

1.4 פרוטוקולים, שכבות וארכיטקטורה של מערכות תקשורת

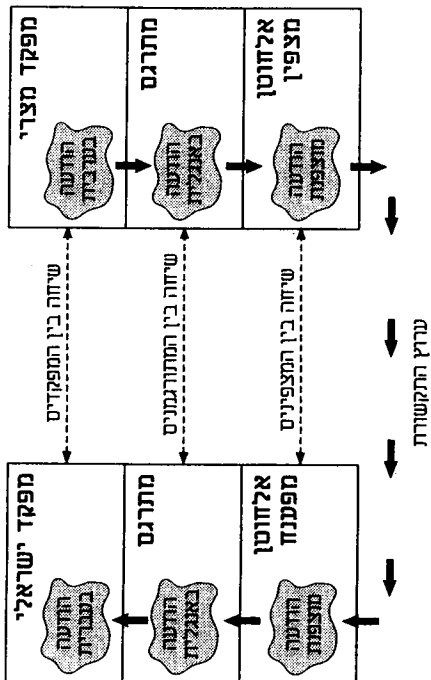
בשלושת הסעיפים הקודמים הצגנו מושגים בסיסיים בנושא תקשורת מחשבים, ודנו בסוגים שונים של רשתות תקשורת. בסעיף זה נפנה את תשומת הלב לתכנון של מערכות התקשורת. מערכות תקשורת הן סבוכות ביותר, ועיצובן מהווה אתגר הנדסי לא פשוט. בסעיף זה נכיר את הגישה לפיה מתכננים מערכות כאלה.

1.4.1 חלוקה לשכבות

כדי להפחית את הסיבוך הקיים בעיצוב ובתחזוקה של מערכות תקשורת, נהוג לחלקן לשכבות (layers). כל שכבה ממלאת תפקיד מסוים במערכת התקשורת ומספקת שירותי תקשורת מסוימים לשכבות שמעליה. המימוש של שכבה מתבצע לרוב על-ידי מודול תוכנה. בכל מחשב המהווה תחנת קצה ברשת, יש לממש את כל השכבות מהן מורכבת מערכת התקשורת. שכבה זו במחשב אחד מנהלת שיחה עם שכבה זו במחשב אחר. הכללים והמוסכמות המאפשרים לנהל שיחה כזו נקראים פרוטוקול.

פרוטוקול (protocol) הוא שפה משותפת או הסכם בין הצדדים המעורבים בתקשורת; תפקידו להסדיר את העברת המידע בין הצדדים.

נציג להלן דוגמה להמחשת המושגים 'פרוטוקול' ו'שכבה'. נניח ששני מפקדי כוח רב-לאומי שהופקד על שמירת השלום במזרח התיכון, מעוניינים לשוחח כדי לתאם את פעולותיהם. לרשות המפקדים עומדים מכשירי קשר, אך עליהם להתגבר על שתי בעיות: ראשית, מפקד אחד הוא מצרי שאינו דובר עברית, והשני ישראלי שאינו דובר ערבית. שנית, את ההודעות שעברות בקשר יש להצפין כדי שגורמים עוינים לא יוכלו לצותת להם. אורך 1.10 מתאר כיצד אפשר לפתור בעיות אלה בעזרת מתורגמנים ומצפנים.



איור 1.10 מערכת תקשורת להעברת הודעות נכה רב-לאומי

נניח שהמפקד המצרי רוצה להעביר מסר למפקד הישראלי. לשם כך הוא מעביר הודעה בערבית למתרגם שלו, המתרגם מתרגם את ההודעה מערבית לאנגלית ומעבירה לאלחוטאי המצפין. האלחוטאי מצפין את ההודעה ומשדר אותה. ההודעה נקלטת על-ידי האלחוטאי הישראלי שמפענח אותה. לאחר הפענוח מתקבלת הודעה באנגלית שמועברת למתרגם הישראלי. הוא מתרגם את ההודעה מאנגלית לעברית, ומוסר אותה למפקד הישראלי.

במערכת תקשורת זו יש שלושה סוגי תהליכים: תהליכים פיקודיים, תהליכי תרגום ותהליכי הצפנה. לכן נוה לחשוב על מערכת זו כעל מערכת המורכבת משלש שכבות, כפי שמתואר באיור 1.11. באיור מתוארות השכבות האלה: שכבת הפיקוד הכוללת את שני המפקדים, שכבת התרגום הכוללת את המתורגמנים ושכבת ההצפנה הכוללת את האלחוטנים המצפנים.

- השיחה בין היישויות שבכל שכבה נעשית לפי פרוטוקול מתאים:
- המפקדים משוחחים בי'רגון צבאי; הפרוטוקול שלהם כולל מונחים מתחום זה.
 - המתורגמנים משוחחים ביניהם באנגלית, לכן הפרוטוקול המוסכם עליהם הוא השפה האנגלית.
 - הפרוטוקול של האלחוטנים הוא שפת הצופן שבה הם משתמשים להצפנת ההודעות.

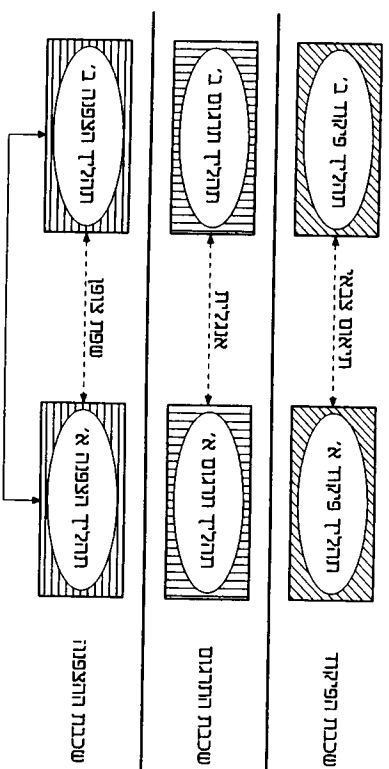
במערכת המחולקת לשכבות, המידע לא מועבר ישירות בין תהליכים עמיתים (כלומר תהליכים באותה שכבה), אלא זורם אנכית: בצד השולח זורם המידע כלפי מטה, ממפקד למתרגם וממתרגם לאלחוטן; בצד המקבל זורם המידע כלפי מעלה, מהאלחוטן למתרגם וממתרגם למפקד. רק בין האלחוטנים קיימת העברה פיזית של מידע בערוץ הפיזי (האופק) המקשר ביניהם.

עם זאת, כל מפקד יכול להניח שהוא משוחח ישירות עם עמיתו, שהרי דבריו מכוונים אל המפקד העמית ולא אל המתרגם: הוא פונה אל המפקד העמית ומצפה לקבל את תגובתו. לכן נוח לחשוב כאילו קיים ערוץ אופקי המקשר ישירות בין המפקדים העמיתים. ערוץ זה לא קיים במציאות, אך המפקדים יכולים לדמות כאילו הוא קיים. לכן הוא נקרא **ערוץ מדומה** (virtual channel). באופן דומה, שני המתורגמנים משוחחים זה עם זה באנלית. באופן פיזי, המידע מועבר גם במקרה זה, באמצעות האלחוטנים, אך למרות זאת, המתורגמנים יכולים לדמות שהם משוחחים ביניהם ישירות באמצעות ערוץ מדומה.

על-פי-דבר, כל תהליך מוסיף להודעה מידע בקרה, המיועד לחיאים עם התהליך העמיתי, ומאפשר לשני התהליכים לתאם את פעולתם. מידע זה לא יועבר כלפי מעלה לשכבה שמעל. לדוגמה: נניח שהאלחוטנים צריכים לשנות מודי פעם את פרוטוקול החצפה שלהם, כדי להקשות את השיפוט. כאשר האלחוטן הישראלי יזום החלפה של פרוטוקול החצפה, הוא יוסיף מידע מתאים להודעה שבה הוא מטפל, למשל, את שם הצופן שאליז יש לעבור. מידע זה יוסיף לאלחוטן המצרי והוא לא יועבר על-ידו כלפי מעלה, למתרגם המצרי.

אור 1.12 מביא תמונה מופשטת יותר של האיוורים הקודמים; מוצגים בו שני המושגים שכבה ופרוטוקול שהצגנו לעיל, ומושג נוסף: מיישק (interface).

כתלק מהארכיטקטורה של מערכת תקשורת, מוגדר מיישק בין כל שתי שכבות סמוכות, הקובע את אופן התקשורת ביניהן. המיישק בין השכבות מוגדר באמצעות אוסף של פעולות אטומיות או פרימיטיבים (primitives). אלה הן פרוצדורות שמציעה כל שכבה לשכבה שמעליה. כאשר שכבה n רוצה להשתמש בשירות של השכבה שממתחתיה (שכבה $n-1$), עליה להפעיל פעולה אטומית של שכבה $n-1$. הפעלת הפעולה האטומית תיעשה על-ידי קריאה לפרוצדורה מתאגמה, הכלולה במיישק שבין שכבה $n-1$ לשכבה n .



אור 1.11 חלוקת המערכת לשכבות

מה היתרון בחלוקת המערכת לשכבות?

הבעיה הסבוכה נחלקת לכמה בעיות קטנות יותר; כל שכבה מתמודדת עם חלק מהן. השכבות הנמוכות מטרות את השכבות הגבוהות מלעסוק במיטלים שכבר טופלו. תכנון בשכבת הפיקוד בדוגמה שלנו.

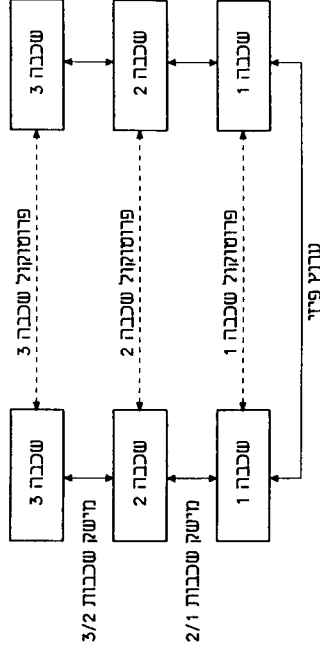
שני המפקדים אינם צריכים לדעת דבר על תהליכי התרגום והחצפה; בשבילים חשוב רק שההודעות שלהם יועברו באמינות (כלומר ללא שיבושים או השמטות). הם אינם צריכים להכיר את פרוטוקול שכבת התרגום (כלומר, את השיטה המשותפת למתורגמנים) או את פרוטוקול שכבת החצפה (שפת הצופן). אם שני המתורגמנים יחליטו להחליף את השיטה האנלית בשפה אחרת, למשל צרפתית, הם לא יצטרכו להודיע על כך למפקדים. השירות שהמפקדים מקבלים לא ישתנה כלל (כמובן בהנחה שהמתורגמנים אכן יודעים צרפתית). באופן דומה, פרוטוקול החצפה לא צריך להיות ידוע למתורגמנים או למפקדים, והחלפת הפרוטוקול לא תפריע לפעולתם.

קבוצה של שכבות והפרוטוקולים שלהן מכונה **ארכיטקטורה של מערכת התקשורת**. ארכיטקטורה של שכבות מקטינה את הסיכוד הקיים בעיצוב מערכת התקשורת, משיח שהיא מאפשרת לעצב כל שכבה בנפרד. בדרך זו, בעיה הנדסית סבוכה מחולקת לכמה בעיות פשוטות יותר.

התקשורת מותבצעת בין שני תהליכי משתמש, המורצים במחשבים שונים; בכל מחשב יכולים לרוץ כמה תהליכי משתמש בו-זמנית; המחשבים מחוברים באמצעות רשת תקשורת; רשת התקשורת מורכבת מערוצים המחברים בין המחשבים; ולבסוף, האותות נושאי המידע מתפשטים בערוצי התקשורת.

לשם המחשת הדברים, נדון להלן ביישום של העברת קבצים, ונתאר את המשימות העיקריות שמערכת התקשורת צריכה למלא כדי לממש יישום כזה.

- **רמת התהליכים.** תוכנת התקשורת של התתנה השולחת צריכה לודא שתהליך ניהול הקבצים במחשב היעד ערוך לקלוט את הקובץ, לאחסנו במערכת הקבצים של מחשב היעד ולתת לו שם – כפי שביקש המשתמש. בנוסף, אם קיימים הבדלים בצורת אחסון הקבצים בין שתי המערכות, אזי באחד משני המחשבים צריך לרוץ תהליך הממיר את הקובץ למבנה הרצוי. למשל: במחשבים גדולים מקובל לאחסן את הבית הראשון בכל מילה, בכתובת נמוכה יותר מאשר את הבית השני של המילה. במחשבים קטנים מקובלת המוסכמה ההפוכה. כדי לשמור על משמעות הנתונים בהעברה כזו, יש להפוך את סדר הבתים בכל מילה.
- **רמת המחשבים.** התוכנה של המחשב השולח צריכה לספק לרשת התקשורת את כתובתו של מחשב היעד ולהבטיח שמחשב זה מוכן לקלוט נתונים, לנהל את העברת הנתונים ולודא שהנתונים מועברים באמינות.
- **רמת הרשת.** רשת התקשורת צריכה לנתב את הנתונים שמעביר המחשב השולח, דרך צומתי המיתוג של הרשת עד שיוגיע למחשב היעד. הרשת צריכה לנסות למצוא נתיבים קצרים ולדאוג שהעומס על הרשת יתחלק על פני כל הרשת באופן שווה ככל האפשר, כדי שלא ייווצרו 'פקקי תנועה'.
- **רמת הערוץ.** בעת העברת הנתונים דרך ערוצי התקשורת צריכים לשמור על תוכן המידע המועבר. לשם כך יש לודא ששייאות הגרמות עקב העשים לא ישבשו את המידע. דבר זה מחייב לגלות את קיומן של שגיאות ולהתגבר עליהן בדרך כלשהי.



איור 1.12 הארכיטקטורה של מערכת התקשורת בין המפקדים

שינוי המימוש של שכבה $n-1$ לא ישפיע על שכבה n (ובדאי לא ישפיע על שכבות אחרות) אם הוא לא יגרום לשינוי הפעולות האוטומיות המוצעות על-ידי שכבה $n-1$. לכן מעצבי המערכת יכולים לשנות בעת הצורך את מימוש השכבות, בלי שהדבר יגרום לזעזועים במערכת כולה.

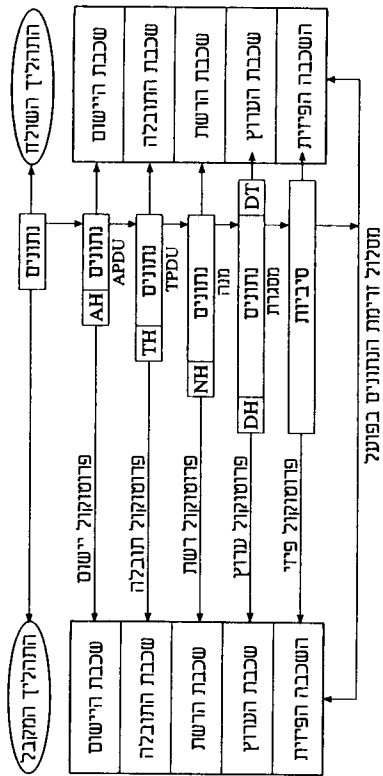
1.4.2 דוגמה לארכיטקטורה של מערכת תקשורת

כעת, אחרי שהצגנו את המושגים הבסיסיים הקשורים לארכיטקטורה של מערכת תקשורת, נעבור להצגת דוגמה למודל אחד של ארכיטקטורה כזו, שבו המערכת מורכבת מחמש שכבות. בסעיף הבא נסקור בקצרה שני מודלים מציאותיים: מודל OSI הכולל שבע שכבות ומודל TCP/IP הכולל ארבע שכבות.

הממוש של יישומי התקשורת שהצגנו בסעיפים הקודמים אינו פשוט, כפי שעשוי אולי להיראות לאדם ה"גולש" ברשת בעזרת תוכנת שיטוט (browser). למעשה, מערכת תקשורת היא מורכבת ביותר ועיצובה מחייב נקיטת גישה הנדסית מבנית. כפי שראינו, ארכיטקטורה של שכבות מאפשרת לחלק את המערכת ליחידות או למודולים, שכל אחד מהם נשען על השירותים שמספקים מודולים אחרים, אך אינו תלוי בפרטי המימוש של מודולים אלה. כאשר מתארים מערכת תקשורת, מתרכזים בחמישה סוגי עצמים:

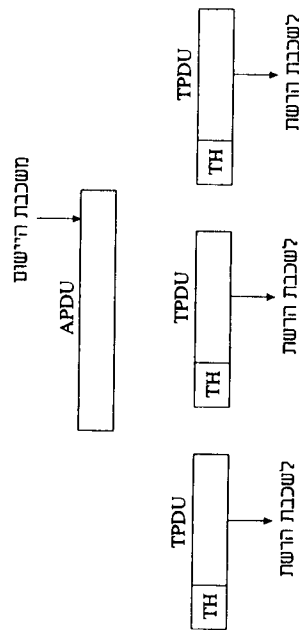
תהליכים, מחשבים, רשתות, ערוצים, אותות

יחידה זו מועברת לשכבת התובלה, המוסיפה לה כותרת, שנשמנה באותיות TH (קיצור של Transport Header). נדגיש כי עבור שכבת התובלה, APDU הוא נתונים, והתבונה בין הנתונים שהעביר המשתמש לכותרת היישום (AH) אינה חשובה.



איור 1.14
כותרות ויחידות נתונים בארכיטקטורה של המש שכבות

ברוב הרשתות יש הגבלה על אורך היחידות שיכולה לקבל שכבת התובלה צריכה לעתים לחלק את ההודעה (APDU) שמגיעה אליה לחלקים קטנים יותר, וליצור מכל חלק יחידה למשלוח (TPDU) על-ידי הוספת כותרת. חלוקה זו מתוארת באיור 1.15.

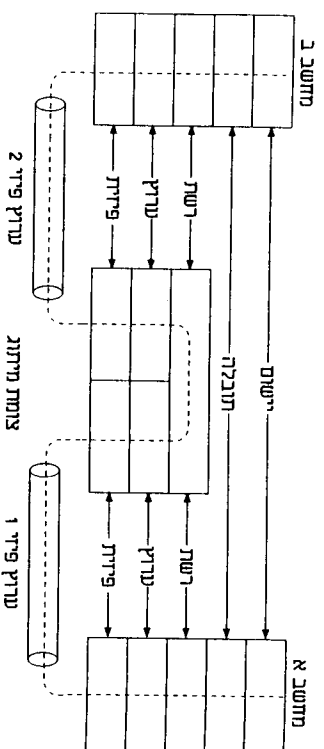


איור 1.15
חלוקה של APDU לחלקים, ויצירת יחידות TPDU

השכבה	השירות המסופק לשכבה שמעליה	תפקיד עיקרי או סיבה לקיום השכבה במודל
5. יישום	מקורות להפעלת יישומי תקשורת ולביצוע מטלות תקשורת (מסופק למשתמש הקצה)	ממוש הלוגיקה המיוחדת לכל יישום. למשל: טיפול בשמות קבצים, העברה של סיסמאות, שיגור וקבלה של דואר אלקטרוני.
4. תובלה	העברה אמينة של נתונים בין שני מחשבים המחברים לרשת	המנגנונים להבטחת תקשורת אמينة (חלקם ייסקרו בפרק 3) אינם תלויים באופי של כל יישום, לכן הגיוני לרכזם בשכבה כללית אחת.
3. רשת	העברה של נתונים ליעד מבוקש, דרך רשת תקשורת	התכנה של שכבות 4 ו-5 אינה צריכה להיות תלויה במבנה ובטכנולוגיה של הרשת. לכן הגיוני שתהיה שכבה שתספק שירות אחיד, ללא תלות בפרטים של הרשת המקשרת בין המחשבים ברגע נתון.
2. ערוץ	ערוץ תקשורת אמין	ניהול הרשת אינו קשור לגינת לטיפול באמינות העברת המידע בערוצי התקשורת, לכן הגיוני שתהיה שכבה שתתגבר על שגיאות ותבטיח העברה אמينة של נתונים בין שני קצוות של ערוץ תקשורת.
1. פיזית	ערוץ תקשורת פיזי (צינור להעברת סיביות)	המנגנונים של גילוי שגיאות והתגברות עליהן אינם תלויים בסוג התווך ובאופי האותות המורדרים בו, לכן הגיוני שתהיה שכבה שתעסוק בשימוש בערוץ הפיזי, כלומר תהיה אחראית על העברה של סיביות מקצה אחד לקצה שני של ערוץ תקשורת.

כל שכבה במודל מוסיפה כותרת להודעה שהיא מקבלת; הכותרת מכילה מידע בקרה המועד לתהליך העמית במושב היעד. איור 1.14 מתאר את הארכיטקטורה בלויית הכותרות.

כל תחת קצה ברשת מכילה חמש שכבות. התקשורת מתבצעת בין שני תהליכי משתמש. התהליך השולח מעביר את הנתונים לשכבת היישום שמוסיפה לתחילת ההודעה **כותרת יישום**; נסמן כותרת זו באותיות AH (קיצור של Application Header). היחידה הנוצרת נקראת **יחידת הנתונים של פרוטוקול היישום** או APDU (Application Protocol Data Unit).



איור 1.16 ארכיטקטורה של שכבות עם צומת מיתוג ברשת

איור 1.17 מתאר באופן מפורט שלבים טיפוסיים בשידור של הודעה, ממחשב שולח למחשב מקבל, בתצורה המתוארת באיור 1.16. עבור על האיור, החל מהפינה השמאלית העליונה, לפי המיספור. באיור מתוארים 19 שלבים של טיפול בהודעה, החל מקידוד המסוף של המשתמש במחשב השולח, עד מסיירתה להחליק של המשתמש במחשב המקבל. בפעל, ההודעה תעבור בדרך-כלל דרך צומתי מיתוג רבים עד שתגיע ליעדה; לשם הפשטות כללנו באיור רק צומת מיתוג יחיד.

כדי ששכבת התובלה של הצד המקבל תוכל להרכיב בחזרה את ההודעה המקורית יש למספר את היחידות (ה-TPDU) של שכבת התובלה. מספר ה-TPDU הוא זוגמה למידע בקרה שמופיע בכותרת של שכבת התובלה (TH).

שכבת הרישת מקבלת משכבת התובלה TPDU ומסיפה לו כותרת NH (Network Header). היחידה המתקבלת נקראת מנה (packet), המנה מועברת לשכבת הערוץ, שמוסיפה לה כותרת (DH) וסיימת (DT - Data-link Trailer). הסיימת דרושה בשכבה זו כדי לאפשר לשכבה הפיזית של מחשב היעד לזהות היכן מסתיימת המסגרת. היחידה שיוצרת שכבת הערוץ נקראת **מסגרת** (frame). המסגרת מועברת לשכבה הפיזית, שמתייחסת אליה כאל זרם של סיביות המשודר בערוץ הפיזי.

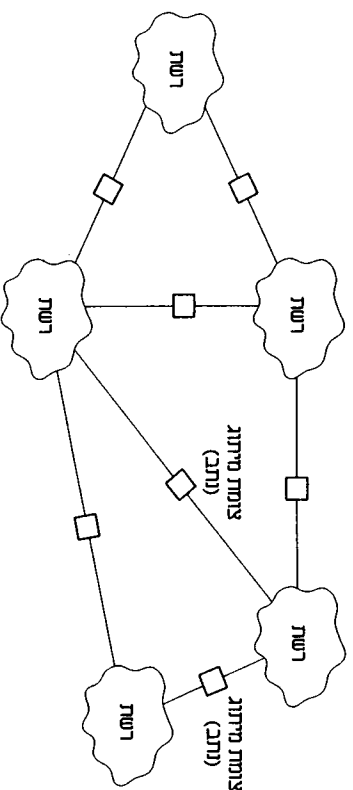
במחשב היעד מתרחש תהליך הפוך. השכבה הפיזית קולטת את זרם הסיביות, מזהה את גבולות המסגרת ומעבירה את המסגרת לשכבת הערוץ. שכבת הערוץ בוחנת את כותרת הערוץ (DH), מסיירה אותה ומעבירה את המנה שבתוך המסגרת לשכבת הרישת. פעולה זו מתרחשת בכל השכבות, כל שכבה בוחנת את הכותרת המתאימה, מסיירה אותה ומעבירה את ההודעה כלפי מעלה. בסוף התהליך יגיע הנתונים המקוריים ששלח התהליך השולח אל התהליך המקבל.

שכבות 4 ו-5 של המודל אינן קשורות לרישת התקשורת, לכן צומתי המיתוג של הרישת אינם צריכים להכיל שכבות אלה. איור 1.16 מציג את הארכיטקטורה שתוארה, בשילוב עם צומת מיתוג ברשת תקשורת. המנות הנוצרות במחשב השולח, צריכות לעבור דרך צומתי בניינים בדרך למחשב היעד. בכל אחד מצומתי הבניינים מוממשות השכבות 1 עד 3. באיור מתואר חיבור שני מחשבים דרך צומת מיתוג יחיד. בצומת יש שתי שכבות ערוץ ושתי שכבות פיזיות, אחת עבור כל ערוץ המקשר את הצומת למחשב. כל שכבת ערוץ (ושכבה הפיזית) פועלת באופן עצמאי ומספקת שירותים לשכבת הרישת בערוץ המתאים. הפרוטוקולים של שכבת התובלה ושכבת היישום פועלים מייקצה-לקצה" בין שני המחשבים.

תקשורת ברשת-על, כלומר, ברשת המורכבת מאוסף של רשתות שהמבנה שלהן והפרוטוקולים התחוגים בהן אינם בהכרח אחידים. גישה זו הוכחה כיעילה ואיפשרה את הגידול המסחרר של רשת האינטרנט. כיום, מודל TCP/IP נפוץ במידה רבה לאין שיעור ממודל OSI. זאת למרות שמודל OSI תוכנן על-ידי מכון התקנים הבנלאומי (ISO) במטרה להוות תקן בינלאומי.

בהמשך נסקור את התפתחות רשת האינטרנט ונתאר את היישומים הקיימים בה, אולם כדי שנוכל לדון במודל TCP/IP, שעיבורו כאמור, חיבור בין רשתות, עלינו לפתוח בתיאור קצר של מבנה האינטרנט. איור 1.18 מתאר את רשת האינטרנט כאוסף של רשתות מקומיות או ארוכות-טווח, המקושרות על-ידי מחשב מיתוג הנקראים **נתבים (routers)**. כאשר מחשב המחובר לאחרת מן הרשתות המורכבות את רשת האינטרנט מעוניין לשלוח הודעה למחשב המחובר לרשת אחרת, עליו להפנות את ההודעה אל אחד מהנתבים. כאשר ההודעה מגיעה אל נתב, הוא מעביר אותה דרך רשת הנתבים, שהיא השלד של האינטרנט, עד הגיעה לנתב המחובר לרשת היעד. נתב זה ישלח את ההודעה לנתב רשת היעד ומשם היא תועבר אל מחשב היעד.

ההבדל בין מודל TCP/IP למודל OSI הוא: ב-TCP/IP מניחים שהשכבה התחתונה היא רשת ולא ערוץ; מודל TCP/IP אינו כולל על-כן שכבת ערוץ ושכבה פיזית.



איור 1.18
מבנה כללי של רשת האינטרנט

בהשוואה עם המודל בן חמש השכבות שהצגנו בסעיף הקודם, אפשר לראות ששכבות 1 עד 4 ושכבה 7 הן השכבות המרכזיות את המודל שהצגנו. שמות שכבות אלה ותפקידיהן במודל שלנו, נלקחו למעשה מתוך מודל OSI. שכבות השיחה והייצוג לא נכללו במודל שהצגנו. הסיבה לכך היא שבתלך גדול ממערכות התקשורת, הפועלות לפי OSI, לא ממומשות שכבות השיחה והייצוג ותפקידיהן מבוטלים בעצם על-ידי שכבת המשוב. נסקור כאן בקצרה את תפקידי שכבת השיחה ושכבת הייצוג.

תפקיד **שכבת השיחה** במודל OSI הוא לספק בקרה לתקשורת בין יישומים. שכבה זו מקימה, מנהלת ומסיימת קשרים (שירות) בין יישומים. אחד מהתפקידים הקשורים בכך הוא אספקת מנגנון של התאוששות מתקלות במערכת התקשורת. למשל, כאשר השיחה בין היישומים מיועדת להעברת קובץ גדול, שכבת השיחה יכולה להכניס לתוך הקובץ נקודות בדיקה (checksum). אם העברת הקובץ משתבשת מסיבה כלשהי ונקטעת, שכבת השיחה של מחשב היעד אינה צריכה לדווח שידור חוזר של הקובץ כולו, אלא רק של הנתונים המופיעים אחרי נקודת הבדיקה האחרונה שהתקבלה אצלה. **שכבת הייצוג** במודל OSI ממלאת שלוש תפקידים: תמרות בין ייצוגים שונים של נתונים, דחיסת נתונים והצפנה. נושאים אלה הם חשובים ומעניינים, אך הם חורגים ממסגרת ספר זה.

מודל OSI, למרות היותו מבוסס על תקן בינלאומי של ISO, לא זכה לתפוצה רחבה. רשת האינטרנט אינה מבוססת על מודל זה אלא על מודל TCP/IP (שיחואר מילד). בשלושת הפרקים הבאים של הספר נתאר ביתר פירוט את שלוש השכבות התחתונות של מודל OSI. פרק 5 יעסוק בתת-שכבה של שכבת הערוץ הקשורה למימוש רשתות מקומיות. השכבות הגבוהות יותר של מודל OSI לא ידועו בהמשך ספר זה.

מודל TCP/IP

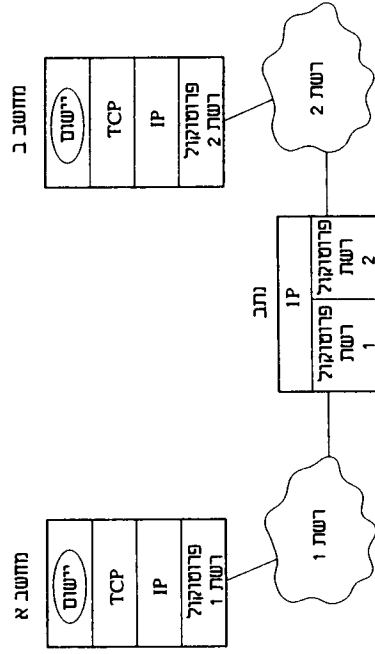
מודל TCP/IP פותח כחלק מרשת ARPANET שהוקמה ביוזמת משרד ההגנה האמריקאי. רשת ARPANET הפכה ברבות הימים לרשת האינטרנט המפורסמת, משום כך מודל TCP/IP משמש כבסיס לרשת זו. אוסף הפרוטוקולים המקורי שפותח, כלל המישה פרוטוקולים. המודל קרא על שם שני הפרוטוקולים החשובים, שהם IP (קיצור של Internet Protocol) ו-TCP (קיצור של Transmission Control Protocol). מודל TCP/IP פותח מתוך כוונה לאפשר

מתפקיד זה, צריכה שכבה זו לטפל בכתובות ולנתב את המברקים בין הנתבים המתבררים בין הרשתות, עד לרשת המכילה את מחשב היעד. הפרוטוקול של שכבה זו נקרא IP.

שכבת **התובלה** מבטיחה שהמיידע, המועבר מקצה לקצה, יועבר באמינות; לשם כך צריכה שכבת התובלה של מחשב היעד, לאסוף את המברקים המגיעים אליה, לוודא שכולם הגיעו ללא שגיאה ולהרכיב מהם את ההודעה המקורית. הפרוטוקול העיקרי של שכבה זו נקרא TCP. קיים פרוטוקול נוסף, פשוט יותר שנקרא UDP, המספק שירות לא אמיני.

לבסוף, שכבת **היישום** כוללת פרוטוקולים עבור היישומים השונים העומדים לרשות המשתמשים. באופן מקורי כללה שכבה זו שלושה פרוטוקולים עיקריים: FTP (פרוטוקול העברת קבצים, SMTP (פרוטוקול דואר אלקטרוני) ו-TELNET (פרוטוקול כניסה למחשב מרוחק). בהמשך הפרק, כשנזון ברשת האינטרנט, נתאר יישומים אלה ביתר הרחבה.

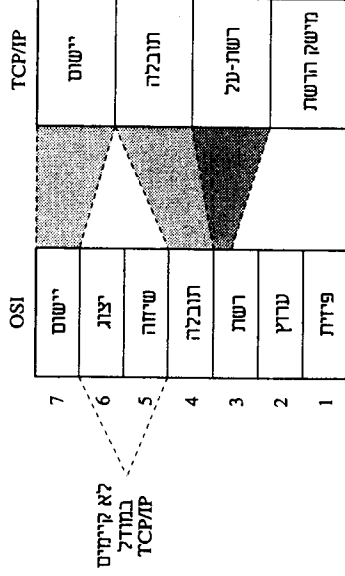
איור 1.20 מתאר את התצורה שבה פועלים פרוטוקולים אלה ברשת-העל. פרוטוקול רשת כלשהו קיים בכל רשת, והוא מאפשר למחשבי הקצה לשלוח הודעות למחשב יעד, לשלוח הודעות בתוך הרשת המקומית אליה הם משתייכים, או, כאשר מחשב היעד מצוי ברשת אחרת, אל נתב. פרוטוקול IP ממומש בכל מחשבי הקצה והנתבים, והוא מאפשר להעביר מברקי נתונים ממחשב קצה, דרך נתבים אחדים, עד מחשב היעד. פרוטוקול TCP ממומש רק במחשבי הקצה.



איור 1.20 תצורה באמצעות הארכיטקטורה של מודל TCP/IP

- הארכיטקטורה של TCP/IP כוללת ארבע שכבות:
- שכבת מישק הרשת (host-to-network layer);
 - שכבת רשת-העל (internet layer);
 - שכבת התובלה (transport layer);
 - שכבת היישום (application layer).

ההקבלה בין שכבות אלה לבין השכבות של מודל OSI מתוארת באיור 1.19.



איור 1.19 השכבות של מודל TCP/IP מול השכבות של מודל OSI

תפקידה של שכבת **מישק הרשת** הוא לגשר על-פני ההבדלים הקיימים בין הרשתות השונות שמהם מורכבת רשת-העל, כך לא יצטרכו השכבות הגבוהות יותר להיות מודעות לשוני ויכולו לראות את כל האינטרנט כאילו זו רשת אחת, בעלת מבנה אחיד. בכל אחת מהרשתות המחוברות לרשת-העל שכבת מישק הרשת היא שונה; היא מכירה את מבנה המנות או המסגרות של אותה רשת ותפקידה להמיר מנות אלה למבנה אחיד, הנקרא **מברק IP** (IP datagram). במודל TCP/IP לא מפורט פרוטוקול כלשהו לשכבה זו, אלא משתמשים בפרוטוקול המתאים, לפי הרשת שאליה רוצים להתחבר.

שכבת **רשת-העל** היא השכבה שימדייקה; יחד את כל הרשתות המרכיבות את האינטרנט. תפקיד שכבה זו הוא לאפשר העברה של מברקי IP דרך רשתות שונות, עד שגיעו ליעדן. כחלק

נניח שיישום כלשהו במחשב A מעוניין לשלוח הודעה ליישום במחשב B, הנמצא ברשת אחרת. שכבת היישום של מחשב A מעבירה את ההודעה ל-TCP. פרוטוקול TCP מחלק את ההודעה ליחידות, הנקראות כאמור, מברקי IP, ומעביר כל מברק לפרוטוקול IP, בצירוף הנחיה לשלוח את המברק למחשב B.

פרוטוקול IP קובע לאיזה נתב יש לשלוח את המברק, ומעביר את המברק לשכבת המישק לרשת. שכבה זו יוצרת מהמברק מנהל המראים לרשת 1 ושבו מברק ה-IP הוא שדה המידע, ומעבירה אותו לפרוטוקול הרשת של רשת 1, שמעביר אותו לנתב. הנתב מסיר את הכותרת של שכבת מישק הרשת, ופרוטוקול IP בודק את תוכן המברק, כדי לקבוע לאן צריך לשלוח אותו. במקרה שלט המברק מגיע לרשת 2, ושם נשלח אל מחשב B. כאשר שכבת IP במחשב B מקבלת את המברק, היא מעבירה אותו ל-TCP. פרוטוקול TCP אוסף את המברקים המגיעים ומרכיב מהם את ההודעה המקורית, המועברת לשכבת היישום של מחשב B.

1.4.4 סיוכום מונחים

- **שכבה** היא מודל של מערכת תקשורת. כל שכבה (פרט לשכבה הפיזית שהיא התחתונה ביותר) מתבססת על השירותים שמספקת השכבה שמתחתיה. כל שכבה (פרט לשכבת היישום שהיא העליונה ביותר) מספקת שירותי תקשורת לשכבה שמעליה.
- **תהליכים עמיתים** הם תהליכים השייכים לאותה שכבה ומורצים במחשבים שונים השייכים לרשת.
- **פרוטוקול** הוא הסכם בין צדדים המעורבים בתקשורת לגבי אופן ניהול התקשורת. לכל שכבה במודל שכבות יש פרוטוקולים לפיהם מתקשרים ביניהם התהליכים העמיתים.
- **ארכיטקטורה של מערכת תקשורת** היא קבוצה של שכבות והפרוטוקולים שלהן.
- **ערוץ מוזמה** (או **וירטואל**) הוא ערוץ תקשורת בין שתי שכבות עמיתות. ערוץ זה אינו קיים באופן פיזי אלא ממומש באמצעות התוכנה של השכבות הנמוכות יותר (וכמוכן בעזרת הערוץ הפיזי).
- השירותים שמספקת כל שכבה שמעליה, מוגדרים באמצעות אוסף **פרימיטיבים** או **פעולות אטומיות**, שהן פרוצדורות השייכות למישק של השכבה.
- **מודל OSI** או **מודל המערכות הפתוחות** הוא מודל שפותח על-ידי ISO (מכון התקנים הבינלאומי) במטרה ליצור תקני תקשורת בינלאומיים.

3.2 פרוטוקול עצור-והמתן

בסעיף זה נציג פרוטוקול בסיסי של שכבת הערוץ הנקרא פרוטוקול עצור-והמתן. נפתח את הפרוטוקול בשלישה שלבים, החל מפרוטוקול פשוט, המספק שירות לא-אמין, ללא אישורים, וכלה בפרוטוקול מורכב, המספק שירות אמין. לאחר-מכן נדון ביעילותו של פרוטוקול עצור-והמתן ונסביר באילו תנאים פרוטוקול זה אינו יעיל. בסעיף הבא נציג פרוטוקולים מתקדמים יותר, המאפשרים להגמבר על בעיות היעילות של פרוטוקול עצור-והמתן.

מערכת התקשורת שעוסק בה בסעיף זה, מורכבת משני צמתים המחוברים ביניהם בערוץ נלין. בשלב ראשון נדון בפרוטוקול חזר-כיווני, שבו רק צד אחד משדר מידע. למען הקיצור נקרא לתחנה המשדרת את המידע - תחנה א, ולתחנה שאליה מיועד המידע נקרא תחנה ב.

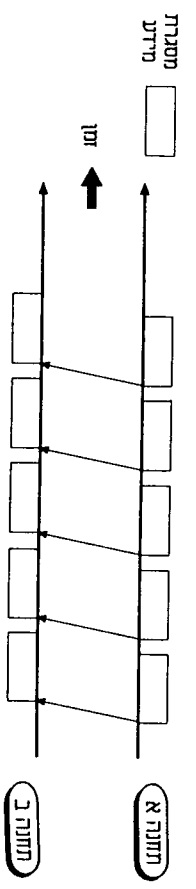
3.2.1 שירות לא-אמין, ללא אישורים

נפתח בדרישה פשוטה:

דרישת ואשונה: המערכת תאפשר לתחנה א להעביר נתונים לתחנה ב.

שימו לב, לא דרשנו שהעברת הנתונים תהיה אמينة ושהשולח יקבל אישור על תקינות העברה; כלומר, בשלב זה אנו דורשים מערכת שתספק שירות לא-אמין, ללא אישורים.

תצורת המסגרות לאורך ציר הזמן מתוארת באיור 3.14.



איור 3.14 תצורה של מסגרות בין שתי תחנות בפרוטוקול ללא אישורים

{פרוטוקול 1: שירות לא-אמין וללא אישורים}

תחנה שולחת

בעצ לולאה (אינסופית):

קבל מנה משכבת הרשת

צור מסגרת לשידור

העבר את המסגרת לשכבה הפיזית

סוף הלולאה.

תחנה מקבלת

בעצ לולאה (אינסופית):

{האירוע היחיד האפשרי הוא הגעת מסגרת }

המתן לאירוע (הגעת מסגרת)

קבל מסגרת מהשכבת הפיזית

בודד את שדה המידע (המנה)

העבר את שדה המידע לשכבת הרשת

סוף הלולאה.

היתרון של פרוטוקול זה הוא **מוחזרות** שבה מועברים הנתונים – אין המתנה לאישור ואין שידורים חוזרים; לכן פרוטוקול זה מתאים (בתנאים מסוימים) לשידור מידע קולי, או וידאו. ייתכן למשל, שמאזין יעדיף לקבל מידע קולי ללא עיכובים, גם במחיר של שיבושים מסוימים בקבלת המידע.

אבל, במרבית היישומים של תקשורת מחשבים דרושה רמת אמינות יותר גבוהה מזו שמספק פרוטוקול 1. הצד השולח מעוניין לדעת לפחות אם המידע ששלח הגיע לתעודתו. נסח אפוא דרישה נוספת שהמערכת תצטרך למלא.

נתאר פרוטוקול שכבת ערוץ שתפקידו לענות על הדרישה שהגדרנו. פרוטוקול זה יהיה הראשון בסדרה של פרוטוקולים, לכן ייקרא פרוטוקול 1.

למען הפשטות, נניח בהמשך כי לשכבת הרשת של תחנה א יש תמיד מנות מוכנות לשידור, ולכן שכבת הערוץ יכולה לפנות אליה בכל עת, כדי לקבל מנה.

נניח כי התוכנה של שכבת הערוץ מכילה את הנהלים האלה:

1. נוהל בשם המתן אשר מאפשר לה להתמתן להתרחשות של אירוע כלשהו. נרשום בסוגריים את האירועים להם ממתנינים. למשל, המשפט "המתן (הגעת מסגרת)" גורם להתמתן (עד שמגיעה מסגרת). בשלב זה האירוע האפשרי היחיד הוא הגעה של מסגרת נתונים מהשכבה הפיזית לשכבת הערוץ של תחנה ב, אך בהמשך יהיו אירועים נוספים.
2. נוהלים לקבלת מנה משכבת הרשת (בתחנה א) ולהעברה של מנה לשכבת הרשת (בתחנה ב).
3. נוהלים לקבלת מסגרת מהשכבה הפיזית (בתחנה ב) ולהעברה של מסגרת לשכבה הפיזית (בתחנה א).

נשתמש בנהלים אלה כדי לתאר בעברית מבנית את האלגוריתם של פרוטוקול 1. (קרא את הפרוטוקול בעמוד הבא).

שתי שכבות הערוץ מבצעות לולאה אינסופית. תחנה א מקבלת בכל מחזור של הלולאה מנה נוספת, יוצרת ממנה מסגרת ומעבירה אותה לשידור. תחנה ב ממתינה להגעת המסגרת, קולטת אותה, ומעבירה את המנה שבתוכה לשכבת הרשת. התחנה לא בודקת אם המסגרת שובשה, ולא מאשרת את קבלתה.

פרוטוקול 1 עונה על הדרישה הראשונה שהצגנו לעיל; הוא מאפשר להעביר מידע מתחנה א לתחנה ב. בפועל, פרוטוקול זה מתאים רק למערכת תקשורת שתחת העד שלה מסוגלת לקלוט ולעבד את כל המסגרות המגיעות אליה, בקצב הגעתן, ושבה אפשר להתעלם משיבושים במידע המועבר. אם תנאים אלה אינם מתקיימים – מסגרות אחדות יאבדו, ותחת המקור לא תקבל שום מידע על כך.

3.2.2 שירות לא-אמין עם אישורים

דרישה שנייה: המערכת תאפשר העברת נתונים מתחנה א לתחנה ב ותזווח לשולח אם הנתונים התקבלו ביעד באופן תקין.

כדי למלא דרישה זו, שכבת הערוץ צריכה לספק לשכבת הרשת שירות עם-אישורים. שכבת הערוץ תנסה להעביר בהצלחה את כל המסות; היא לא תבטלה שכל מנה תעבר ללא שיבוש, אבל היא תדווח לשכבת הרשת אם המנה הועברה בהצלחה.

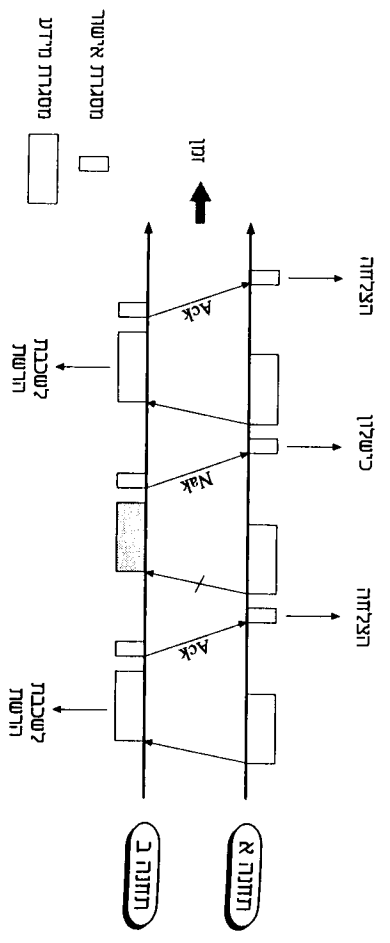
נתאר כעת פרוטוקול למילוי הדרישה השנייה. פרוטוקול זה, שייקרא פרוטוקול 2, יפעל כדלקמן: בכל פעם ששכבת הערוץ של תחנה ב מעבירה מנה לשכבת הרשת שמעליה ומתמנה לטיפול במנה נספת, היא תשלח מסגרת תגובה מתאימה לתחנה א. שכבת הערוץ של תחנה א תמתין להגעת התגובה ורק לאחר מכן תשלח מסגרת נתונים נוספת, ותדווח לשכבת הרשת על תוצאות הפעולה. פרוטוקול מסוג זה נקרא פרוטוקול "עצור-והמתן" (stop-and-wait) או "שליח-והמתן" (send-and-wait) כי לאחר משלוח כל מסגרת תמתן המקור עוצרת וממתנת*.

מסגרת התגובה היא **מסגרת בקרה** (control frame). מסגרות בקרה מאפשרות להתלכים עמיתים של שכבת הערוץ להאם את פעולותיהן ואינן מכילות מידע המגיע מהשכבות הגבוהות או מיועד אליהן.

נפתח בתיאור רצף של שידורי מסגרות לפי המדיניות של עצור-והמתן שתוארה לעיל. איור 3.15 מציג רצף שידור של שלוש מסגרות שהשנייה שבהן משתבשת.

תחילה נעיר שהמדיניות של עצור-והמתן כוללת טיפול בבקרת הזרימה, שהרי היא מונעת כל אפשרות שתחנה ב תוצף: תחנה זו שולחת תגובה רק אחרי שסיימה לטפל במסגרת, ומונעת שבפק זמן זה תחנה א לא תשלח מסגרות נוספת. אולם כיצד יוכל הפרוטוקול לענות על הדרישה של דיווח על הצלחה או כישלון בהעברת מסות?

* נחיל בהצגת גרסה חלקית של הפרוטוקול ובהמשך ניג פרוטוקול עצור-והמתן מלא, המספק שירות אמין.



איור 3.15 רצף שידורי מסגרות לפי המדיניות של "עצור-והמתן"

כדי שהפרוטוקול יאפשר לשכבת הערוץ לדווח לזולח על הצלחה או כישלון בהעברת מסות, יש לתכננו כדלקמן:

1. שכבת הערוץ של תחנה ב תבדוק כל מסגרת שהיא מקבלת ותשלח שני סוגי תגובות: אם המסגרת שהתקבלה נמצאה תקינה, שכבת הערוץ בתחנה ב תעביר את המנה שבמסגרת לשכבת הרשת אצלה ותשלח לתחנה א מסגרת תגובה מסוג אישור; מסגרות אלה מסומנות ב-Ack (קיצור של Acknowledgment). אולם, אם התגלעה שגיאה במסגרת שהתקבלה, שכבת הערוץ בתחנה ב לא תעביר את המנה לשכבת הרשת ותשלח לתחנה א מסגרת הדעת שיבוש; מסגרות אלה נסמן ב-Nak.
2. כאשר תחנה א תקבל מסגרת אישור (Ack) היא תדווח לשכבת הרשת על הצלחה בשידור המנה. אם המיעד הדעת שיבוש (Nak) היא תדווח לשכבת הרשת על כישלון. אם לא הגיעה כל תגובה מתחנה ב במשך פרק זמן מוגדר, תניח תחנה א שהמסגרת ששלחה אבדה, ולכן תדווח גם במקרה זה על כישלון בהעברת המנה.

נעיר שבשירות מסוג זה שכבת הערוץ אינה מנסה לתקן שיבושים או להתגבר על אובדנים, היא מסתפקת בהעברת מסות תקינות לשכבת הרשת של תחנה ב ובדיווח "הצלחה" או "כישלון" לשכבת הרשת של תחנה א. השירות **שכבת הרשת מקבלת משכבת הערוץ** הוא **שירות לא-אמין**, אולם במקרה של כישלון, שכבת הרשת של תחנה א תדע על הכישלון ותוכל

האלגוריתם של שתי התרחנות בפרוטוקול 2 מוצג להלן. שכבת הערוץ מדווחת לשכבת הרשת, באמצעות הנהל זווית, על הצלחה או על כישלון בהעברת כל מסגרת.

פרוטוקול 2: הפרוטוקול מספק שירות עם אשורים {

תחנה שלח

בצע לולאה (אינסופית):

קבל מנה משכבת הרשת

צור מסגרת לשידור

העבר את המסגרת לשכבת הפיזית ואתחל קוצב-זמן

המתן (תגובה, פסיקת זמן) {המתן לתגובה מתחנת היעד או לתום פרק הזמן}

אם התקבל אישור תקינות דווח (הצלחה)

אחרת, דווח (כישלון)

סוף הלולאה.

תחנה מקבלת

בצע לולאה (אינסופית):

המתן (הגעת מסגרת)

קבל מסגרת מהשכבת הפיזית

אם המסגרת תקינה, בצע:

1. צור מסגרת תגובה מסוג Ack

2. בודד את שדה המידע והעבר אותו לשכבת הרשת

אחרת

1. צור מסגרת תגובה מסוג Nak

העבר לשכבת הפיזית את מסגרת התגובה

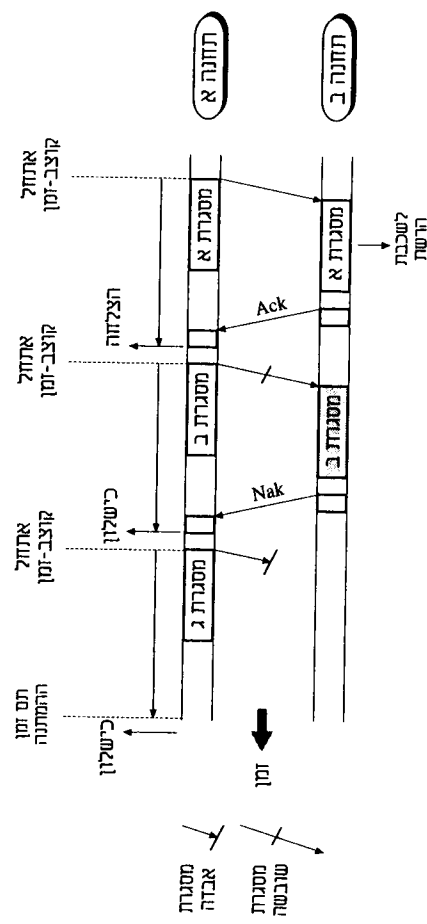
סוף הלולאה.

כמו בפרוטוקול 1, גם בפרוטוקול 2, שכבת הערוץ של תחנה א מבצעת לולאה אינסופית, שבה היא מקבלת מנה משכבת הרשת, יוצרת ממנה מסגרת ומעבירה את המסגרת לשכבת הפיזית לשידור. אולם לאחר השידור שכבת הערוץ אינה פונה מיד לקבלת מנה נוספת משכבת הרשת,

לנסות להתגבר עליו, למשל על-ידי שידור חוזר של המנה ששידורה נכשל. כלומר, הפרוטוקול מאפשר לשכבה הגבוהה יותר להתגבר על שיבושים, אך אינו עושה זאת בעצמו.

תחנה א משרדת מסגרת ונכנסת למצב המתנה; במקרה שהמסגרת אבדה ולא הוחזרה כל תגובה, ההמתנה עלולה להימשך לנצח. כדי למנוע מצב כזה, צריכה תחנה א להגביל את ההמתנה בדרך כלשהי. ואכן, תחנה א משתמשת בקוצב-זמן (timer); קוצב-זמן הוא התקן המודד פרק זמן נתון ולאחר מכן מייצר אות פסיקה, או פסק-זמן (time-out). בדרך-כלל מכונים את הקוצב לזמן הרצוי, והוא מונה את הזמן לאחור, עד הגיעו לזמן 0, ואז מייצר אות פסיקה. כאשר שכבת הערוץ שולחת מסגרת תגובה, היא מאתחלת את קוצב הזמן (על-ידי נוהל מתאים). כאשר מתקבלת מסגרת תגובה, שכבת הערוץ עוצרת את קוצב-הזמן. אם לא מגיעה מסגרת תגובה, קוצב הזמן יגיע ל-0 ויצור אירוע שייסיים את ההמתנה (אשר נוצר על-ידי הנהל המתן). למען הפשטות ניחש בשלב זה כי לא ייתכן שמסגרת תגובה תתקבל אחרי שהסתיים זמן ההמתנה שנקבע (בהמשך נסיר תהנה זו).

איור 3.16 מתאר שידור של מסגרות מידע ומסגרות תגובה ומציג גם את הטיפול בקוצב-הזמן. המסגרת הראשונה מגיעה ללא שיבוש ומתקבל עבורה אישור תקינות. שכבת הערוץ מדווחת על הצלחה בשידור. המסגרת השנייה משתבשת ומתקבלת עבורה הודעת שיבוש. שכבת הערוץ מדווחת על כישלון. המסגרת השלישית אובדת ברשת. שכבת הערוץ ממתינה עד תום הזמן שהוקצב, ולאחר-מכן מדווחת על כישלון.



איור 3.16 שידור מסגרות בפרוטוקול 2

3.2.3 שירות אמין

לציג כעת דרישה נוספת.

דרישה שלישית: המערכת תאפשר העברת נתונים מתחנה א לתחנה ב ותבטיח שכל הנתונים יתקבלו בתחנה ב כסדרם נשגם תקינים ולא כפילויות.

פרוטוקול 2 שפיתחתו לעיל דיווח על הצלחה או על כישלון בהעברת מנה, אולם לא התגבר על שיבושים ואובדנים של מסגרות. כדי לענות על הדרישה השלישית, צריך לפתח פרוטוקול אמין. פרוטוקול כזה צריך לטפל בכל מקרה של שיבוש או אובדן מסגרות. *פרוטוקול 3*, שיוצג להלן, מתגבר על שיבושים ואובדנים באמצעות שידור חוזר של כל מסגרת ששובשה או אבדה (שיטת ABR).

שידור חוזר יתבצע בפרוטוקול 3 לפי הכלל הזה:

- כאשר תחנה א תקבל מסגרת הודעת שיבוש (Nak), או כאשר החמותנה תופסק על-ידי אירוע פסק-זמן, התחנה תשדד שוב את המסגרת האחרונה ששודרה.

נתאר כעת דוגמה שתמחיש את הפרוטוקול שאנו וקוקים לו, ולאחר מכן נביא תיאור גרסה ראשונית שלו.

3.6 איור הונג פרמטות קשר צבאות

- נוהל הדיבור ברשתות קשר צבאות דומה למדי לפרוטוקול שאנו וקוקים לו. לפי נוהל זה:
- הייל משדר הודעה קצרה ומסיים אותה במילה עבורי (בדומה לסימון סוף המסגרת).
 - ההייל מחכה לתגובה מהמאזין, התגובה יכולה להיות ירתי (אם ההודעה נקלטה והובנה) או 'אמור שנית' (אם ההודעה שובשה).
 - במקרה שהתגובה היא 'אמור שנית' או כאשר לא מגיעה תגובה לאחר פרק זמן סביר, ההייל משדד שוב את ההודעה.

אלא מפעילה את קוצב-הזמן שלה ויוזמת המתנה עד שמתקבלת תגובה מתחתית היעד, או עד תום זמן המתנה שהוקצב. אם התקבל אישור תקינות (Ack), שכבת הערוץ תדווח לשכבת הרשת על הצלחה בשידור המנה. אולם אם התקבלה מסגרת הודעת שיבוש (Nak), או הסתיים הזמן שהוקצב, שכבת הערוץ תדווח על כישלון. בכל מקרה שכבת הערוץ אינה מבצעת שידור חוזר בפרוטוקול זה.

שכבת הערוץ של תחנה ב ממתנה להגעת מסגרות; כאשר מגיעה מסגרת היא בודקת אם המסגרת תקינה. אם לא התגלה שיבוש במסגרת, מעבירה שכבת הערוץ את המנה לשכבת הרשת ויוצרת מסגרת תגובה מסוג Ack. כאשר מתגלה שיבוש, המסגרת אינה מעבירה (ואינה נשמרת) ושכבת הערוץ יוצרת מסגרת Nak. לאחר-מכן מועברת מסגרת התגובה לשכבת הפיזית, כדי לשדר אותה לתחנה א.

נדון כעת בקביעת זמן החמתנה. זמן זה צריך להיות ארוך יותר מהזמן הצפוי לקבלת התגובה מתחתית היעד. הדוגמה הבאה עוסקת בקביעת זמן זה.

3.5 קביעת זמן מהמתנה לקבלת תגובה

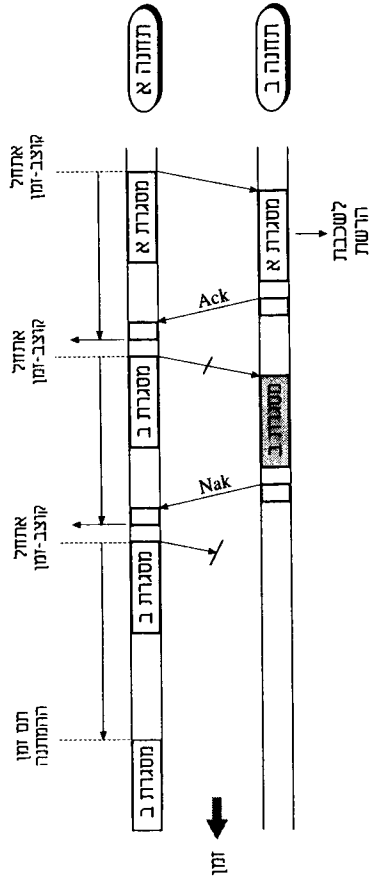
נניח שהזמן הממוצע, מתחילת שידור מסגרת על-ידי תחנה א עד גמר הגעתה לתחנה ב, הוא 10 מילי-שניות, זמן העיבוד הממוצע של מסגרת בתחנה ב הוא 2 מילי-שניות, והזמן הממוצע מתחילת שידור מסגרת האישור על-ידי תחנה ב ועד גמר הגעתה לתחנה א הוא 2 מילי-שניות. לאיזה ערך יש לאתחל את קוצב הזמן של תחנה א כדי שזמן החמתנה שלה לפני ביצוע שידור חוזר יהיה גדול מספיק?

תחנה א מצפה לקבל אישור כ-14 מילי-שניות לאחר תחילת שידור המסגרת. מאחר שמדובר בזמנים ממוצעים ורוצים לחסותן את הסיכוי שיתבצע פסק-זמן שאין בו צורך, יש לאתחל את קוצב הזמן לערך גדול יותר מאשר 14 מילי-שניות.

פרוטוקול 2 הוא פרוטוקול חד-כיווני – רק תחנה א משדרת מידע. אולם למימוש דרוש ערוץ דו-כיווני למחצה, או ערוץ דו-כיווני (asynchronous), משום שיש צורך להעביר מסגרות אישור מתחנה ב לתחנה א.

פונה לשכבת הרשת לקבל מנה נוספת, אלא משדרת שוב את אותה מסגרת במחזור הבא של הלולאה.

איור 3.17 מתאר שידור של שתי מסגרות, לפי גרסה זו של פרוטוקול 3. המסגרת הראשונה מגיעה ליעדה באופן תקין ותחנה ב מאשרת את קבלתה. המסגרת השנייה משתבשת ותחנה ב מחזירה מסגרת Nak. תחנה א משדרת אותה שוב, אך הפעם המסגרת אובדת ואינה מגיעה לתחנה ב. תחנה א מחכה לתגובה של תחנה ב עד תום פרק הזמן שהוקצב, ובסיומו שולחת את אותה מסגרת בשלישית.



איור 3.17 שידור מסגרות פרוטוקול של שידור חוזר עם תגובות Ack-1 Nak

פרוטוקול 3 (גרסה א) מאפשר להתגבר על כמות כלשהי של שיבושים ואובדנים במסגרות הנשלחות על-ידי תחנה א, שהרי כל מנה תישלח שוב ושוב, עד שיתקבל עבורה אישור תקינות. למרות זאת יש עדיין בעיות בפרוטוקול זה.

הבעיות נובעות מכך שפרוטוקול הוא אלגוריתם מבזיר, כלומר, אלגוריתם שמתבצע על-ידי כמה מבצעים (מעבדים) המרוחקים זה מזה. כדי שהמבצעים השונים יוכלו לבצע את האלגוריתם, עליהם להעביר מידע זה לזה. הבעיה במערכות תקשורת היא שהעברת המידע אורכת זמן לא ידוע, ושהמידע המועבר יכול להשתבש, או אף לא להגיע ליעדו. לפני שנתאר רצף אירועים המכשיל את הפרוטוקול שלנו, נעשה אתחולתא קלה ונתאר, באמצעות דוגמה, את המושג "אלגוריתם מבזיר" ואת הבעיה שיכולה להתעורר בהעברת מידע.

להלן תיאור של האלגוריתם :
 { פרוטוקול 3 גרסה א : הפרוטוקול מתגבר על שיבושים ואובדן מסגרות ומספק שירות אמין }

תחנה שולחת

קבל מנה (ראשונה) משכבת הרשת
 צור מסגרת לשידור
 בצע לולאה (אינסופית) :

העבר עותק של המסגרת לשכבה הפיזית ואחל קוצב-זמן

המתן (תגובה, פסיקת זמן)

אם התקבלה תגובה מסוג אישור, בצע :

1. קבל מנה משכבת הרשת { אם לא התקבל אישור, לא תועבר
2. צור מסגרת לשידור מנה נוספת, והמנה תשודר שוב

במחזור הבא של הלולאה }

תחנה מקבלת

בצע לולאה (אינסופית) :

המתן (הגעת מסגרת)

קבל מסגרת מהשכבה הפיזית

אם המסגרת שהגיעה תקינה, או

1. צור מסגרת תגובה מסוג אישור

2. בודד את שדה המידע והעבר אותו לשכבת הרשת

אחרת

צור מסגרת תגובה מסוג חודעת שיבוש

העבר את מסגרת התגובה לשכבה הפיזית

סוף הלולאה.

תחנה א מבצעת לולאה אינסופית, שבה היא מקבלת מנות משכבת הרשת, יוצרת מקו מסגרות ומשדרת אותן. עם שידור כל מסגרת מאותחל קוצב-זמן, והתחנה ממתינה לתגובה או לפסק-זמן. אם לא מגיעה תגובה, או שהתגובה היא חודעת שיבוש, שכבת הערוץ אינה

כאשר מפקד מתורדד את חייליו לפני יציאה למבצע צבאי, הוא נותן להם חוראות המחוות אלגוריתם שיביא לביצוע המשימה. זהו אלגוריתם מבוזר, משום שהוא צריך להתבצע בו-זמנית, על-ידי כל החיילים. כדי שפעולות החיילים תחיינה מתואמות, הם צריכים להעביר ביניהם מידע. למשל: חייל מסוים צופה על מוצב של האויב ומדווח על תנועות כלי רכב; כאשר הוא מבחין ברכב תנוכס למוצב הוא צריך לדווח על-כך למפקד, ולהמתין לאישור מהמפקד. אולם החודעה של החייל עלולה להשתבש ברשת הקשר, לכן חורה לו המפקד להמתין 5 דקות לאישור ואז לחזור שוב על החייון. חוראה זו עלולה ליצור שיבוש כאשר המפקד קולט דיווח של החייל, מאשר אותו, אך האישור משתבש ולא מגיע לאוזני החייל. כאשר יחלפו 5 דקות, החייל יחזור על החודעה ותמפקד עלול לחשוב שהתקבלו שני דיווחים על כניסת כלי רכב, בשעה שלאמיתו של דבר נכנס רק כלי רכב אחד.

באופן דומה, כאשר כמה מעבדים מבצעים פרוטוקול נכלומר, אלגוריתם תקשורת מבוזר, הם צריכים להעביר חודעות מאחד לשני, כדי לתאם את פעולתם ולהגיב נכון על כל מהלך אירועים. מתכנן האלגוריתם צריך להבטיח כי גם במקרה של אובדן או שיבוש המידע המועבר בין המעבדים, ידע כל מעבד כיצד לפעול. מסיפיק שיהיה צירוף מקרים אחד שבו האלגוריתם אינו פועל נכון כדי לעורר בעיה. כדי שהאלגוריתם יהיה אפיון יש לחשוב מראש על כל האפשרויות השונות.

נעבור אפוא להצגת תבנית הראשונה בפרוטוקול.

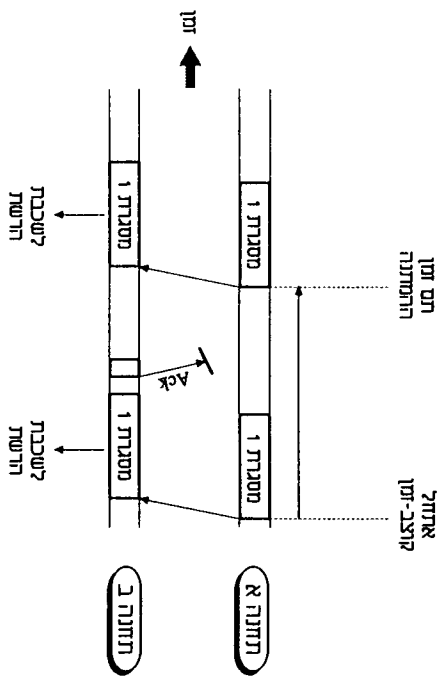
שאלה 3.2

האם לדעתך פרוטוקול 3 (גרסה א) עלול לגרום לכך שמסגרת מסוימת תתקבל יותר מפעם אחת בשכבת הרשת של תחנת היעד? נמק את תשובתך.

אם השבת נכון על שאלה 3.2, גילית מדוע פרוטוקול 3 (גרסה א), אינו מספק עדיין שירות אמין. מסגרות תגובה, ממש כמו מסגרות נתונים אינן מגיעות תמיד ליעדן, גם הן יכולות להשתבש או לאבד. במקרה של אובדן מסגרת תגובה, עלול הפרוטוקול להעביר לשכבת הרשת את אותה מנה, כמה פעמים. הבה נתבונן ברצף אירועים הכולל אובדן מסגרת תגובה, שבו מנה מעברת פעמיים לשכבת הרשת של תחנה 2:

1. תחנה א שולחת מסגרת נתונים לתחנה 2.
2. המסגרת מתקבלת בהצלחה; שכבת הערוץ של תחנה 2 מעבירה את המנה לשכבת הרשת ושולחת מסגרת אישור.
3. מסגרת האישור אובדת.
4. זמן ההמתנה של שכבת הערוץ בתחנה א פג, לכן היא מניחה כי המסגרת ששלחה שובשה או אבדה ושולחת אותה שוב.
5. גם השידור החוזר מוכתר בהצלחה ושכבת הערוץ של תחנה 2 מעבירה את המנה שהיא קבלה לשכבת הרשת.

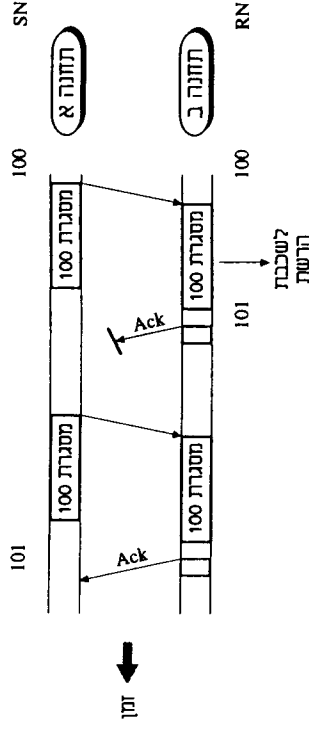
רצף אירועים זה מתואר באיור 3.18.



איור 3.18 הגברה לפולה של שניה לשכבת הרשת

המקרה המתואר באיור נוגד את ההגדרה של שירות אמין; הגדרה זו קובעת כי כל מנה צריכה להגיע ליעד פעם אחת בלבד.

איור 3.18 מתאר רצף אירועים המתחיל לאחר שתחנה א' שלחה את מסגרת 99, תחנה ב' קיבלה את המסגרת ואישרה את קבלתה, והאישור התקבל בתחנה א'. במצב זה ערכם של שני המונים הוא 100.



איור 3.19 מספור המסגרות פותר את בעיית אובדן האישורים

שאלה 3.3

הסבר מדוע ערכם של שני המונים במצב זה הוא 100.

תחנה א' שולחת את מסגרת 100 ומאתחלת את קוצב הזמן שלה. כאשר תחנה ב' מקבלת את המסגרת, היא מעבירה את המנה שבתוכה לשכבת הרשת (אנו מניחים שהמסגרת הועברה ללא שגיאות), מעדכנת את מונה הקבלה לערך 101, ושולחת מסגרת אישור. מסגרת האישור אובדת, לכן תחנה א' ממתינה עד תום הזמן שהוקצב ואז שולחת את מסגרת 100 שוב. גם השיזור החוזר מוכתר בהצלחה, אלא שעכשו ערך שדה המספור (100) קטן מערך מונה הקבלה של תחנה ב' (101), לכן היא לא מעבירה את המנה לשכבת הרשת ולא מעדכנת את ערכו של מונה הקבלה, אלא רק שולחת תגובת Ack.

שאלה 3.4

להלן נתונות הגדרות בשפת פסקל; מוגדר בהן טיפוס רשומה בשם frame, שיכול לשמש כמבנה נתונים עבור מסגרת:

הפתרון: מספור המנות הנשלחות

כדי לפתור בעיה זו, צריכה שכבת הערוץ של תחנת היעד להבחין בין מנה שמתקבלת בפעם הראשונה, לבין מנה המתקבלת שוב, לאחר שכבר התקבלה בעבר. אפשר להשיג זאת על-ידי מספור המנות הנשלחות. לצורך מספור המנות נבדקת סדר קבלתן יש לקיים תנאים אלה:

- כל תחנה צריכה להחזיק משתנה שישמש כמונה.
 - כל מסגרת צריכה לכלול שדה מספור.
- נתאר נושאים אלה להלן.

שתי התחנות צריכות להשתמש במשתנים שהם מונים, כפי שנסביר להלן:

- תחנה א' תשתמש במונה שליחה, שיאחסן את מספר המנה הבאה שצריכה להישלח; אחרי שיתקבל אישור על המנה שנשלחה, תחנה א' תוסיף 1 למונה (עדכון). מקובל לקרוא למונה זה SN (Sequence Number).
- תחנה ב' תשתמש במונה קבלה, שיאחסן את מספר המנה הבאה שצריכה להתקבל (או שתחנה ב' דורשת לקבל). מקובל לקרוא למונה זה RN (Request Number). העדכון של מונה זה יתבצע לאחר העברה של מנה תקינה לשכבת הרשת.

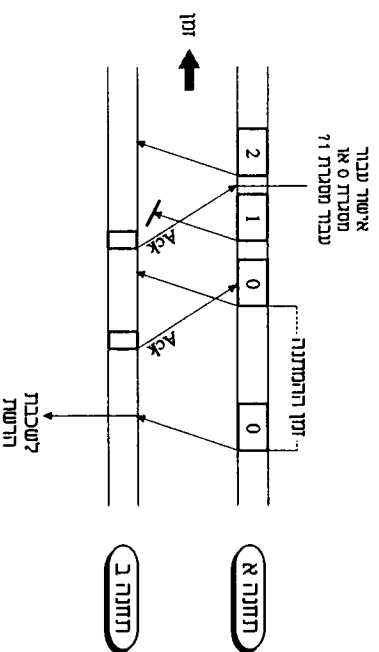
יש להוסיף שדה מספור לכוותרת של כל מסגרת מידע; תחנה א' כותבת לשדה זה את ערכו של מונה השליחה. כאשר תחנה ב' מקבלת מסגרת, היא משווה את ערכו של שדה זה עם מונה הקבלה. אם מתקיים שוויון, תחנה ב' מסיקה שהמנה שבתוך המסגרת הגיעה בפעם הראשונה, ולכן מעבירה אותה לשכבת הרשת ומעדכנת את מונה הקבלה (על-ידי הוספת 1 לערכו). אולם, אם ערך שדה המספור קטן מערכו של מונה הקבלה, התחנה מגיחה שמנה זו התקבלה כבר בעבר ונשלחה שוב, משום שהאישור עבורה אבד. לכן, במקרה זה, המנה לא תועבר לשכבת הרשת ומונה הקבלה לא יעודכן. כדי לאפשר לתחנה א' לעבור לשידור המסגרת הבאה, תשלח תחנה ב' תגובת Ack בכל מקרה.

בהמשך, התיאור הסכמטי של מסגרת יכול שדה מידע ושדה מספור. כאשר נאמר, למשל, 'מסגרת 100' נהכוון למסגרת שמספרה הסידורי הוא 100 (כלומר היא מכילה את מנה מס' 100).

הגעת האישור. במקרה כזה, תתנה א תבצע שידור חוזר, למרות שיתפכן כי המסגרת הקודמת שנישלחה, התקבלה בהצלחה. נתאר רצף אירועים שבו משתבשת פעולת הפרוטוקול, בגלל המתנה קצרה מדי לפני שידור חוזר (איור 3.20):

1. תתנה א שולחת מסגרת מספר 0. תתנה ב מקבלת אותה ומאשרת את קבלתה.
2. זמן ההמתנה של תתנה א פג לפני שהאישור הגיע, והיא משדרת שוב את מסגרת 0.
3. השידור החוזר מצליח ותתנה ב שולחת שוב מסגרת אישור (היא אמנם מגלה שמסגרת זו התקבלה כבר, ולכן לא מעבירה את המנה לשכבת הרשת).
4. תתנה א מקבלת את האישור ומשדרת את מסגרת מספר 1. מסגרת זו אובדת.
5. האישור השני עבור מסגרת 0 מגיע. תתנה א מניחה בטעות, שהאישור מתייחס למסגרת 1.

איור 3.20 מתאר רצף אירועים זה.



איור 3.20
המתנה קצרה מדי לאישור יוצרת בעיה בפרוטוקול 3

ברצף האירועים שבאיור, תתנה א מפרשת (לפי הפרוטוקול שהצגנו) את האישור השני של מסגרת 0 כאישור עבור מסגרת 1 ולכן שולחת את מסגרת 2. אבל מסגרת 1 אבדה ולכן תתנה א חיכתה צריכה לשדרה שנית. כיוון שלא עשתה זאת, המנה שנישלחה מתחנת המקור לא תועבר לשכבת הרשת. הפרוטוקול נכשל אפוא במילוי תפקידו. למעשה רצף אירועים זה עלול אף לשבש את פעולת הפרוטוקול לחלוטין.

```
const pakSize = 1000;
type
    bit = 0..1;
    posInt = 1..maxInt;
    packet = array [0..pakSize] of bit;
    frameKind = (Ack, Nak, Data);
    frame = record
        num : posInt;
        case kind : frameKind of
            Data: (info : packet);
            Ack, Nak: ();
        end;
    end;
```

- א. כתוב פונקציה בשפת פסקל שתקבל מנה ומספר סידורי, תיצור מסגרת ותחזיר אותה לתכנית הקוראת.
- ב. כתוב בשפת פסקל פרוצדורה שתיצור מסגרת הודעת שיבוש.
- ג. כתוב פונקציה בשם isFirst שתקבל כפרמטר מסגרת, ותחזיר true לגבי מסגרת שהתקבלה בתתנה ב לראשונה, ו-false לגבי מסגרת שהתקבלה כבר קודם. השתמש במשתנה הגלובלי RA.

3.2.3.1 תבניה של שידור חוזר בטרם זמן

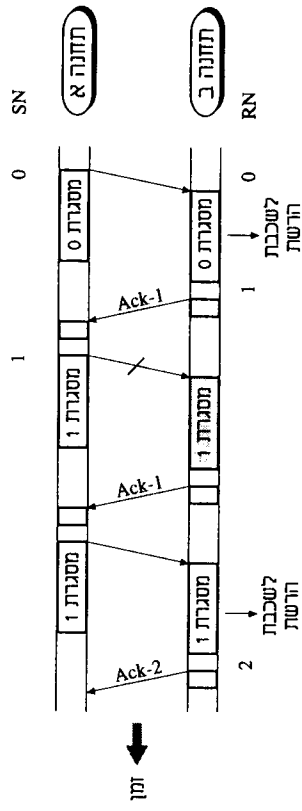
מספר המסגרות פותר את הבעיה שבה נתקלנו, לכן אפשר לחשוב כי פרוטוקול 3, כפי שהוצג, הוא תקין. אך תכנון של פרוטוקולים הוא משימה קשה, הדורשת לתת את הדעת על כל צירוף אפשרי של אירועים. נעבור אפוא לתיאור בעיה נוספת שיכולה להיווצר בפרוטוקול שהוצג לעיל.

הבעיה שנתאר נובעת משידור חוזר המבוצע לפני הגעת האישור. אין אפשרות להיות בטוחים שזמן ההמתנה, שנקבע לפני ביצוע שידור חוזר, יספיק תמיד. האישור עשוי להתמהמה בגלל עומס בתתנה ב, או בגלל סיבות אחרות, וזמן ההמתנה שהוצקב לתתנה א עלול להשתיים לפי

לאחר קבלת מסגרת 0, מעדכנת תחנה ב את RN לעד 1, ולכן האישור (Ack-1) מודיע לתחנה א לשלוח את מסגרת 1.

מסגרת 1 שובשה, לכן המנה אינה מועברת לשכבת הרשת וערכו של RN לא מעודכן.

תחנה ב שולחת שוב Ack-1 כדי לדרוש מתחנה א שידור חוזר של מסגרת 1. תחנה א מקבלת דרישה זו ושולחת שוב את מסגרת 1. הפעם השידור מוצלח והמנה מועברת לשכבת הרשת.



איור 3.21 זוגמה לספור אישורים

שאלה 3.5

סרטט איור, בדומה לאיור 3.21, אבל הנח שמסגרת 1 אובדת ולא מגיעה כלל לתחנה ב.

כעת עלינו להראות שמספר האישורים פותר את הבעיה שהגענו לעיל. איור 3.22 מראה את רצף האירועים שיוצג באיור 3.20, בתוספת מספרי האישורים. שני האישורים של מסגרת 0 הם Ack-1 והם מורים לתחנה א לשדר את מסגרת 1. לאחר קבלת האישור השני, תחנה א אינה משדרת את מסגרת 2, אלא חוזרת לשדר את מסגרת 1. כאשר השידור החוזר יגיע לתחנה ב, היא תעביר את המנה לשכבת הרשת, והפרוטוקול ימשיך לפעול כרגיל.

תכנון פרוטוקולים הוא נושא מרכזי; יש לבדוק בהירות את כל ציודי האירועים האפשריים ולוודא שהפרוטוקול מספל נכון בכל מקרה של אובדן או שיבוש של מידע.

3.2.3.2 הפתרון: מספור האישורים

כדי לתקן את הפרוטוקול, צריכה תחנה ב לכלול במסגרות התגובה שהיא שולחת, את מספרה הסידורי של המסגרת הבאה שהיא מצפה לה (ערכו של RN). תוספת המספור למסגרות התגובה מחליפה את התגובות Ack ו-Nak, ומאחדת את שני סוגי התגובות לסוג אחד בלבד. בהמשך נקרא לסוג זה של תגובות בשם הכולל **מסגרת אישור** או בקיצור **אישור**. כאשר תחנה א תקבל את האישור, היא תשדר את המסגרת הנדרשת על-ידי תחנה ב. אם אירע שיבוש, תחנה ב לא תעדכן את RN ועל-ידי כך תדרוש מתחנה א לשדר שוב את אותה המסגרת.

לפני שעבור להסביר כיצד שינוי זה פותר את הבעיה עליה הצבענו, נבדוק בדוגמה הבאה, תאם מיספור האישורים אמנם מעביר לתחנה א מידע על שיבושים במסגרות שקלטות בתחנה ב. (ולכן יכול להחליף את התגובות Ack או Nak).

דוגמה 3.7 מספור האישורים מספיק אפקטיבי

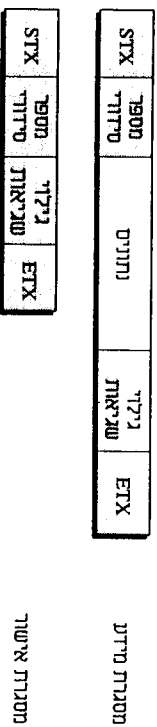
בדוגמה זו נראה כיצד מספור מסגרות האישור מעביר לתחנה א מידע על שיבושים. נעשה זאת באמצעות רצף האירועים הזה:

1. תחנה א שולחת מסגרת 0;
2. תחנה ב מקבלת אותה ומאשרת את הקבלה;
3. תחנה א מקבלת את האישור ושולחת מסגרת מס' 1;
4. המסגרת מגיעה ותחנה ב מגלה בה שיבוש;
5. תחנה א שולחת את מסגרת 1 שוב.

איור 3.21 מתאר רצף זה. נסמן את האישורים באיור באמצעות הצירוף n -Ack, שפירושו שתחנה ב קיבלה בהצלחה את המסגרת n -1, ומבקשת לקבל את המסגרת n .

השדה לגילויי שגיאות במסגרת האיזור דרוש כדי להבטיח שהמידע שבשדה המספור לא שובש. אם מתגלה שגיאה במסגרת האיזור, תחנה א' מתעלם ממנה ותתנתג כאילו האיזור לא הגיע.

איור 3.23 מתאר את מבנה המסגרות (בתחנה שתווים מיוחדים משמשים לציון גבולות המסגרת).



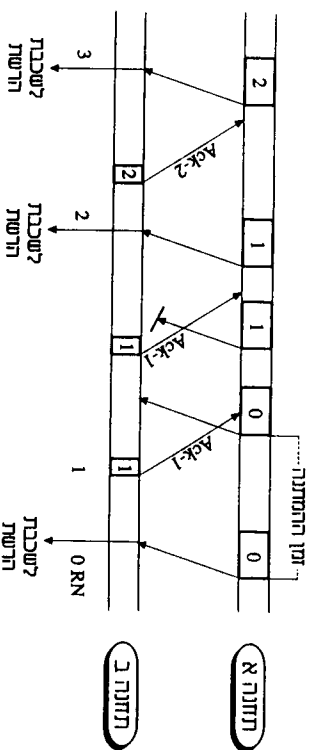
איור 3.23
 מבנה מסגרת פידע ומסגרת אישור פרוטוקול 3.

נתאר עתה את האלגוריתם של פרוטוקול 3.

כאמור, כל אחת מן התחנות משתמשת במונה כדי לזכור את מספר המסגרת שהיא צריכה לשלוח או לקבל. תחנה א' משתמשת במונה SN שמאחסן את המספר הסידורי של המסגרת הבאה שתישלח. תחנה ב' משתמשת במונה RA שמאחסן את המספר הסידורי של המסגרת הבאה שהיא מצפה לקבל.

נמשיך להניח כי לשכבת הרשת של תחנה א' יש תמיד מנות מוכנות לשידור. בעמוד הבא מתואר האלגוריתם של תחנה א'. קרא אותו לפני קריאת הקטע להלן.

תחנה א' מבצעת לולאה אינסופית; בכל מחזור של הלולאה היא משדרת מסגרת ומאתחלת את קוצב הזמן לעוד שנקבע. לאחר מכן היא ממתינה לקבלת אישור או לפסקת זמן המתרחשת כאשר תם זמן ההמתנה. אם התקבל אישור, בודקת התחנה את מספרן. אישור תקינות צריך להכיל מספר הגודל ב-1 מ-SN, משום שהתחנה ב' מצפה לקבל את המסגרת הבאה. אם התקבל אישור כזה, שכבת הערוץ מעודכנת את ערכו של SN, פונה לשכבת הרשת, מקבלת את המנה הבאה לשידור, ויוצרת ממנה מסגרת חדשה.



איור 3.22
 מספור האישוורים פותר את הבעיה של איור 3.20.

תיקון זה פותר את הבעיות של הפרוטוקול שהצגנו, ואפשר להראות (אם כי לא נביא כאן הוכחה פורמלית) שהמעם מתקבל פרוטוקול אמין, שיכול להתגבר על כל צירוף של שיבושים ואובדנים וביניהם גם זמן המתנה קצר מדי. מובן שאמינות הפרוטוקול תלויה ביכולת גילוי השגיאות; אם שגיאה במסגרת לא תתגלה, עלולה להגיע לשכבת הרשת מנה משובשת. אולם כפי שראינו בפרק 1, בשיטת CRC (השיטה הנפוצה לגילוי שגיאות) החסתברות לכך נמוכה מאוד.

3.2.4 מימוש פרוטוקול 3

כדי לממש את פרוטוקול 3 (בגרסתו הסופית) מסגרת המידע צריכה להיות מורכבת משדות אלה:

1. שדה מידע
2. שדה מספור
3. שדה לגילוי שגיאות

מסגרת אישור צריכה להכיל שני שדות:

1. שדה מספור
2. שדה לגילוי שגיאות

{פרוטוקול 3 (גרסה סופית): הפרוטוקול מתגבר על שינושים ואובדן מסגרות ומספק שירות אמין}

תחנה שולחת

$$SN \leftarrow 0$$

קבל מנה (ראשונה) משכבת הרשת

צור מסגרת לשידור

בעז לולאה (אינסופית):

העבר העתק של המסגרת לשכבה הפיזית ואתחל קוצב-זמן

המתן (תגובה, פסיקת זמן)

אם התקבל אישור תקינות בעז:

$$SN \leftarrow SN + 1.1$$

2. קבל מנה משכבת הרשת {אם לא התקבל אישור תקינות, לא תועבר

3. צור מסגרת לשידור מנה נוספת, והמנה תשודר שוב במחזור הבא

של הלולאה}

סוף הלולאה.

אם המסגרת שנשלחה יושבה, יתקבל אישור שמספרי שווה ל-SN. אישור כזה משמע שתחנה ב דורשת לשלוח את המסגרת שוב. גם אם לא התקבל אישור (כלומר זמן ההמתנה נגמ), או שהתקבל אישור משובש, יש לשלוח את המסגרת שוב. בכל המקרים הללו, שכבת הערוץ אינה מקבלת מנה חדשה ואינה מעדכנת את SN. משמעות הדבר היא שבמחזור הבא של הלולאה תשודר שוב אותה מסגרת.

קרא כעת את האלגוריתם של תחנה B, בעמוד הבא.

שכבת הערוץ של תחנה B מבצעת לולאה שבה היא ממתינה להגעת מסגרת. כאשר מתקבלת מסגרת מהשכבה הפיזית, בודקת תחנה B שני תנאים: האם המספר הסידורי של המסגרת שהתקבלה שווה למספר הסידורי שהיא מצפה שיהיה למסגרת זו (RN) והאם המסגרת תקינה. אם המספרים שווים ולא התגלו שגיאות במסגרת, מתחלת שכבת הערוץ את המנה ומעבירה אותה לשכבת הרשת; אחר-כך היא מעדכנת את RN (כדי שתצפה למסגרת הבאה).

תחנה מקבלת

$$RN \leftarrow 0$$

בעז לולאה (אינסופית):

המתן (הגעת מסגרת)

קבל מסגרת מהשכבה הפיזית

אם המספר הסידורי של המסגרת שהתקבלה שווה לערכו של RN

וגם המסגרת שהגיעה תקינה או

בעז:

בודד את שדה המידע והעבר אותו לשכבת הרשת

$$RN \leftarrow RN + 1$$

צור מסגרת אישור (והשם את ערכו של RN לשדה המספר שלה)

העבר לשכבה הפיזית את מסגרת האישור

סוף הלולאה.

אולם, אם מספר המסגרת אינו שווה לערכו של RN, מסיקה שכבת הערוץ כי המנה שבמסגרת התקבלה כבר בעבר (וכנראה שודרה שוב, מפני שהאישור עבורה אבד) ולכן היא אינה מעבירה אותה לשכבת הרשת. מובן שגם מנה הרשומה בתוך מסגרת משובשת לא מועברת לשכבת הרשת. אישור עבור הגעת המסגרת נשלח בכל מקרה, אבל במקרה זה ערכו של RN לא עודכן, כך שהאישור יודיע לתחנה א שעליה לשדר שוב את אותה מסגרת.

3.2.4.1 קביעת גודלו של שדה המספור

פרוטוקול 3 יש בכל זאת בעיה: עם התמשכות השידור גדלים המספרים הסידוריים של המסגרות, ולבסוף לא יהיה בשדה המספור מקום לאחסנם. נשאלת השאלה: כמה סיביות יש לחקות עבור שדות המספור? מובן שגודל השדה (בסיביות) יקבע את המספר הסידורי המרבי האפשרי. כדי לחקטין את התקורה, רצוי להגביל את תחום המספרים הסידוריים של המסגרות.

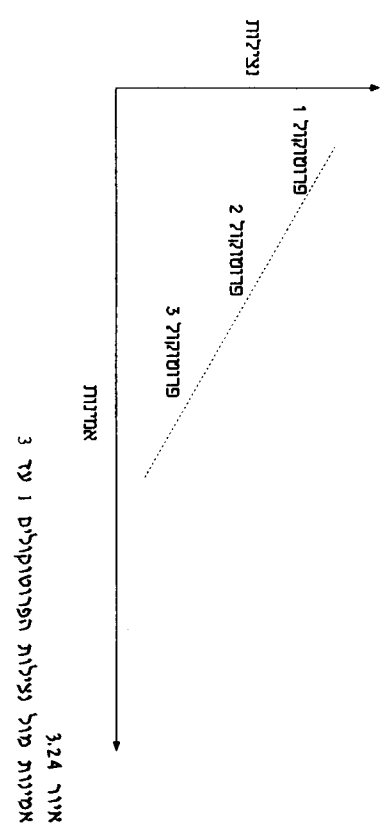
מבחינת רמת האמינות. פרוטוקול 1 מספק שירות לא-אמין, ללא אישורים – זהו שירות ברמת אמינות נמוכה; מחצית השני, פרוטוקול 3 מספק שירות אמין. שיפור האמינות מושג במחיר של הפחתת היעילות.

כיצד אפוא מודדים יעילות? קיימים מודדים שונים; נגזיר להלן מדד חשוב אחד, ובהמשך נגזיר מודדים נוספים.

נצילות הערוץ (channel utilization) – אחוז הזמן שבו הערוץ מנוצל לשידור מוצלח של נתונים.

הדגש הוא על המילים 'מוצלח' ו'נתונים'; כלומר מסגרת שהשתבשה אינה נחשבת כשידור מוצלח, ולכן היא מפחיתה את נצילות הערוץ; גם שידור של שדות נוספים, פרט לשדה המידע, מפחית את נצילות הערוץ, כי אינו נחשב לשידור נתונים. באופן דומה, שידור מסגרת אישור מפחית את נצילות הערוץ.

אם מתעלמים משיבושים, הנצילות המרבית של פרוטוקול 1 יכולה להגיע ל-100%. בפרוטוקול זה הערוץ יכול להיות מנוצל כל הזמן, כי לא ממתניינים לתגובה ולא מבצעים שידור חוזר. נצילות הערוץ של פרוטוקולים 2 ו-3 נמוכה יותר. האיור הבא מציג את הפרוטוקולים שהצגנו עד כה בגרף המתאר אמינות מול יעילות. הגרף מציג תמונה איכותית, אנו לא מתחייבים ליחידות של אמינות או נצילות, אלא ליחס בין הפרוטוקולים.



בפרוטוקול עצור-המתין יש להבחין בכל שלב רק בין המסגרת שמצפים לה, לבין המסגרת הקודמת, שבב המקבלה. שהרי לא ייתכן שתגיענה לנתנת היעד מסגרות קודמות יותר*.

שאלה 3.6

הנתונה א-ב מתקשרות בפרוטוקול עצור-המתין חד-כיווני, עם שידורים חוזרים. תחנה ב אישרה את קבלתה של מסגרת מספר 12. הסבר מדוע תחנה ב יכולה לצפות בשלב זה אך ורק לקבלת מסגרת 12 או מסגרת 13.

כדי להבחין בין המסגרת הבאה לה מצפים, לבין המסגרת הקודמת, מספיק למספר את המסגרות במספרים 0 ו-1 לסירוגין. אפשר להסתפק אם-כן בשדה מספר בודל של סיבית אחת.

אפשר להכניס שינוי זה לאלגוריתם של פרוטוקול 3, אם עדכון המשתנים יעשה על-ידי הוספת 1, מודולו 2. למשל, בשפת פסקל יעדכן מונה השליחה באמצעות הממשט:

$$SN := (SN + 1) \bmod 2;$$

שאלה 3.7

תאר מה קורה בפרוטוקול 3 (מבחינת כמות השידורים החוזרים), כאשר קובעים זמן המתנה כך שבחלק גדול מן המקרים זמן זה תם לפני הגעת האישור.

3.2.5 היעילות של פרוטוקול "עצור והמתן"

בסעיף זה נזון ביעילותו של פרוטוקול עצור-המתן. לכל אחד מאתנו יש תחושה אינטואיטיבית של המושג 'יעילות', אך לפני שנוכל לדון ביעילות באופן מדויק יותר, עלינו להגדיר כיצד מודדים יעילות של פרוטוקול. במהלך הסעיף הצגנו שלושה פרוטוקולים למילוי דרישות שונות ממערכת תקשורת. כל אחד מפרוטוקולים אלה הוא שיפור ביחס לקודמו,

* דבר זה נובע מן התנוחה שהערוץ הפיזי עובד לפי מודל FIFO, כלומר מסגרות מתקבלות לפי סדר שידורן ולא ייתכן שמסגרת יחשיגי מסגרת ששודרה לפניה.

הצילות היא הזמן שהערוץ עסוק בשידור נתונים, כלומר, זמן שידור מסגרת, מחולק בזמן הכולל להשלמת הטיפול של הפרוטוקול במסגרת. הזמן הכולל (לפי ההנחות שלנו) הוא זמן השידור ועוד פעמיים זמן ההתפשטות, כלומר:

$$U = \frac{T_r}{T_r + 2 \times T_p} = \frac{1}{1 + \frac{2T_p}{T_r}}$$

מנוסחה זו נובע שכאשר השבר $\frac{T_p}{T_r}$ הוא קטן מאוד, הצילות תהיה קרובה ל-1 (כי נקבל שהצילות היא 1 חלקי 1 ועוד שבר קטן); אולם, כאשר זמן ההתפשטות גדול ביחס לזמן שידור מסגרת, נקבל צילות נמוכה יותר. למשל, כאשר זמן ההתפשטות שווה לזמן השידור נקבל ש- $1 = \frac{T_p}{T_r}$ ולכן צילות הערוץ תהיה שליש. כאשר T_p גדול יותר מ- T_r , נקבל צילות נמוכה מאוד.

מסקנה זו אינה מתיישבת בקלות עם הניסיון היומיומי שלנו. אנו רגילים למצבים בהם זמן ההתפשטות הוא קטן מאוד, כמעט זניח. יש לזכור כי מהירות התפשטות של אותות אלקטרומגנטיים היא גבוהה ביותר (כ-300,000 קמ"ש), ולכן כאשר המקור והיעד אינם מרוחקים זה מזה, זמן ההתפשטות אכן קטן ביותר. דוגמה 3.6 הציגה שידור מידע ברשתות קשר צבאיות, ברשתות כאלה זמן השידור הוא ארוך יחסית (שניות אחדות) וזמן ההתפשטות הוא קצר – פחות ממילי-שנייה (או, אלפית שנייה). לכן, חיל המשדר הודעה יכול להמתין עד שהוא שומע 'רות' או 'אמור שנית' ללא בזבז זמן רב. אולם לגבי תקשורת למרחקים ארוכים, נוהל דומה עלול להיות בעייתי:

השהיית התפשטות טיפוסית של שידור ללוויין היא כרבע שנייה (כפי שחישבת בפרק 2, בשאלה 2.7). ניח שאורך מסגרת הוא 2000 סיביות וקצב השידור הוא 100K סל"ש. נחשב את הזמן הדרוש לשידור מסגרת על-ידי חלוקת אורך המסגרת (בסיביות) בקצב השידור (בסיביות לשנייה):

$$\frac{2,000 \text{ bits}}{100,000 \text{ bps}} = \frac{2}{100} = 20 \times 10^{-3} \text{ s}$$

אנו מעוניינים לפשט את הדיון בהמשך, לכן נתעלם מגורמים המשפיעים על צילות הערוץ. נמנה אותם להלן:

1. שגיאות ושידורים חוזרים (ניח שההסתברות לשגיאה היא זניחה);
2. שדות תקורה במסגרת (ניח כי אורך הכותרת ואורך הסיומת זניחים ביחס לאורך שדה המידע);
3. זמן הטיפול במסגרת בתחתית היעד וזמן שידור מסגרת האישור (ניח שזמן זה זניח ביחס לזמנים האחרים).

כאשר מתעלמים מגורמים אלה, נשארים שני גורמים הקובעים את צילות הערוץ של פרוטוקול עצור-והמתן:

1. **השהיית ההתפשטות (propagation delay)** – הזמן הדרוש לאות כדי לעבור מקצה אחד של הערוץ לקצה שני.
2. **זמן שידור מסגרת (transmission time)** – הזמן מתחילת שידור מסגרת עד סיום השידור.

נסביר זאת: במשך זמן שידור המסגרת, הערוץ מנוצל באופן מלא (אנו מניחים שהמסגרת כוללת נתונים בלבד, והערוץ עסוק בהעברת סיביות הנתונים של המסגרת). אולם מרגע ששדרה הסיבית האחרונה במסגרת ועד שסיבית זו מגיעה לתחתית היעד, הערוץ בטל ואינו מנוצל, משום שתחתית המקור עצרת וממתינה לתגובה. כאשר הסיבית האחרונה מגיעה לתחתית היעד, התחנה מטפלת במסגרת (בודקת אם שובשה וכיו') ומשדרת מסגרת תגובה. כאמור, אנו מתעלמים מזמן הטיפול ומזמן שידור מסגרת התגובה, אך תחתית המקור צריכה להמתין עד **קבלת** מסגרת התגובה. מסגרת התגובה תתקבל רק אחרי שהאות התפשט מתחתית היעד לתחתית המקור. כלומר, בסך הכל הערוץ לא יהיה מנוצל במשך זמן ששווה לכפליים השהיית ההתפשטות.

- נפתח נוסחה מתמטית על-פי הסבר זה.
- סמן: T_r – זמן שידור מסגרת (בשניות)
 T_p – זמן (השהיית) ההתפשטות (בשניות)
 U – צילות הערוץ

$$T_p = \frac{10^3 \text{ m}}{2 \times 10^8 \frac{\text{m}}{\text{s}}} = \frac{10 \times 10^3}{2 \times 10^9} = \frac{5 \times 10^3}{10^9} = 5 \times 10^{-6} \text{ s}$$

נחלק את זמן ההתפשטות בזמן השידור, נקבל:

$$\frac{T_p}{T_f} = \frac{5 \times 10^{-6} \text{ s}}{10^{-3} \text{ s}} = 5 \times 10^{-3}$$

קיבלנו שבר קטן, ולכן נצילות הערוץ קרובה ל-1 (או ל-100%).

ב. הפעם אורך הערוץ הוא 200 ק"מ, לכן זמן ההתפשטות הוא:

$$T_p = \frac{200 \times 10^3 \text{ m}}{2 \times 10^8 \frac{\text{m}}{\text{s}}} = 10^{-3} \text{ s}$$

זמן זה שווה לזמן השידור, לכן נקבל:

$$\frac{T_p}{T_f} = \frac{10^{-3} \text{ s}}{10^{-3} \text{ s}} = 1$$

ולכן נצילות הערוץ היא שליש (כ-33%).

ג. במקרה זה הערוץ ארוך במיוחד, לכן נקבל זמן ההתפשטות ארוך במיוחד:

$$T_p = \frac{50 \times 10^6 \text{ m}}{2 \times 10^8 \frac{\text{m}}{\text{s}}} = \frac{5 \times 10^7}{2 \times 10^8} = \frac{5}{20} = 0.25 \text{ s}$$

נחלק זמן זה בזמן השידור ונקבל:

$$\frac{T_p}{T_f} = \frac{0.25 \text{ s}}{10^{-3} \text{ s}} = 250$$

נצילות הערוץ תהיה:

$$U = \frac{1}{1 + 2 \times 250} \approx \frac{1}{500} = 0.002$$

נצילות של 0.2% בלבד.

כלומר 20 מילי-שנייה. מאחר שהזמן עד הגעת האישור הוא חצי שנייה נזמן ההתפשטות הליך

והזמן או 500 מילי-שניות, נקבל שנצילות הערוץ המרבית היא כ-4 אחוזים בלבד:

$$\frac{20}{520} = \frac{1}{26} \approx 4\%$$

כלומר יהיה מעט זמן שידור והרבה זמן המתנה. מצב זה אף יחריף ככל שקצב השידור יהיה גבוה יותר. נביא להלן דוגמה שבה ננתח את נצילות הערוץ ונשווה ערוצים בעלי אורך שונה.

דוגמה 3. חישוב נצילות של ערוץ

נחשב את נצילות הערוץ בפרוטוקול עצור-והמתן שבו גודל המסגרות הוא 1000 סיביות, קצב השידור הוא 1M סי"ש, ומהירות ההתפשטות היא 2×10^8 מטרים לשנייה (זו מהירות אופטימית של התפשטות אות חשמלי בכבל), עבור ערוץ באורכים האלה:

- א. 1 ק"מ
- ב. 200 ק"מ
- ג. 50,000 ק"מ.

נחשב את זמן השידור על-ידי חלוקת אורך המסגרת (בסיביות) בקצב השידור (בסיביות לשנייה).

בקצב של 1M סי"ש נקבל:

$$\frac{1,000 \text{ bits}}{1,000,000 \text{ bps}} = 10^{-3} \text{ s}$$

כלומר זמן שידור מסגרת בתכונים אלה הוא 1 מילי-שנייה.

א. זמן ההתפשטות מחושב על-ידי חלוקת אורך הערוץ במהירות ההתפשטות של האות בערוץ (נוכח כי זמן = מרחק/מהירות):

לסיכום דוגמה זו אפשר לומר כי בקצב שידור מהיר יחסית (IM) וכאשר אין אפשרות להגדיל ללא סוף את אורך המסגרת (משיקולים שתיארנו בסעיף 3.1.3.7), פרוטוקול עצור-המתן הוא סביר רק כאשר אורך הערוץ אינו גדול מדי (ולכן השהיית ההתפשטות קטנה יותר מאשר זמן השידור). כאשר הערוץ ארוך, יש צורך בפרוטוקול מתוחכם יותר כדי לקבל נצילות סבירה של הערוץ.

שאלה 3.8

חזור על הדוגמה לעיל, אך הפעם תח שקצב השידור הוא 1K סלי"ש.

בהמשך ניצג פרוטוקולים של שידור ברצף, שמטרתם לשפר את נצילות הערוץ של פרוטוקול עצור-המתן, בלי להוריד את רמת האמינות.

3.2.6 פרוטוקול דו-כיווני

נוסף כעת עוד דרישה.

דרישה רביעית: שתי התחנות יוכלו לשדר נתונים. הערוץ יספק שירות אמין דו-כיווני.

כדי למלא דרישה זו, צריכה כל תחנה להכיל את התוכנה של התחנות א ו-ב גם יחד. כלומר, כל תחנה תחזיק מונה שליחה, קוצב זמן, מונה קבלה וכדומה, ותבצע הן את הפעולות של השולח והן את הפעולות של המקבל. גישה זו אפשרית, אך במקרים רבים אפשר להימנע ממשלוח מסגרות אישור וכך לשפר את נצילות הערוץ.

לעמים קרובות אפשר 'להעמיס' את מסגרת האישור על גבי מסגרת נתונים שנשלחת ממלא. שיטה זו נקראת **שיטת שק הקמה לאישורים (piggybacking)**. היתרון של שיטה זו הוא האפשרות להימנע ממשלוח מסגרות אישור מיוחדות. יש בכך חיסכון משום שמסגרת האישור, כמו כל מסגרת, מכילה: כותרת, שדה לגילוי שגיאות וסימון של סוף המסגרת ותחילתה. שידור סיביות אלה במיוחד עבור האישור הוא בזבזני, אם אפשר לצרף את האישור למסגרת נתונים הנשלחת ממלא.

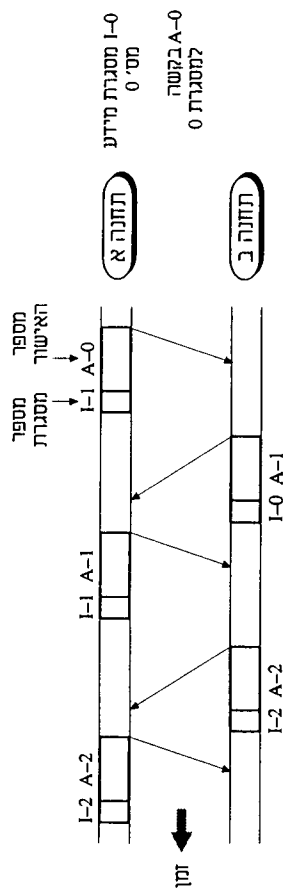
שיטת שק-הקמה מסבכת את הפרוטוקול, כי צריך לטפל באופן מיוחד במצב שבו לתחנה אין נתונים לשידור. במקרה כזה, שכתב הערוץ צריכה להיות מסוגלת לשדר, לאחר פרק זמן מסוים, מסגרת אישור עצמאית (לא בתוך מסגרת מידע), כדי לאפשר לתחנה שמנגד להמשיך בשידורים.

כדי לממש שיטה זו, יש לחוסף שדה לכל מסגרת מידע, עבור מספר האישור. כמו-כן יש לחוסף שדה "סוג", שיציין אם זו מסגרת מידע (שיכולה להכיל גם אישור) או מסגרת אישור עצמאית. שדה זה יכול להיות באורך סיביות אחת. באיור 3.25 מתואר מבנה של מסגרת מידע בפרוטוקול דו-כיווני, עם שיטת שק-הקמה לאישורים (לא הצגנו את סימון גבולות המסגרת):

סוג	מספר המסגרת	מספר האישור	נתונים	שגיאות גילוי

איור 3.25 מבנה מסגרת בפרוטוקול דו-כיווני

איור 3.26 מתאר סדרת שידורים בפרוטוקול דו-כיווני.



איור 3.26 שידור מסגרות בפרוטוקול דו-כיווני עם אישורים בשיטת שק-הקמה

תחנה א משדרת את מסגרת 0; מסגרת זו מתקבלת בתחנה ב; תחנה ב מצרפת את האישור לקבלתה (שמסומן A-1) למסגרת 0 שהיא מעבירה לתחנה א. תחנה א מאשרת את קבלת מסגרת 0 יחד עם מסגרת 1 שהיא שולחת. באופן זה ממשיכה ה"שיחה" בין התחנות.

במצב המתואר באיור 3.26 אין צורך במסגרות אישור מיוחדות, כי לשתי התחנות יש כל הזמן מסגרות לשידור. אם תחנה כלשהי צריכה לשלוח אישור ואין לה מסגרת מוכנה לשידור, היא

3.2.7 סיכום

בסעיף זה הצגנו פרטוקול ערוץ בסיסי, המכונה פרטוקול עצור-המתן; פרטוקול זה מתייב את התנהגות המשדרת מסגרת לעצור ולהמתין לתגובה מתחתנת היעד, לפני שהיא משדרת מסגרת נוספת. פיתחנו את הפרטוקול בהדרגה, החל מפרטוקול המספק שירות לא-אמין (לא יעיל), דרך פרטוקול שמספק שירות לא-אמין, עם אישורים (והוא פחות יעיל) וכלה בפרטוקול שמספק שירות שהוא אמין (לאד הכי פחות יעיל מבין השלישה).

המדרגו את נעילות הערוץ כמדד יעילות והישבנו את הנעילות בפרטוקול עצור-המתן. ראינו שגם כאשר מניחים שאין שגיאות שידור, נעילות הפרטוקול היא נמוכה כאשר השתיית ההתפשטות גדולה ביחס לזמן הדרוש לשידור מסוגר.

להלן נסכם בטבלה את שלושת הפרטוקולים שהצגנו בסעיף זה.

פרטוקול	שירות	בקרת שגיאות	בקרת זרימה	נעילות
1	לא-אמין	אין	אין	גבוהה
2	לא-אמין, עם אישורים	יש, אך הפרטוקול לא נותנבר על שגיאות	יש	תלויה באורך הערוץ
3	אמין	יש	יש	תלויה באורך הערוץ

תמתין זמן קצוב להגעת מנה משכבת הרשת; (כדי למדוד את זמן ההמתנה יש צורך בקובץ-זמן נוסף בכל תחנה). אם בתום זמן זה לא תתקבל מנה לשידור, התנהגות תשדר מסגרת אישור עצמאית.

נביא להלן את האלגוריתם המילולי של פרטוקול 4, פרטוקול עצור-המתן די-כיווני, עם אישורים, בשיטת שיק הקמור. מאחר שהפרטוקול הוא די-כיווני, דרוש רק אלגוריתם אחד, שמשדרת את שתי התחנות. הפרטוקול מתבצע בו-זמנית בשתי התחנות המתקשרות ביניהן. אנו מניחים שלשכבת הרשת יש תמיד מנות מוכנות לשידור, לכן אין צורך לטפל באופן מיוחד במשלוח אישור במקרה שאין מנות לשידור.

{פרטוקול 4: פרטוקול עצור-המתן די-כיווני, מתגבר על כל צירוף של שיבושים ואובדנים }

SN ← 0
RN ← 0

קבל מנה (ראשונה) משכבת הרשת

בצע לולאה (אינסופית):

צור מסגרת לשידור {הנס למסגרת מספר סידורי SN ומספר אישור RN}

תעבר עותק של המסגרת לשכבת הפיזית

המתן (המתת מסוגרת)

קבל מסוגרת מהשכבה הפיזית

אם המסגרת תקינה אז:

אם המספר הסידורי שלה שווה ל-RN

בצע:

1. בודד את שדה המידע ותעבר אותו לשכבת הרשת

2. עדכן את RN

אם שדה האיזור של המסגרת שווה ל-SN {התקבל אישור עבור המסגרת שנישלחה}

בצע:

1. קבל מנה חדשה משכבת הרשת

2. עדכן את SN

סוף הלולאה.

