραμμή SCL κρατιέται σε επίπεδο 0 (Clock Stretching). Ο μηδενισμός της σημαίας αυτής ισοδυναμεί με την συνέχιση της λειτουργίας του TWI, οπότε θα πρέπει από πριν το πρόγραμμα να έχει καθορίσει τα περιεχόμενα των καταχωρητών TWAR, TWSR και TWDR, διότι δεν επιτρέπεται η αλλαγή τους κατά την διάρκεια λειτουργίας του TWI.

**TWEA:** TW Enable Acknowledge ελέγχει την παραγωγή του σήματος ACK. Εάν το bit αυτό είναι 1, τότε η συσκευή μπορεί να απαντήσει με ACK στις περιπτώσεις που έχουν ήδη περιγραφεί προηγουμένως (εάν “ακούσει” την διεύθυνσή της και κάθε φορά που λαμβάνει κάποιο byte). Αντιθέτως, εάν γίνει 0, τότε θεωρούμε ότι η συσκευή έχει αποσυνδεθεί από τον δίαυλο, αφού δεν πρόκειται να απαντήσει με ACK σε καμία περίπτωση. **TWSTA:** Χρησιμοποιείται για την παραγωγή σήματος START. Το hardware αρχικά ελέγχει εάν ο δίαυλος είναι ελεύθερος και εάν όχι περιμένει μέχρι να μεταδοθεί στον δίαυλο το σήμα STOP. **TWSTO:** Εάν το bit αυτό πάρει την τιμή 1, τότε θα μεταδοθεί το σήμα STOP στον δίαυλο και στην συνέχεια θα μηδενιστεί. **TWWC:** Write Collision Flag – Χρησιμοποιείται για τον εντοπισμό συγκρούσεων κατά την εγγραφή στον δίαυλο, όπως για παράδειγμα όταν προσπαθήσουμε να γράψουμε στον καταχωρητή TWDR ενώ η τιμή του TWINT είναι 0, κάτι το οποίο φυσικά απαγορεύεται. **TWEN:** Είναι το bit ενεργοποίησης του TWI. Εάν είναι ίσο με 1, τότε το TWI Interface παίρνει τον έλεγχο των δύο pins. Στην αντίθετη περίπτωση απενεργοποιείται πλήρως. **TWIE:** Interrupt Enable – Εάν αυτό το bit είναι 1 και το bit I στον Status register είναι επίσης 1, τότε ενεργοποιούνται οι διακοπές του TWI.

**TWSR:** Τα πρώτα 5 bits χρησιμοποιούνται για να περιγράψουν την κατάσταση του διαύλου, σύμφωνα με προκαθορισμένες τιμές που περιγράφονται αναλυτικά στο datasheet του AVR. Τα δύο τελευταία bit αντιστοιχούν στον prescaler που χρησιμοποιείται για τον καθορισμό της συχνότητας του Clock. Οι πιθανές τιμές είναι: 0 -> 1, 1 -> 4, 2 -> 16, 3 -> 64.

TWDR: Είναι εγγράψιμος μονάχα εάν το bit TWINT είναι 1. Στον καταχωρητή αυτόν υπάρχει πάντα το τελευταίο byte που έχει μεταδοθεί στον δίαυλο.

TWAR: Στον καταχωρητή αυτόν αποθηκεύεται η διεύθυνση των 7 bit που αντιστοιχεί στον μικροελεγκτή και συγκρίνεται κάθε φορά που μεταδίδεται το σήμα START στον δίαυλο. Το τελευταίο bit χρησιμοποιείται για την ενεργοποίηση της αναγνώρισης της General Call Address από τον συγκεκριμένο μικροελεγκτή. Εάν η τιμή του είναι 1, τότε η συσκευή είναι προγραμματισμένη να απαντά με ACK, κάθε φορά που θα ακούει την διεύθυνση 0000000.

Ο AVR μπορεί να λειτουργήσει σαν Master Transmitter/Receiver, Slave Transmitter/Receiver. Για περισσότερες λεπτομέρειες και δείγματα κώδικα μπορείτε να δείτε το datasheet του AVR.