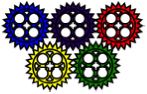
**Нижегородский Институт Информационных Технологий**



**Программа «Робототехника. Инженерно-технические кадры инновационной России»**



**Международные Состязания Роботов 2012**



**Проект «Ученик Барабанщика»**

**Авторы: Петр Изюмкин**

**Сергей Куленков**

**Руководитель: Колотов А.В.**



**Описание:**

Прежде, чем готовить проект к творческой категории Международных Сотсязаний Роботов, мы проанализировали какими способностями должны обладать роботы, чтобы быть принятыми в мире Людей. Наиболее интересными из них, нам показались следующие: способность выражать эмоции, способность обучаться и способность к творчеству.

В связи с этим, возникла идея разработки такого робота, который использовал эти три грани человеческой натуры.

В итоге, родился проект Ученик Барабанщика. Данный робот способен слушать учителя, запоминая набиваемый им ритм. Затем, ему надо повторить «пройденный урок». Как результат, робот ожидает поощрения в виде апплодисментов, которые влияю т на эмоции, выражаемые в итоге на лице робота.

**История создания:**

Изначально (на региональном этапе) робот состоял из двух независимых частей – головы, выражающей эмоции в зависимости от оценок, и робота-барабанщика, умевшего воспроизводить как заранее заложенные ритмы, так и работоть в режиме обучение-воспроизведение.

|  |  |
| --- | --- |
| https://lh3.googleusercontent.com/-eIcngFqdRcU/T16Fbf_X_HI/AAAAAAAABqE/60EemxvMVdU/s576/_MG_5714.jpg | https://lh5.googleusercontent.com/-j-6InQQPCak/T16QGxvDi6I/AAAAAAAACME/PHQ5qJB2rA4/s912/_MG_5908.jpg |

**Принцип действия:**

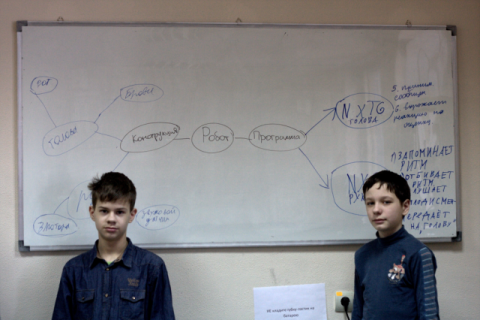
Алгритм работы робота можно разделить на следующие основные шаги:

Фаза обучения и воспроизведения:

1. Ожидать появления звуковых всплесков и запоминать интервалы между ними. Всего может быть запомнено до 100 элементов ритма.
2. Как только, обнаружился конец в изучаемом ритме, воспроизвести его, выделяя отдельные части ритма для каждой руки робота.
3. Ожидать появления апплодисментов.
4. Если апплодисментов не последовало, послать негативный сигнал через Bluetooth на «голову»
5. Если звучат апплодисменты, послать положительный сигнал через Bluetooth на «голову».
6. Дождаться окончания апплодисментов и перейти к обучению следующего ритма.

Асинхронная фаза выражения эмоций:

1. Принять сигнал через Bluetooth от «рук»
2. Используя конечный автомат, определить какую из трех эмоций необходимо отобразить по принятому сигналу. Возможные эмоции: грусть, нейтральная эмоция, радость.
3. Отобразить эмоцию, если состояние конечного автомата изменилось.
4. Ожидать прихода нового сигнала.



**Технические характеристики:**

Конструктивно, робот состоит из рук и головы. За каждый элемент отвечает отдельный NXT блок. Между блоками установленно Bluetooth соединение.

Руки выполнены из трех сервомоторов и одного датчика звука. Каждый сервомотор отвечает за соответствующую руку. Сервомоторы закреплены на подвижных платформах для того, чтоб обеспечить возможность игры на любых барабанах. Программное обеспечение для этой части конструкции написано на языке Not Exactly C.

Голова выполнена из двух моторов, один из которых отвечает за движение губ, другой – за движение бровей. Программное обеспечение головы выполнено в среде NXT-G.

|  |  |
| --- | --- |
|  |  |

**Решенные проблемы:**

1. В начале конечный автомат, ответственный за лицо робота, контролировал 6 различных эмоций. Оценка робота на публике показала, что количество эмоций для улучшения презентативного вида должно быть сокращено до трех.
2. В первой версии алгоритма обнаружения апплодисментов, из-за его неправильной работы, слишком долгие апплодисменты расценивались роботом, как следующий урок, как ритм, который должен быть запомнен. В последней версии алгоритма, робот не переходит в режим обучения, до тех пор, пока апплодисменты не закончатся.

**Возможные улучшения:**

1. Конструкция робота и алогритм могут быть доработаны для того, чтобы запоминать урок для каждой из рук по отдельности. Это приведет к тому, что робота можно будет обучать игре довольно сложным музыкальным партиям.
2. Алогритм работы может быть усовершенствован, чтобы попытаться сделать выводы почему игра робота получили отрицательную оценку. Данный вывыод будет направлен на изменение внутренних переменных алгоритма обучения: чувствительность датчика звука, изменение в скорости работы моторов и т.п.
3. Режим обучения может быть доработан таким образом, что робот будет запоминать все партии, получившие хорошую оценку, а затем прибавлять одну или несколько случайных партий во время озвучивания очередного урока. Таким образом, будет возможность эмулировать способность робота импровизировать.

**Исходные коды:**

Программа для элемента «руки».

*//Master*

#define BT\_CONN 1

#define OUTBOX 0

#define K\_SLOW 1.3

#define SOUND\_TRSH 60

**int** V[100];

**int** y = 1;

**sub** BTCheck(**int** conn) {

**if** (!BluetoothStatus(conn)==NO\_ERR) {

TextOut(5,LCD\_LINE2, *"ERROR"* );

Wait(1000);

Stop(**true**);

}

}

**task** Left() {

PosRegSetAngle(OUT\_C, -35);

Wait(V[0]\*K\_SLOW);

**for** (**int** n=1; n<y; n=n+3) {

*//Up*

Wait(V[n]\*K\_SLOW);

PosRegSetAngle(OUT\_C, 0);

Wait(120);

PosRegSetAngle(OUT\_C, -35);

Wait((V[n+1]+V[n+2]-120)\*K\_SLOW);

V[n] = 0;

}

}

**task** Middle() {

PosRegSetAngle(OUT\_A, -35);

Wait((V[0]+V[1])\*K\_SLOW);

**for** (**int** n=2; n<y; n=n+3) {

*//Up*

Wait(V[n]\*K\_SLOW);

PosRegSetAngle(OUT\_A, 0);

Wait(120);

PosRegSetAngle(OUT\_A, -35);

Wait((V[n+1]+V[n+2]-120)\*K\_SLOW);

V[n] = 0;

}

}

**task** main() {

**int** n,v;

SetSensor(IN\_4, SENSOR\_SOUND);

BTCheck(BT\_CONN); *// check slave connection*

OnFwd(OUT\_ABC, 25);

Wait(500);

SetMotorRegulationTime(MS\_10);

SetMotorRegulationOptions(OUT\_REGOPTION\_NO\_SATURATION);

PosRegSetMax(OUT\_B, 0, 15);

PosRegSetMax(OUT\_C, 0, 15);

PosRegSetMax(OUT\_A, 0, 15);

PosRegEnable(OUT\_B);

PosRegEnable(OUT\_C);

PosRegEnable(OUT\_A);

PosRegSetAngle(OUT\_B, -5);

PosRegSetAngle(OUT\_C, -5);

PosRegSetAngle(OUT\_A, -5);

SendRemoteString(BT\_CONN,OUTBOX, *"2"*);

**while** (1)

{

ClearScreen();

PlayTone(440,250);

TextOut(10,LCD\_LINE1,*"Ready to study"*);

y = 1;

**do** {

n = 0;

**while** (n < 150)

{

**if** (SENSOR\_4 > (SOUND\_TRSH+5)) {

*//NumOut(50, 64-y\*8, SENSOR\_4);*

V[y-1]=n\*10;

**until**(SENSOR\_4 < (SOUND\_TRSH-5));

n=0;

y++;

}

**else** {

n++;

Wait(10);

}

}

} **while** (y == 1);

y--;

PlayTone(440,250);

TextOut(10,LCD\_LINE2,*"Try to repeat"*);

*//until(SENSOR\_4 == 1);*

StartTask(Left);

StartTask(Middle);

PosRegSetAngle(OUT\_B, -35);

**for** (n=0; n<y; n=n+3) {

*//Up*

Wait(V[n]\*K\_SLOW);

PosRegSetAngle(OUT\_B, 0);

Wait(120);

PosRegSetAngle(OUT\_B, -35);

Wait((V[n+1]+V[n+2]-120)\*K\_SLOW);

V[n] = 0;

}

Wait(1000);

StopTask(Left);

StopTask(Middle);

TextOut(10,LCD\_LINE3,*"Waiting for applause"*);

n=0;

**while** (n<300) {

**if** (SENSOR\_4 > (SOUND\_TRSH+5)) {

TextOut(10,LCD\_LINE4,*"Positive"*);

SendRemoteString(BT\_CONN,OUTBOX, *"2"*);

TextOut(10,LCD\_LINE5,*"Noise detected"*);

**do** {

**until**(SENSOR\_4<(SOUND\_TRSH-5));

TextOut(10,LCD\_LINE6,*"Silence"*);

n=0;

**do** {

Wait(10);

n=n+1;

v=SENSOR\_4;

**if** (v>(SOUND\_TRSH+5)) { **break**; }

} **while** (n<100);

} **while** (n<100);

**break**;

}

n++;

Wait(10);

}

**if** (n==300) {

TextOut(10,LCD\_LINE4,*"Negative"*);

SendRemoteString(BT\_CONN,OUTBOX,*"1"*);

Wait(1000);

}

}

}

Программа для элемента «Голова».