

MODULE 3

Technology inzetten voor het automatisch rijden

Naam: Bram van Gils
PCN: 401949

Contents

Module 2.1	2
Code	2
Module 2.2 – 2.5	3
Flow-Chart.....	3
Code:	4
Variabelen en CheckPaused Method.....	4
CalibrateValues Method	4
PowerMotors + StopMotors Method	4
Evade Method.....	5
Main Method	5
Bronnen.....	6

Module 2.1

Code

```
while (true)
{
    // Read input message.
    // readMailboxIn() is non-blocking and returns "" if there is no message.
    readMailboxIn("EV3_INBOX0", msgBufIn);
    if (strcmp(msgBufIn, "") != 0)
    {
        if(strcmp(msgBufIn, "Forward") == 0)
        {
            displayBigTextLine(8, "Forward");
            setMotor(motorB, 50);
            setMotor(motorC, 50);
        }

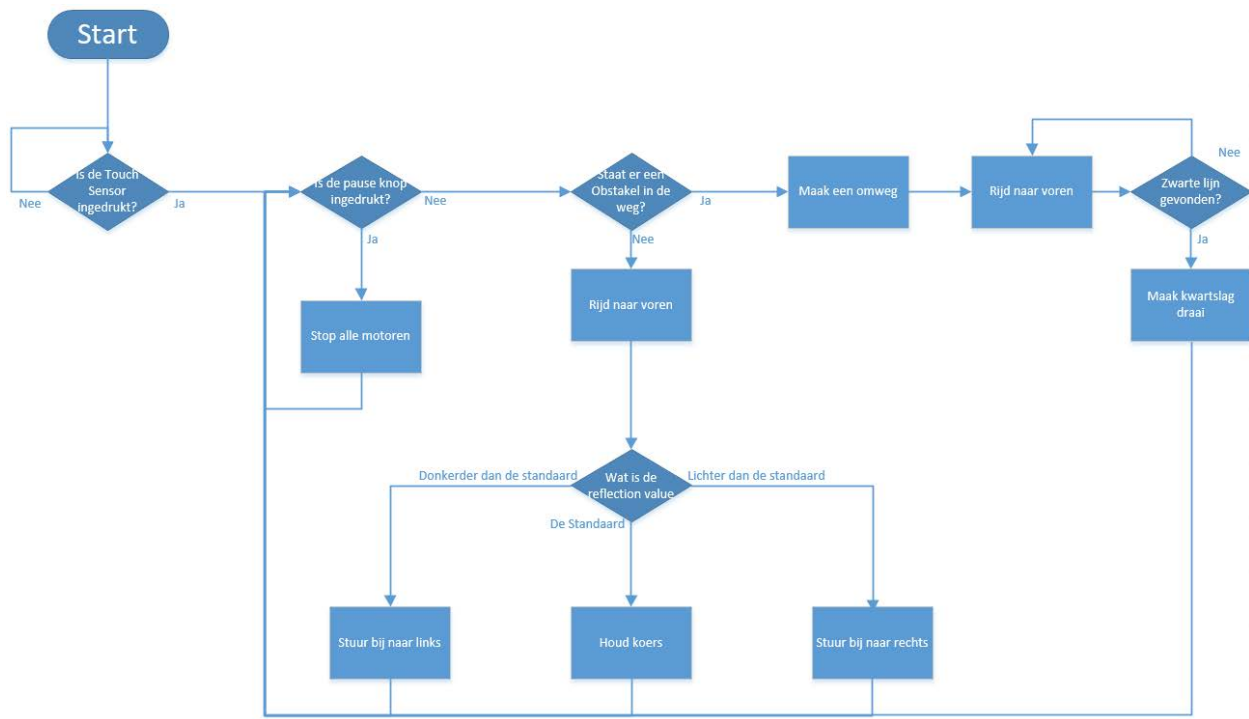
        if(strcmp(msgBufIn, "Backward") == 0)
        {
            displayBigTextLine(8, "Backward");
            setMotor(motorB, -50);
            setMotor(motorC, -50);
        }

        if(strcmp(msgBufIn, "Left") == 0)
        {
            displayBigTextLine(8, "left");
            setMotor(motorB, 0);
            setMotor(motorC, 50);
        }

        if(strcmp(msgBufIn, "Right") == 0)
        {
            displayBigTextLine(8, "Right");
            setMotor(motorB, 50);
            setMotor(motorC, 0);
        }
    }
}
```

Module 2.2 – 2.5

Flow-Chart



Link voor betere Weergave:

<https://drive.google.com/open?id=1mMaBmnQnCb5VIEaaypnY8kOFj9mnrW-c>

Code:

Variabelen en CheckPaused Method

```
int DarkestValue = getColorReflected(S3);
int LightestValue = getColorReflected(S3);
int StandardLightValue;
int MaxEnginePower = 10;
int Offset;
int HalfRange;

bool ButtonState;
bool PauseState = false;
|
void CheckPaused()
{
    if(getButtonPress(buttonEnter) && !ButtonState) // check of er een verandering is in de button state
    {
        PauseState = !PauseState;
    }

    ButtonState = getButtonPress(buttonEnter);
}
```

CalibrateValues Method

```
void CalibrateValues()
{
    if (DarkestValue > getColorReflected(S3)) // stel de donkerste waarde bij
    {DarkestValue = getColorReflected(S3);}

    if (LightestValue < getColorReflected(S3)) // stel de lichtste waarde bij
    {LightestValue = getColorReflected(S3);}

    StandardLightValue = (LightestValue + DarkestValue) / 2; // vind de algemene waarde die gevolgd wordt
    HalfRange = (LightestValue - DarkestValue) / 2; // vind de afwijkings range
    Offset = StandardLightValue - getColorReflected(S3); // vind de mate van afwijking

    displayBigTextLine(8,"Offset = %d", Offset); // debug
}
```

PowerMotors + StopMotors Method

```
void PowerMotors()
{
    CalibrateValues();

    if(Offset < 0)
    {
        setMotor(motorC,(1+Offset / HalfRange) * MaxEnginePower); //Stuur naar rechts
        setMotor(motorB,MaxEnginePower);
    }
    else if(Offset > 0)
    {
        setMotor(motorC,MaxEnginePower);
        setMotor(motorB,(1-Offset /HalfRange) * MaxEnginePower); //Stuur naar links
    }
    else
    {
        setMotor(motorB,MaxEnginePower);
        setMotor(motorC,MaxEnginePower); // recht vooruit
    }
}

void MotorStop()
{
    setMotor(motorB, 0);
    setMotor(motorC, 0);
}
```

Evade Method

```
void Evade()
{
    setMotorSyncEncoder(motorB, motorC, -100, 180, 20); // maak kwartslag draai.
    waitUntilMotorStop(motorB);
    setMotorSyncEncoder(motorB, motorC, 0, 540, 20); // rijd 50 cm naar voren.
    waitUntilMotorStop(motorB);
    setMotorSyncEncoder(motorB, motorC, 100, 180, 20); // maak kwartslag draai.
    waitUntilMotorStop(motorB);
    setMotorSyncEncoder(motorB, motorC, 0, 540, 20); // rijd 50 cm naar voren.
    waitUntilMotorStop(motorB);
    setMotorSyncEncoder(motorB, motorC, 100, 180, 20); // maak kwartslag graden draai.
    waitUntilMotorStop(motorB);

    setMotor(motorB, 20);
    setMotor(motorC, 20);

    while(getColorReflected(S3) >= StandardLightValue) // zoekzwarte lijn
    {}

    setMotorSyncEncoder(motorB, motorC, -100, 180, 20); // maak kwartslag draai.
    waitUntilMotorStop(motorB);
}
```

Main Method

```
task main()
{
    while (getTouchValue(S1) == 0) {}

    while(true)
    {
        CheckPaused();

        if (!PauseState)
        {
            PowerMotors();
        }
        else
        {
            MotorStop();
        }

        if(getUSDistance(S4) < 20)
        {
            Evade();
        }

        delay(10);
    }
}
```

Bronnen

<https://www.youtube.com/watch?v=BjVeWRNiddE>

<https://stackoverflow.com/questions/16627588/what-is-the-correct-string-terminator-in-c>

https://www.tutorialspoint.com/cprogramming/c_arrays.htm

https://www.le.ac.uk/users/rjm1/cotter/page_22.htm

https://www.tutorialspoint.com/cprogramming/c_variables.htm

<https://www.geeksforgeeks.org/understanding-extern-keyword-in-c/>

<https://stackoverflow.com/questions/15707732/what-does-array-means>