

# FLL Programming 101 NXT-G



Chapter #1  
September 2007  
Version 1.1b

# Legal Stuff

---

© 2006 INSciTE in agreement with, and permission from FIRST and the LEGO Group. This document is developed by INSciTE and is not an official FLL document from FIRST and the LEGO Group. This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc-sa/2.0/>

or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

LEGO®, ROBO LAB, and MINDSTORMS™ are trademarks of the LEGO Group used here with special permission. FIRST™ LEGO® League is a trademark owned by FIRST (For Inspiration and Recognition of Science and Technology) and the LEGO Group used here with special permission. INSciTE™ is a trademark of Innovations in Science and Technology Education.

INSciTE

PO Box 41221

Plymouth, MN 55441

[www.hightechkids.org](http://www.hightechkids.org)

# Creative Commons License

---

High Tech Kids is committed to making the best possible training material. Since HTK has such a dynamic and talented global community, the best training material and processes, will naturally come from a team effort.

Professionally, the open source software movement has shown that far flung software developers can cooperate to create robust and widely used software. The open source process is a model High Tech Kids wants to emulate for much of the material we develop. The open source software license is a key enabler in this process. That is why we have chosen to make this work available via a Creative Commons license. Your usage rights are summarized below, but please check the complete license at: <http://creativecommons.org/licenses/by-nc-sa/2.0/>.

# Credits

---

This presentation was developed by Doug Frevert. It is based on the work of Fred Rose. The accompanying labs were originally done in RCX Code by Joel Stone and converted to ROBOLAB by Doug Frevert. A portion of the material is taken from *“Building LEGO Robots for FIRST LEGO League”* by Dean Hystad. Amy Harris defined the 10 programming steps. Eric Engstrom, Jen Reichow, and Ted Cochran reviewed ongoing drafts. Eric taught the first class and helped modify the content accordingly.

# תכנות מחשב הרובוט

---

- מטרות
  - לימוד תוכנת הרובוט
    - FLL, NXT, NXT-G
  - מבנה התוכנה
    - תיאוריה
    - דוגמאות בשפת NXT
    - ניסיון מעשי
- שיעור זה לא מהווה REF לכל פקודת NXT

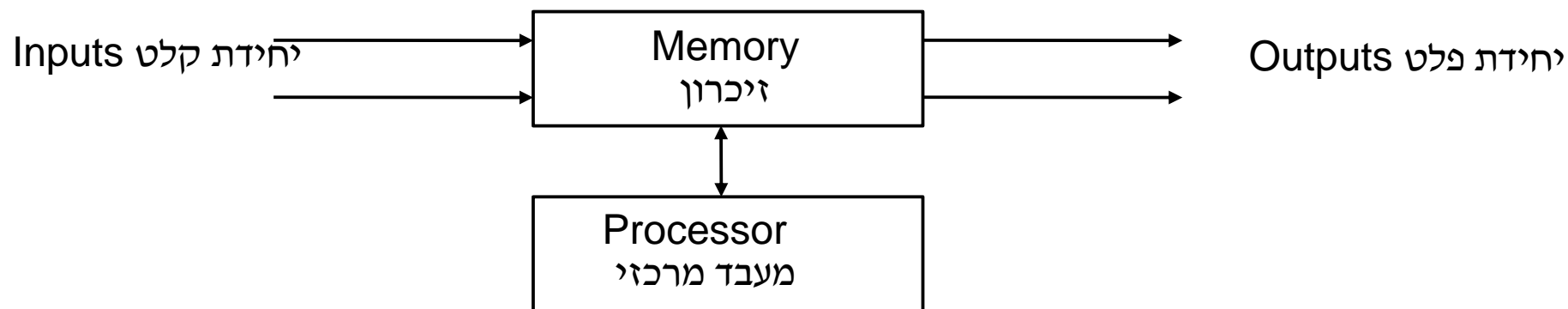
# נושאי ההרצאה

---

- Sensors
- Lab #3
- נושאים מתקדמים
- פתרון בעיות
- מבנה מחשב
- סביבת התכנות
- Moving and Turning
- Lab #1
- MyBlocks and Loops
- Lab #2

# מבנה מחשב

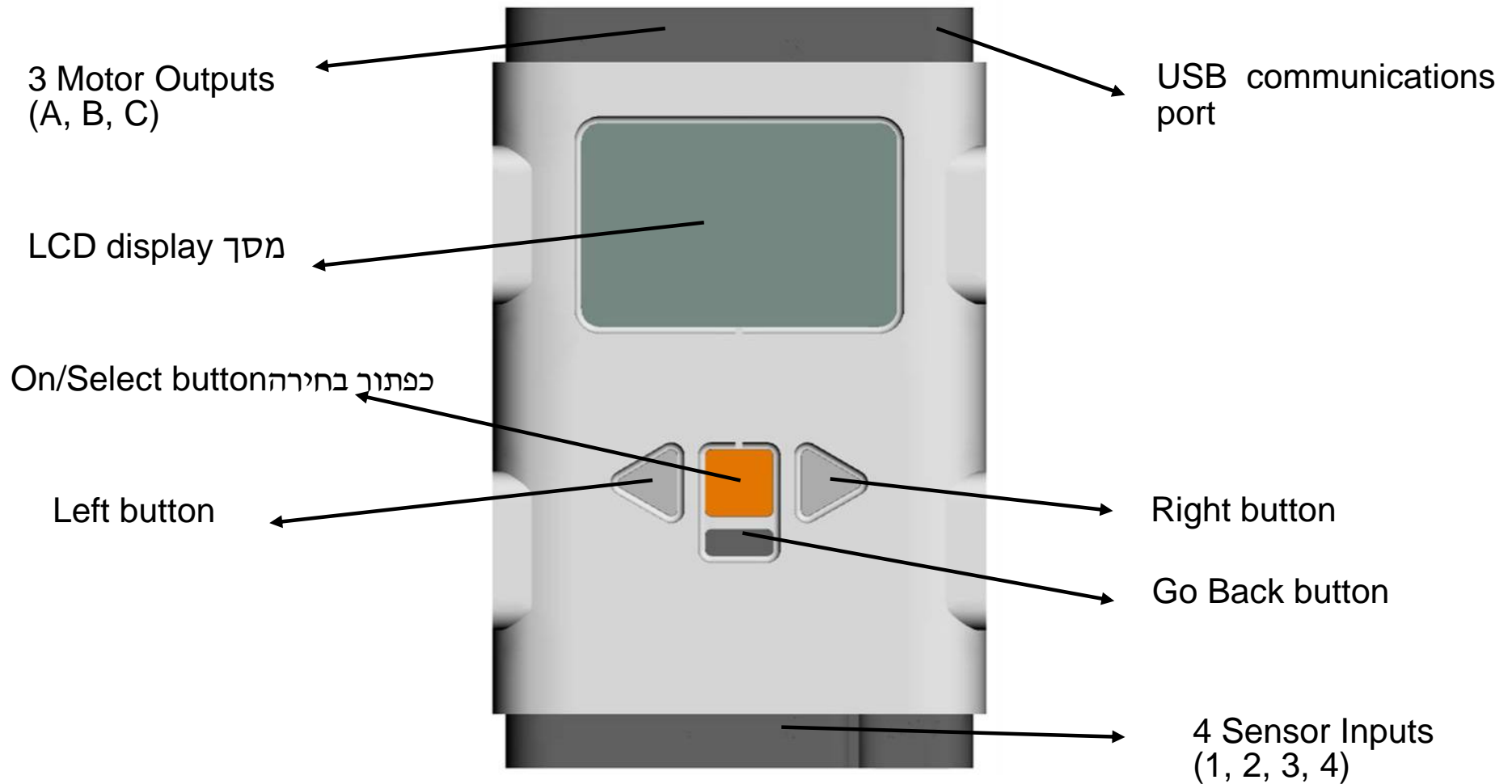
# The Computer (Generic)



- המעבד המרכזי מבצע את הפקודות
- הזיכרון מאחסן את התוכנית וכן נתונים (משתנים)
- יחידת הקלט מספרת למחשב על העולם החיצון
- יחידת הפלט מספרת לעולם החיצון על המחשב



# NXT

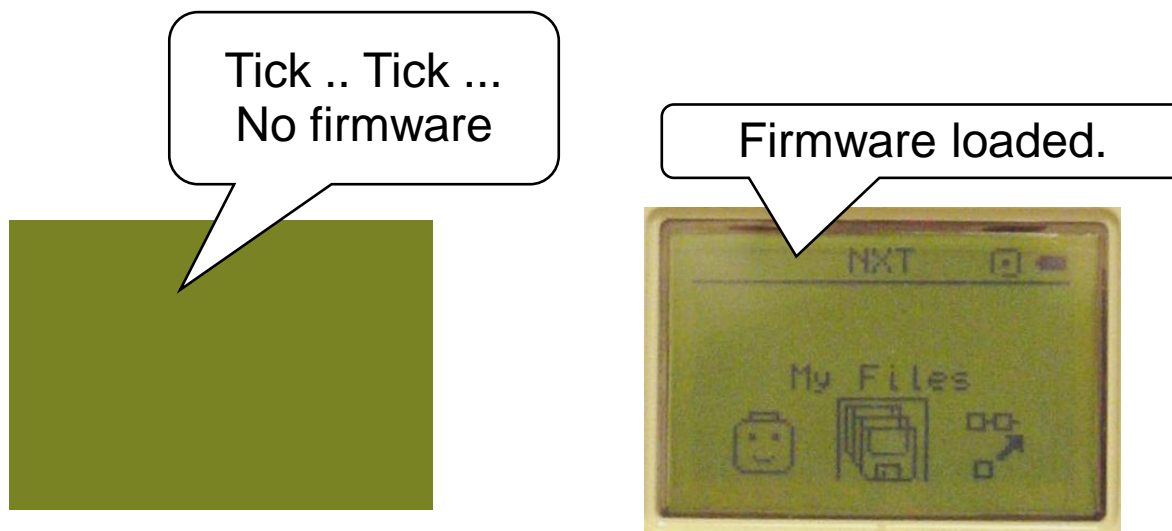


**Processor:** 32 bit ARM Atmel AT91SAM256 running at 50 Mhz

**Memory:** 64K Static RAM, 256K Flash

# טעינת קושחה

- בכדי לאפשר יכולת הרצת תוכניות על הלבנה החכמה יש לטעון ראשית את הקושחה.
- ביצוע טעינת קושחה יבוצע:
  - טעינת גרסת קושחה חדשה
  - כאשר הלבנה החכמה לא מגיבה כמו שצריך

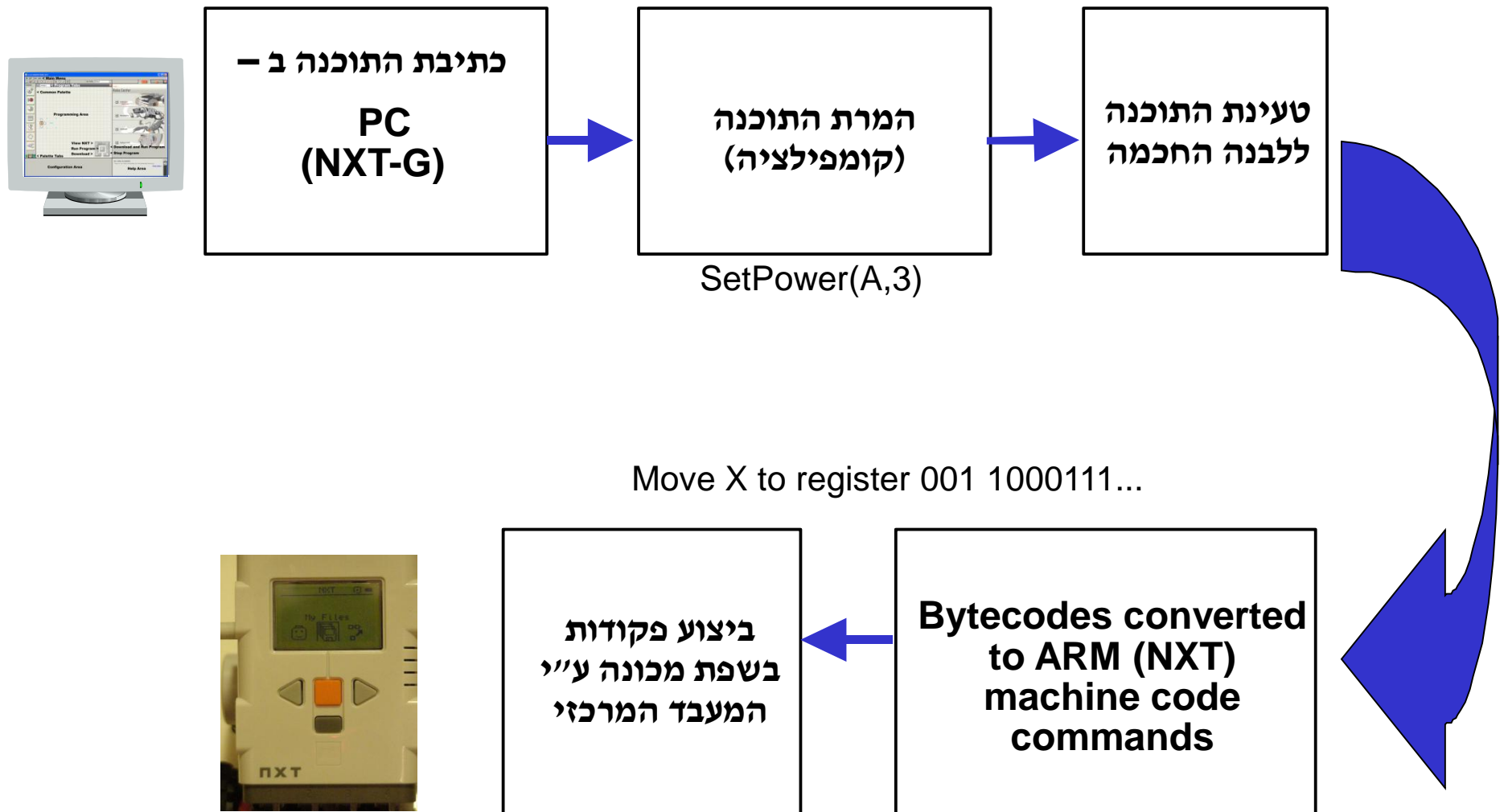


# כתיבת תוכנית מחשב

---

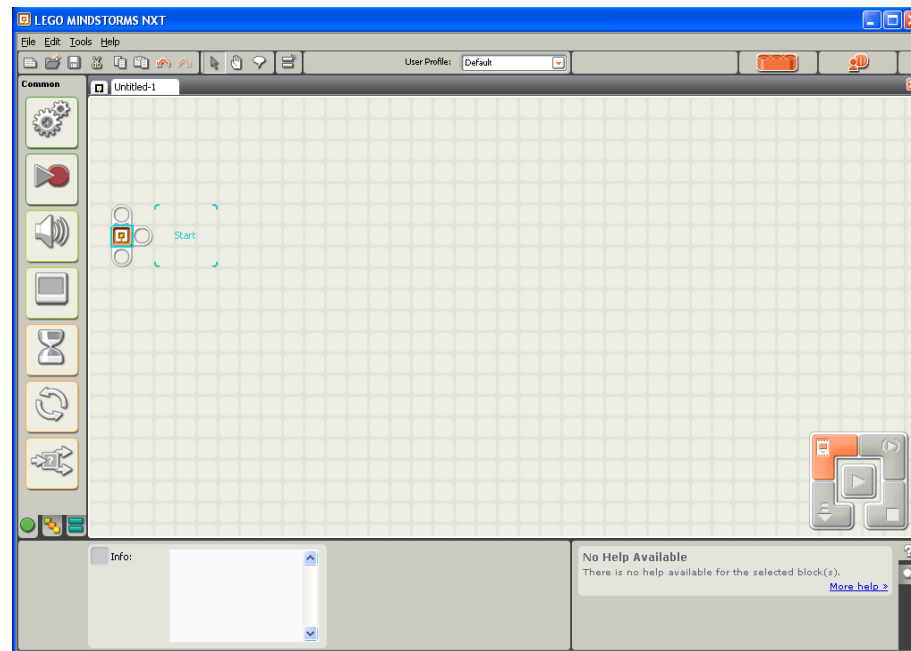
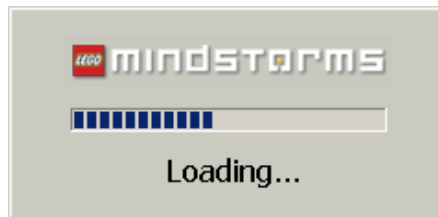
- הגדרת המשימה
  - מקורות הקלט ומה יגיע מהם
  - מקורות הפלט ומה יועבר בהם
- חשיבה על אלגוריתם למימוש
- מימוש האלגוריתם בשפת המחשב

# Running a computer program (NXT)



# NXT-G

## סביבת העבודה



# פתיחת לוח עבודה ב - NXT-G

The screenshot shows the LEGO MINDSTORMS NXT software interface. The main window is titled 'LEGO MINDSTORMS NXT' and contains a 'Getting Started' section with 'Getting Started' and 'Software Overview' buttons. A 'Robo Center' sidebar on the right lists 'Vehicles \*Quickstart', 'Machines', 'Animals', and 'Humanoids'. At the bottom, there are 'Start New Program' and 'Open Recent Program' sections. A 'Help and Zoom Panel' is visible at the bottom right.

**Profiles**

**Program Blocks**  
מרכיבי התוכנה

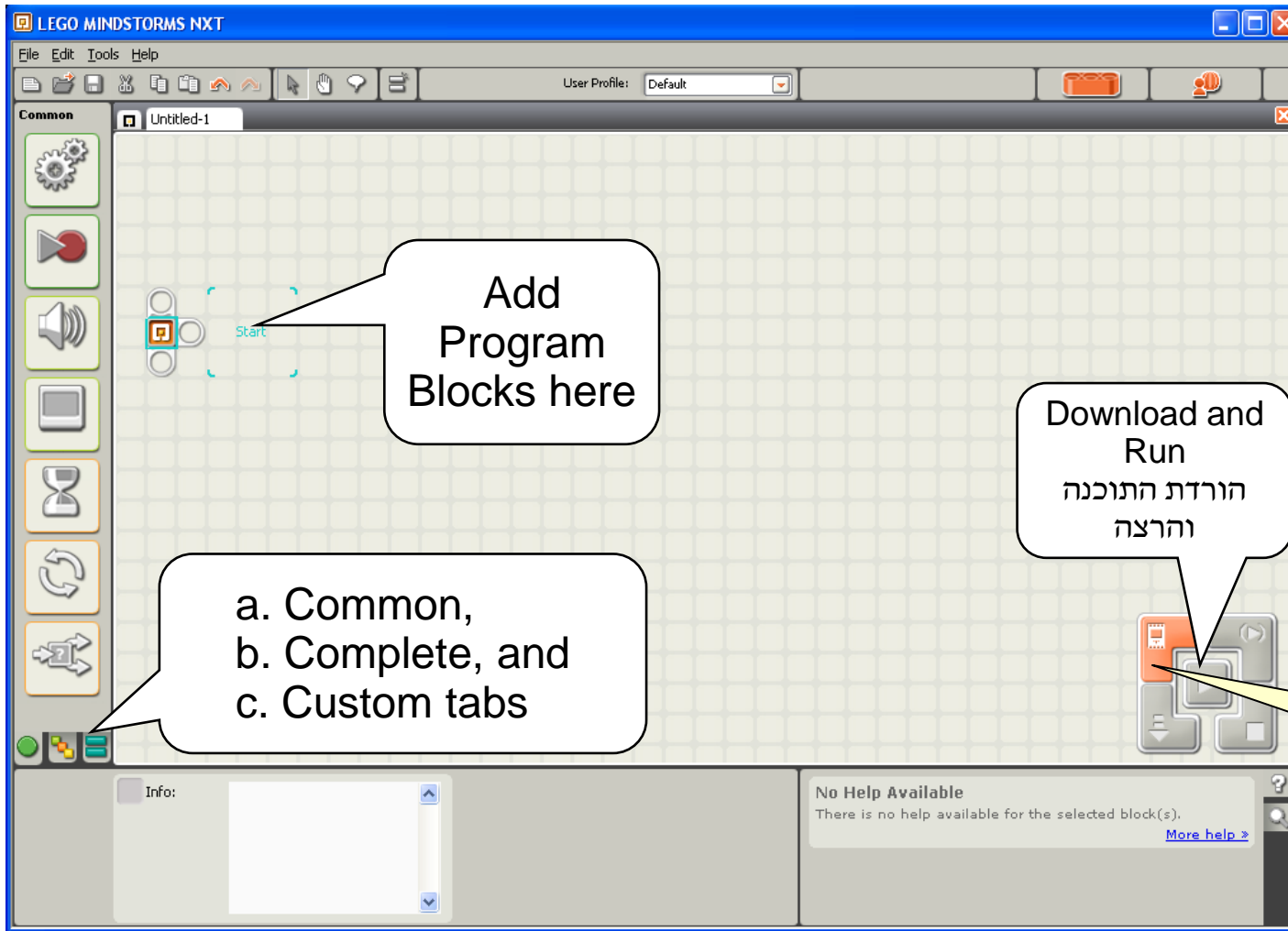
**Select a Program Name**  
בחירת שם לתוכנית

**Block Settings**  
בחירת מצבי הבלוק

**Pre-built Robots, Programs, and Challenges**

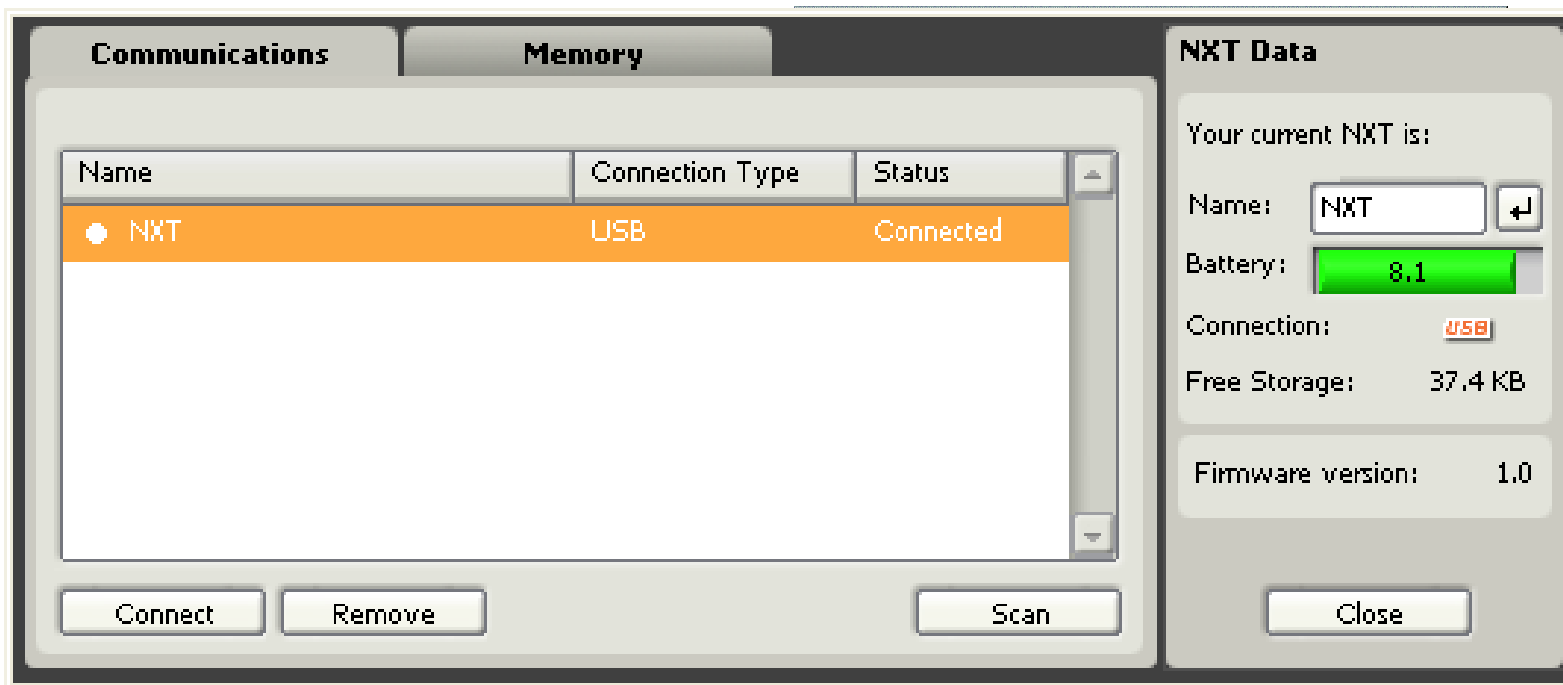
**Help and Zoom Panel**

# NXT-G Work Space



# NXT-G to NXT

## תקשורת



- כאשר מחוברת הלבנה למחשב בעזרת כבל ה-USB או ב-Bluetooth ניתן:
- להגדיר שם ללבנה החכמה
  - לבדוק את מתח הסוללה
  - לבחון את מצב הזיכרון הפנוי
  - גרסת הקושחה



# NXT-G to NXT

## הזיכרון Memory

The screenshot shows the 'Memory' tab of the NXT software interface. It is divided into three main sections: 'NXT Memory Usage', 'Show System Files', and 'NXT Data'.

**NXT Memory Usage:** A vertical bar chart showing memory allocation. The top portion is grey and labeled 'Unused'. Below it is a thin orange bar labeled 'Other'. The bottom portion is purple and labeled 'Program'. A 'Delete All' button is located below the chart.

**Show System Files:** A table listing files on the NXT device. The table has two columns: 'Name' and 'Size'. The 'TestPivot' file is selected and highlighted in orange.

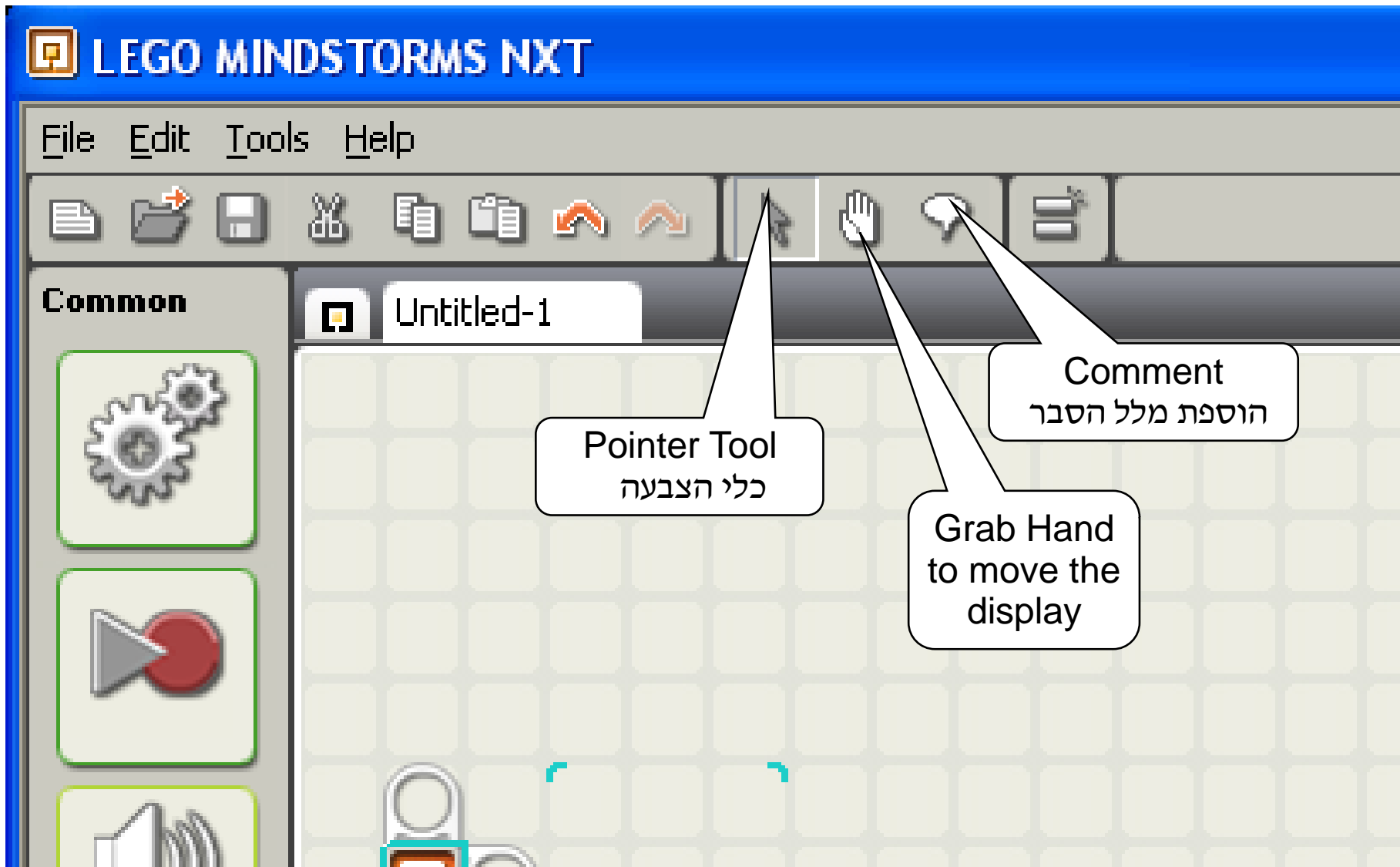
Name	Size
TestPivot	5.9 KB
MbPivotBTest	5.3 KB
UsRadarCont2	10.0 KB
Nest3	12.8 KB
SensorDisplay3	8.1 KB

Buttons for 'Upload', 'Download', and 'Delete' are located below the file list.

**NXT Data:** This section displays information about the current NXT device. It includes a text field for the name 'NXT\_DJF', a battery level indicator showing 7.4, a connection status icon, and 'Free Storage: 87.5 KB'. The 'Firmware version' is listed as 1.03. A 'Close' button is at the bottom.

יש לבחור ואז ניתן למחוק תוכניות, צלילים, גרפיקה וקבצים לא שימושיים  
ניתן לנקות עד 130Kb

# NXT-G Work Space



## טיפים (1)

---

- ללבנה החכמה יכולת לאכסן בזיכרון הרבה תוכניות
- הלבנה החכמה מבצעת כיבוי אוטומטי לטובת חיסכון



- תקשורת שן כחולה (Bluetooth)
- למערכת ה- NXT יכולת תקשורת שן כחולה. במידה ואפשרה ניתן לתקשר עם אלמנטים כמו PC ועוד.
- בזמן תחרות רצוי ומומלץ לא לאפשר את תקשורת השן הכחולה

## טיפים (2)

---

- כיוון חיבור החוטים

- חוטי החיבור למנועים ולגלאים ניתנים לחיבור רק בכיוון אחד, לא ניתן לסובבם!!

- סוללות

- AA

- בגמר הסוללות הקושחה לא נמחקת
- ניתן להשתמש אך ורק עם סוללות NiMH
- בקיט הלימודי המערכת מגיעה עם סוללת ליטיום נטענת
- יש להימנע מתפיסת המנועים בזמן הרצה הדבר מרוקן את הסוללות

# בלוקים שימושיים Common Blocks

• הבלוקים השימושיים הינם בעלי יכולות פעולה



Move Block

• בדומה למשפטים באנגלית

• Move

• Wait for an action

• Display a value

• עם אפשרות לאפיונים רבים

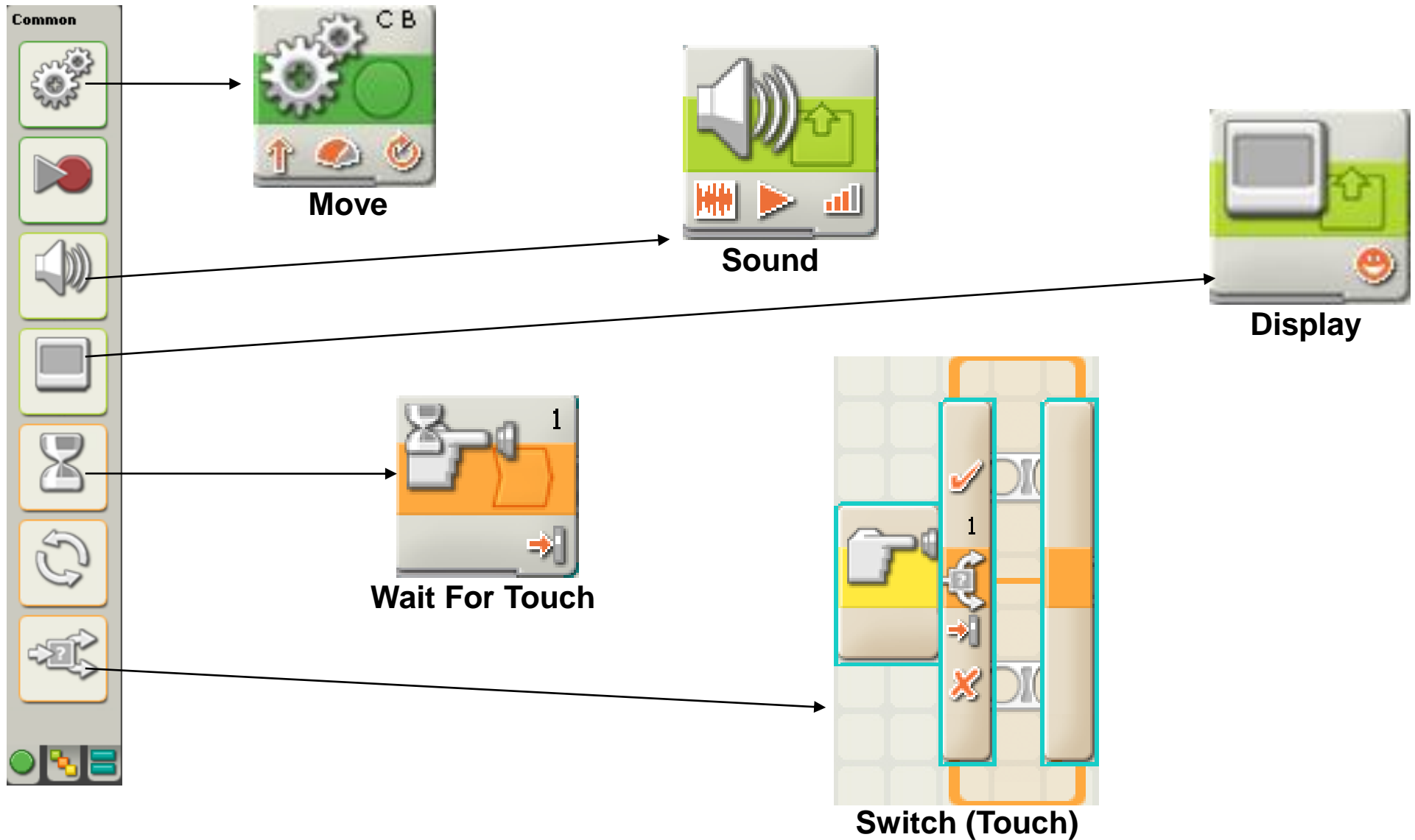
• Move direction, steering, distance, motors used . . .

• Wait for light sensor, light threshold, sensor port, . . .



Move Block Settings

# בלוקים שימושיים Common Blocks



# הוספת בלוק לתוכנית

Click a Block

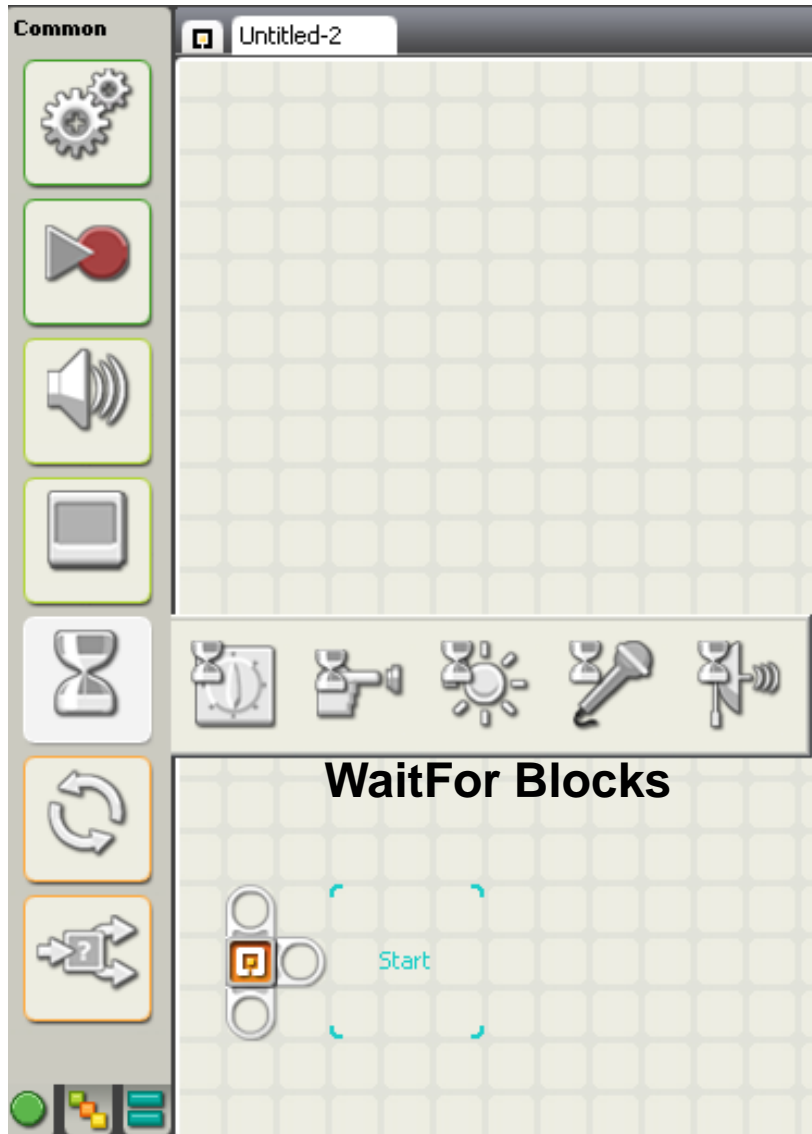
Drag it here

Release when the white position preview marks appear.

Change the settings

- הקש על הבלוק
- גרור את הבלוק לאזור התוכנה הרצוי (-NXT) G יצור מקום לבלוק (לבד)
- שנה את האפיון הרצוי

# WaitFor Blocks



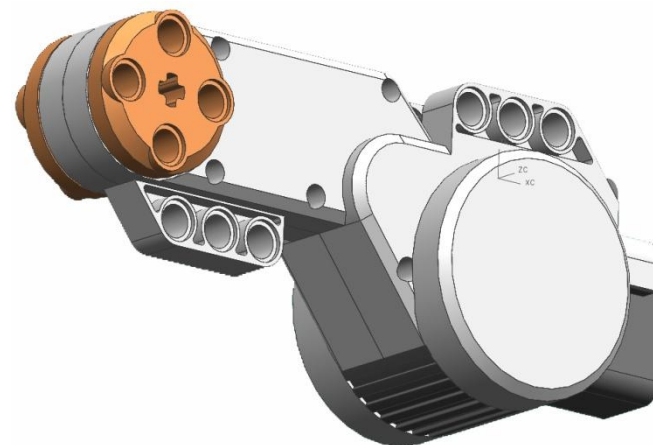
- Click on the hourglass. •
- Click on a WaitFor Block •
- Time •
- Touch •
- Light •
- Sound •
- Distance (Ultrasonic sensor) •
- Others (see settings) •



# המנועים

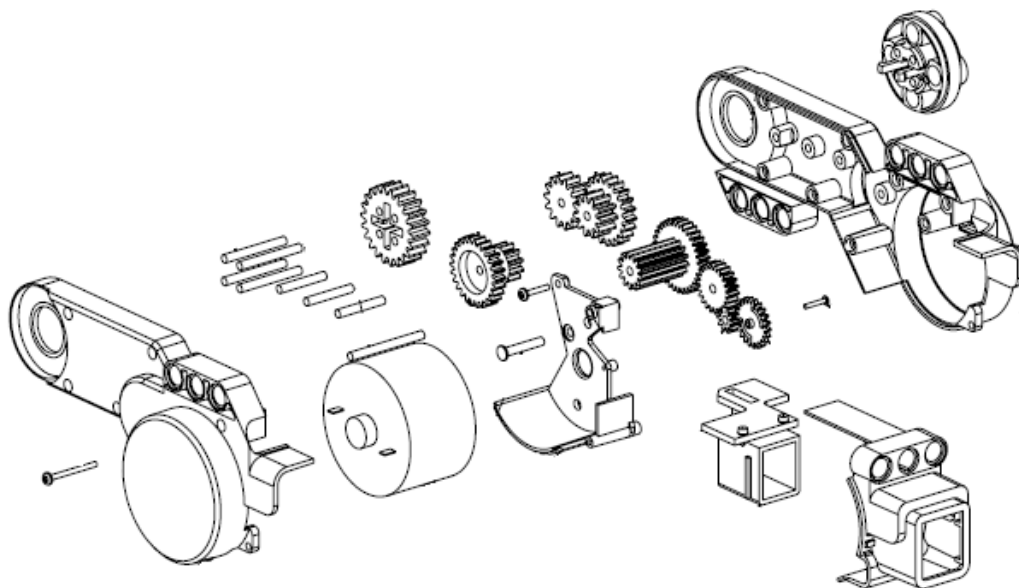
- מנועי זרם ישר 9 וולט עם תמסורת פנימית

- תנועת המנועים הינה היציאה של התוכנית (הפיכת היצירה לרובוט)
- מהירות סיבוב המנועים ללא עומס 150 rpm
- רגישות ודיוק הסרוו היא 1 מעלה
- ברובוט סטנדרטי זמן עבודת המנועים עד לריקון הסוללות הוא כ 3-4 שעות



- חוקי ה-FLL מתירים שימוש של עד שלושה מנועים!!!

# המנועים (המשך)



ניתן לשנות את עוצמת המנוע

Power levels 0-100

Power is adjusted by Pulse Width Modulation

העלאת עוצמת המנוע בהכרח גורמת לציר להסתובב  
מהר יותר

# שימוש ב - Move Block

**Ports A, B, and/or C**

**Power 0-100%**

**Duration**  
Time, Degrees, Rotations, Forever

**Forward, Backward or Stationary**

Port:  A  B  C

Direction:  Forward  Backward  Stationary

Steering:  Spin  Pivot  Arc  Straight

Next Action:  Brake  Coast

**Steering**  
Spin, Pivot, Arc, Straight  
כיוון וצורת נסיעה

**Brake or Coast**  
עצירה או שיוט

# Rotation: Move and Motor Blocks

Right Turn

The image shows a 'Right Turn' block in a programming environment. The block is highlighted with a cyan border and contains a gear icon, a green circle, and three smaller icons (a gear, a half-gear, and a full-gear). Below the block is a control panel with three rows: 'Power:' with a slider set to 75, 'Duration:' with a text box containing '400', and 'Next Action:' with a 'Brake' button. A dropdown menu is open from the '75' value, showing options: 'Unlimited', 'Degrees' (selected with a checkmark), 'Rotations', and 'Seconds'.

לכל מנוע חיישן סיבוב  
עצמי המודד סיבובים  
או מעלות



## מעבדה מס 1-



**משימה:**

**נתון מלבן ABCD**

**הנע את הרובוט מנקודה A ל-B ל-C ול-D**

**ניקוד בונים ינתן על הנעת הרובוט מ-A ל-C ל-D וחזרה ל-A**



# תשובה למעבדה מס - 1

The screenshot displays a sequence of five motor blocks in a software environment. The blocks are labeled as follows:

- A to B**: Motor block with gears B and C, and a green bar.
- Turn to C**: Motor block with gears B and C, a green bar, and a blue highlight.
- B to C**: Motor block with gears B and C, and a green bar.
- Turn to D**: Motor block with gears B and C, and a green bar.
- C to D**: Motor block with gears B and C, and a green bar.

Below the blocks is a control panel with the following settings:

- Port:** A  B  C
- Direction:**  Up  Down  Stop
- Steering:** B   C
- Power:** Slider set to 75
- Duration:** 175 Degrees
- Next Action:**  Brake  Coast

# פתרון בעיות

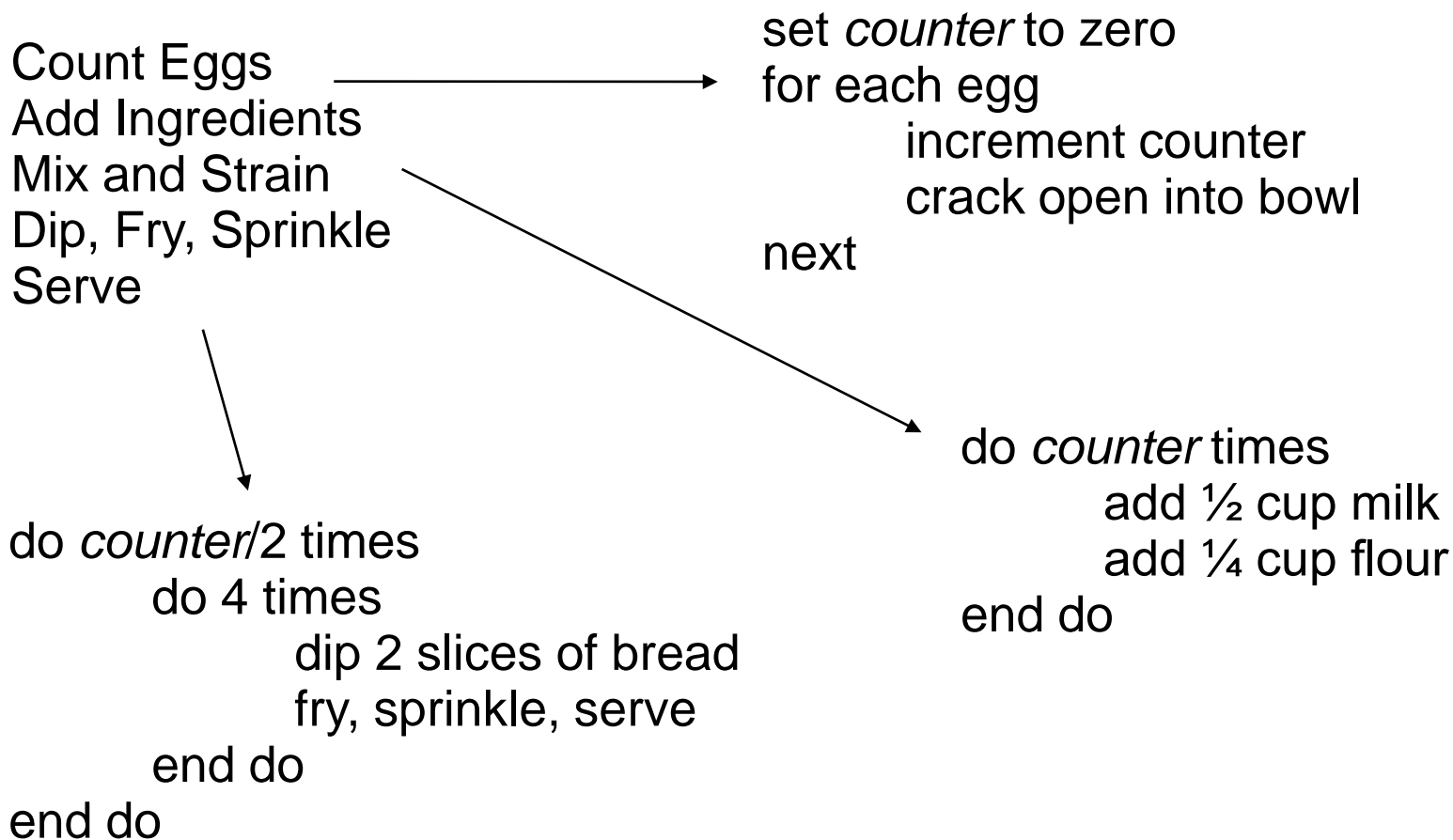
# תהליך גנרי לפתרון בעיות

- הגדרת הבעיה
- ביצוע חשיבה משותפת למציאת פתרונות
- הערכת הפתרונות ובחירה באחד
- מימוש הפתרון הנבחר
- הערכת התוצאות
- הבאת הפתרון לכדי אלגוריתם ממושט והמרתו לשפת התוכנה



# Pseudocode

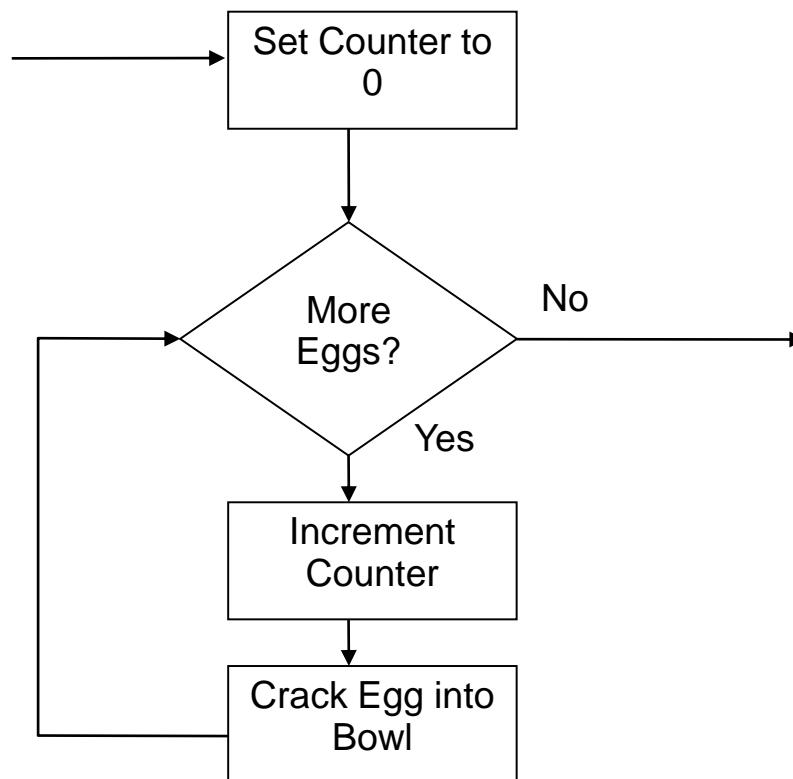
**Pseudocode:** Not software code, not English, but somewhere in between.



# תרשימי זרימה Flowcharts

תרשים זרימה:  
הצגה גרפית של הלוגיקה  
המרה מ – pseudocode  
צעד נוסף קדימה בדרך אל התוכנה

set counter to zero  
for each egg  
    increment counter  
    crack into bowl  
next



## דיבגיג (מציאת בגים) וניתוח ביצועים

- חלוקת התוכנית לקטעים.
- בחינת כל קטע בנפרד
- ביצוע קטעים קטנים
- הבאת הרובוט לנקודה הראשונה, ביצוע משימה
- שימוש בקטעי קוד שעובדים REUSE
  - נסיעה למרחק קבוע
  - פניה בזווית ידועה
- חשיבה כוללת לפתרונות חדשים
- הסתכלות מחוץ לקופסא
  - בקשת עזרה

**אסטרטגיה לשמור על פשטות**

**Keep It Simple Strategies**

**KISS #1: Subroutines תתשגרה**

**#2: Comments הערות**

**#3: Loops לולאות**

# KISS #1: Subroutines תת־שגרות



- עטיפת קטעי קוד מסובכים לחבילות קטנות
- כעת יש לבחון ברמת חבילה

- במערכת NXT-G שגרות נקראות MyBlocks
- בחירה מ - Custom Tab

# תתשגרה: מתי להשתמש

---

- כאשר מבוצעת פעולה מסוימת שוב ושוב במקומות שונים
  - Reuse
- חלוקת משימה לקטעים
  - Modules
- הסתרת קטעים מורכבים

# שמות ל - MyBlock Names



שימושיים ובעלי משמעות הנותנת מידע •

ClearSoccerField *not* Csf\_amy\_3a •

12 characters visible on a MyBlock •

15 characters visible on the NXT

Suggest using “action + to + target”: •

Fwd2Wall or ForwardToWall or Forward\_To\_Wall •

FwdDist •

TurnRight •

Name the task accomplished, not how it was done. •

FollowLine *not* FollowLine1LightSensor •

# MyBlock Creation 1

- For each parameter, add a variable.

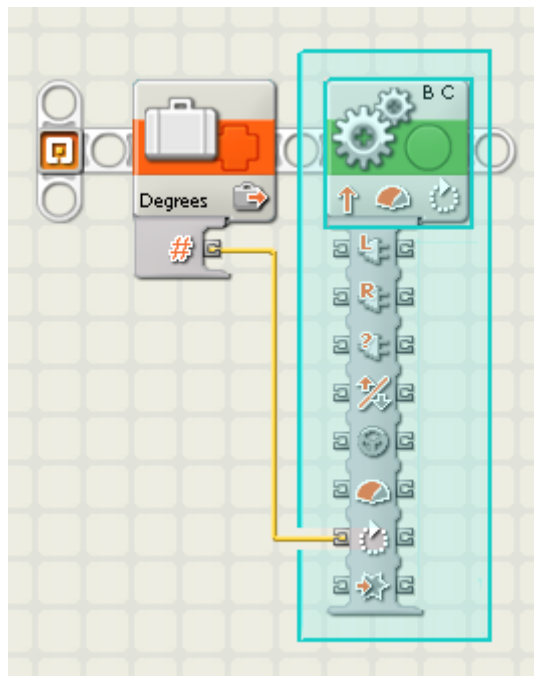
The screenshot shows the MyBlock creation environment. On the left, a palette contains various blocks, including a 'Degrees' variable block. A yellow wire connects this block to a 'BC' (Basic Control) block in the main workspace. A callout bubble points to the 'Degrees' block with the text: "Click here to reveal wire connections." To the right, the 'Edit Variables' dialog box is open, showing a list of variables: Logic 1 (Logic), Number 1 (Number), Text 1 (Text), and Degrees (Number). The 'Degrees' variable is selected. Below the list, the 'Name' field contains 'Degrees' and the 'Datatype' dropdown is set to 'Number'. Callout bubbles point to the dialog with the text: "Create, name, and select the datatype." At the bottom left, a 'Variable' panel shows a similar list of variables, with 'Degrees' selected. To its right, the 'Action' section has 'Read' selected and 'Write' unselected, and the 'Value' field is set to 0.

Add and wire Variable Blocks to your program. •

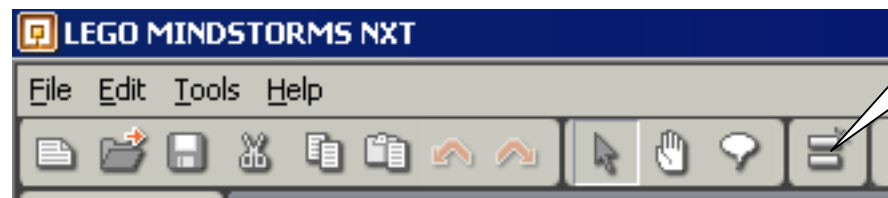


# MyBlock Creation 2

Select blocks to include.



Click the Create My Block button.

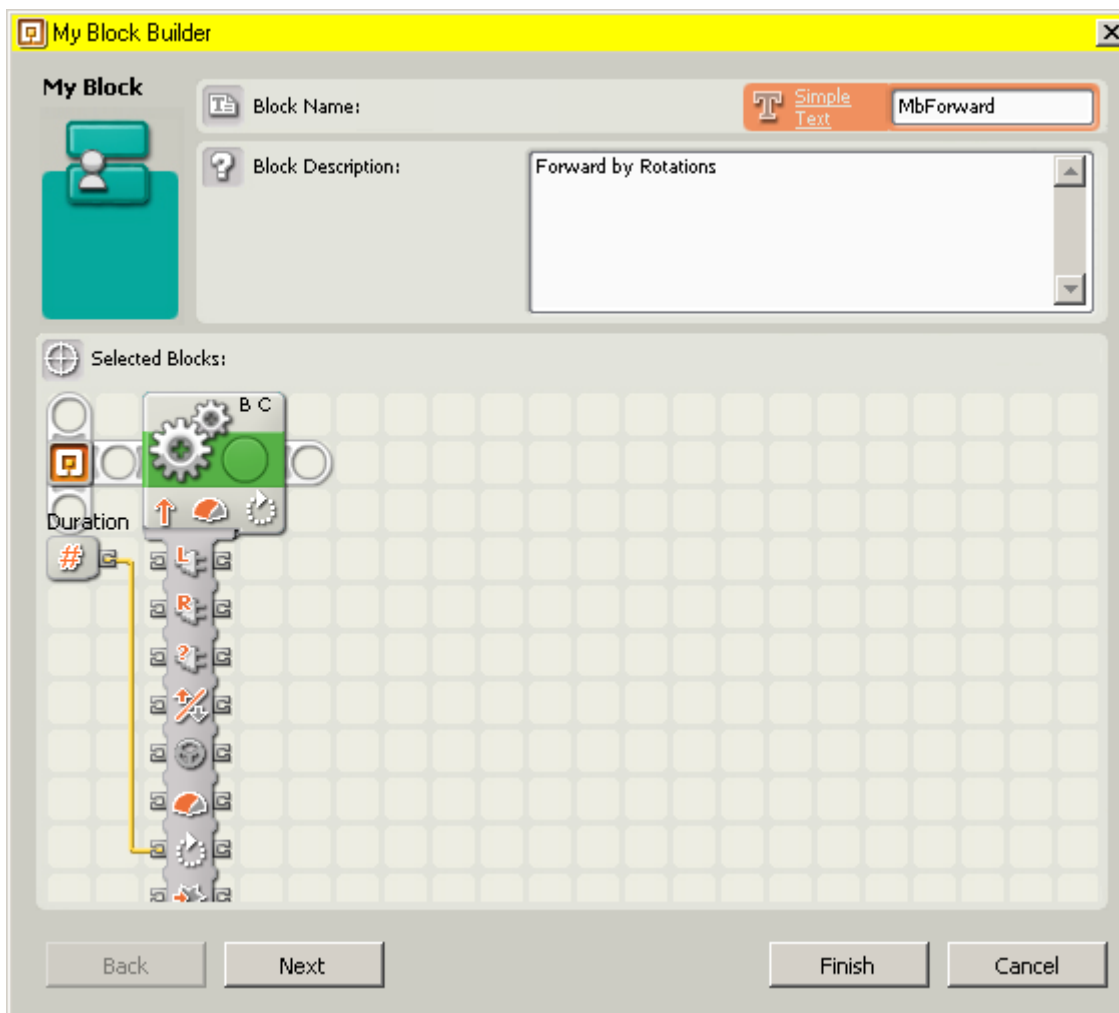


Create My Block button

# MyBlock Creation 3

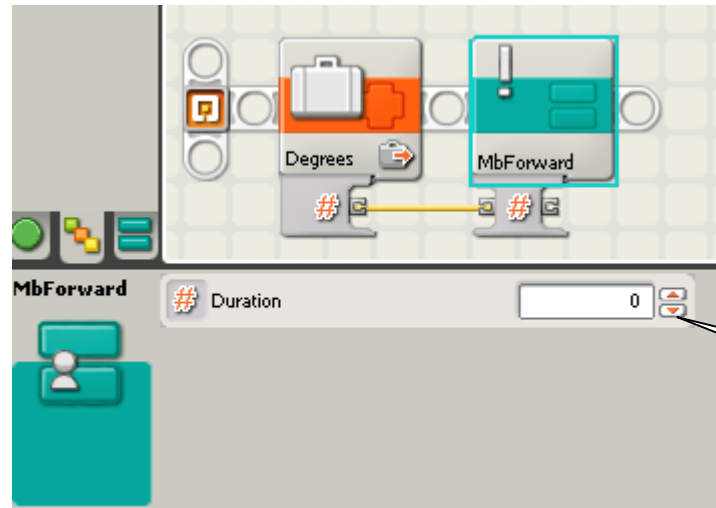
- Name the MyBlock.
- Give it a description.

- Click "Next" to add an icon.



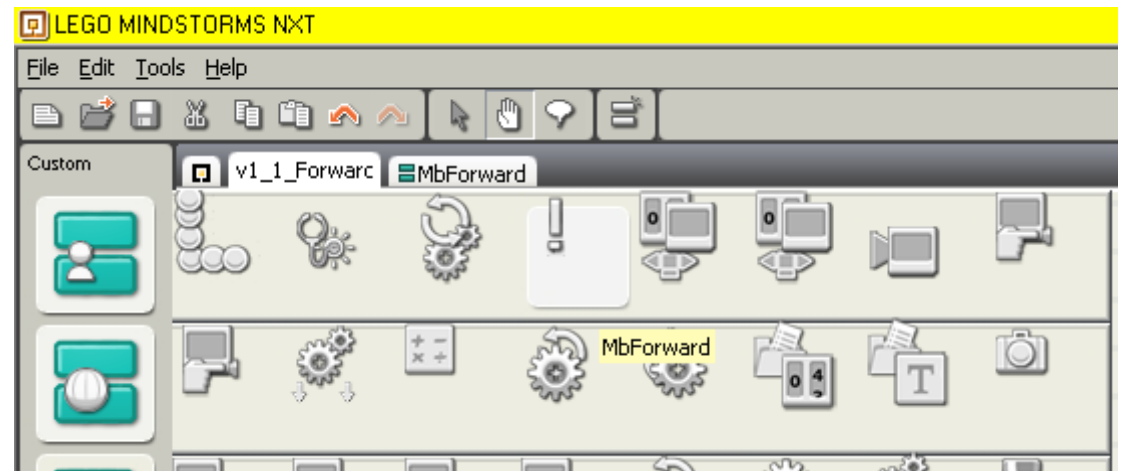
# MyBlock Creation 4

MyBlock replaces the selected Blocks!



Can only set the duration.

To add the new MyBlock to a program, select it from the Custom Tab.



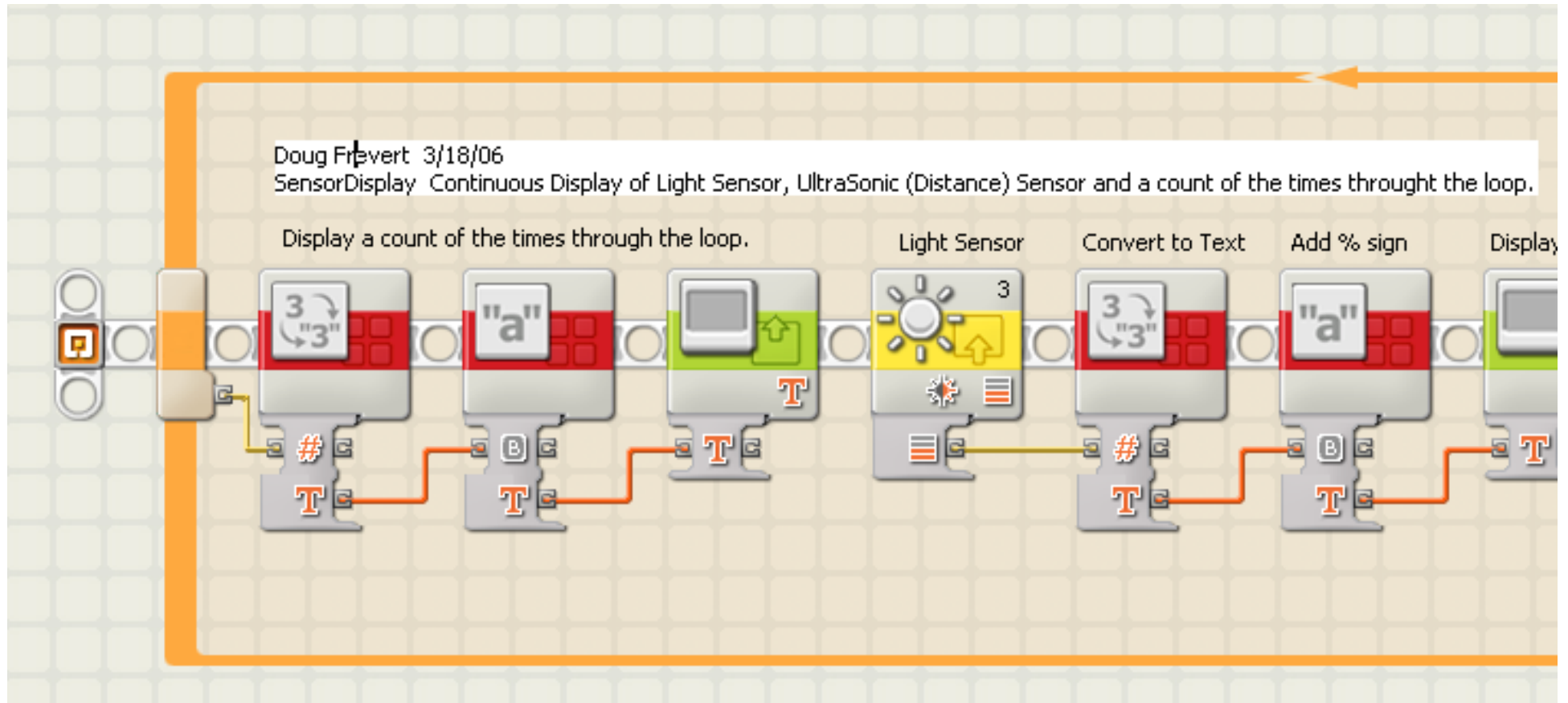
# KISS #2: Comments

---

- Explain the program.
- Programmers forget.
- Teams compete in FLL. More than one person will be working on the program.



# שימוש בהערות



- הוסיפו מי, מתי, כיצד להשתמש בו, ההנחות שנלקחו, ואת התוצאות המצופות.

# KISS #3: לולאות



ה **Loop Block** : טפות תכנות אחרות :

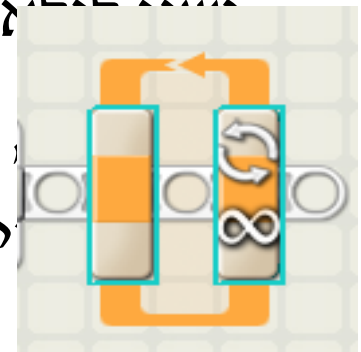
**For ... Next** עשה את הלולאה  $n$  פעמים.

**Do ... Until** עשה זאת. אם לא אובחן משהו, עשה זאת שוב.

עשה לולאות ל... :

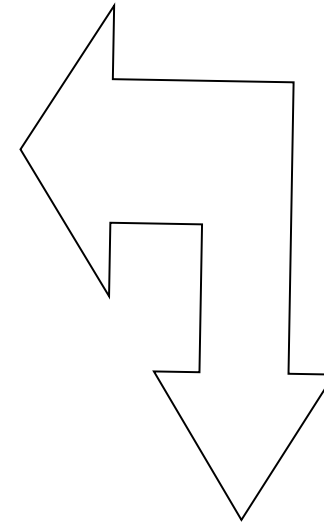
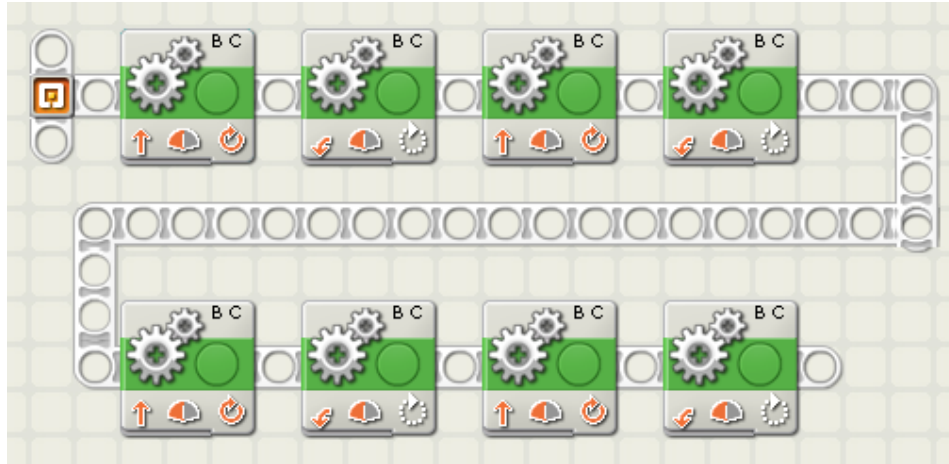
לעד.

(ישן) כולל זמן)

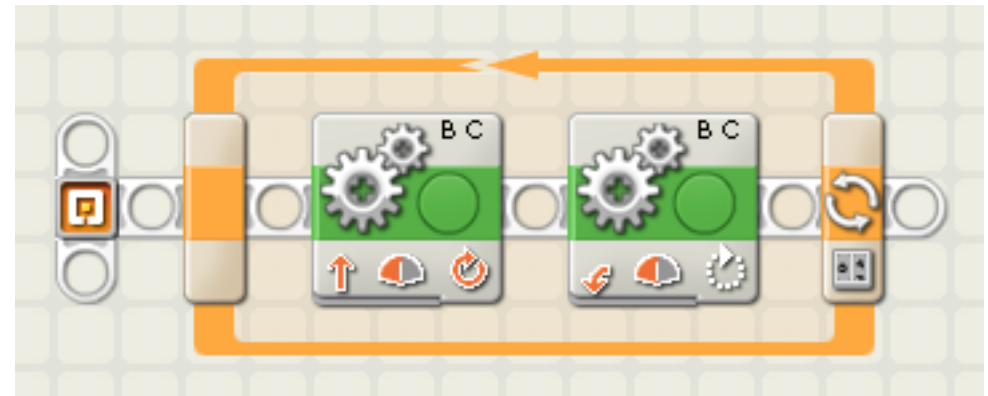


• ספירה

# לולאה פשוטה



- המר זאת
- לדבר פשוט יותר, באמצעות לולאה.





## מעבדה 2



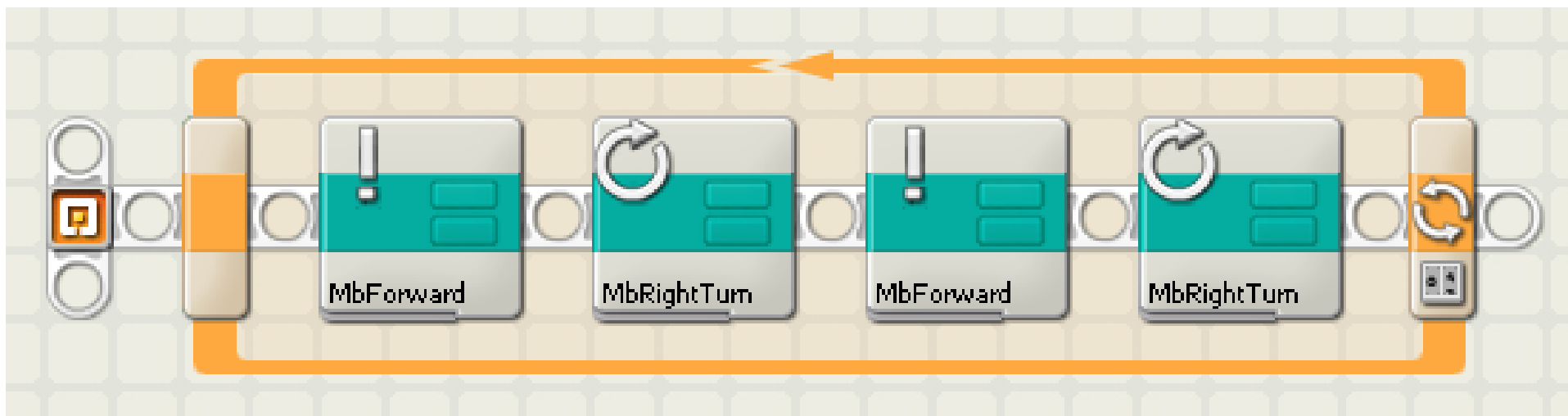
### משימה:

צרו בלוקים של נסיעה ישר ושל פנייה ימינה.  
נועו סביב המלבן פעמיים. סיימו את הפעולה בנקודה A.





## מעבדה 2 - תשובה



תשובה אפשרית אחת.