Smart device

This parameter is used to read or write the control table value of the smart app (R + Smart, R + IoT) connected to the controller and Bluetooth. For more information, see the Smart Control Table. R + Smart

1 Camera 1 Camera selection / zoom camera / camera sensor / Photography / video recording;

This parameter is for using the camera function of smart device.

- **Camera Selection** : Select a camera to use of the built-in camera on smart devices.
- The example below shows alternating between rear camera and front camera.

- **Camera zoom** : allows you to magnify a smart camera of the device.
  (The range of values is from 0 to 255.)

- Below is an example of zooming the camera once every 1.024 seconds.

```
프로그램 시작
{
    조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )

    SMART: 카메라 선택 = 후면 카메라 (1)

    무조건 반복
    {
        횟수 반복 ( 반복횟수 = 0 ~ 5 )
        {
            SMART: 카메라 확대 = 반복횟수

            타이머 = 1.024초
            조건 대기 ( 타이머 > 0.000초 )
        }
    }
}
```

- **Camera Sensor** : used to operate the camera with smart devices, sensors mode.

(For more information on camera sensor mode, please refer to Vision.)

2 Camera 2 [Recording pictures / Recording movies]

This parameter is for using the camera function of smart device.

- **Photo taken** : Use when taking pictures with the camera of smart devices.
  (Shooting when True, Shooting when False)
- Below is an example of taking pictures using the rear camera of the smart device.

3 VISION [Detected color / line detection area / face detection area / motion detection area]

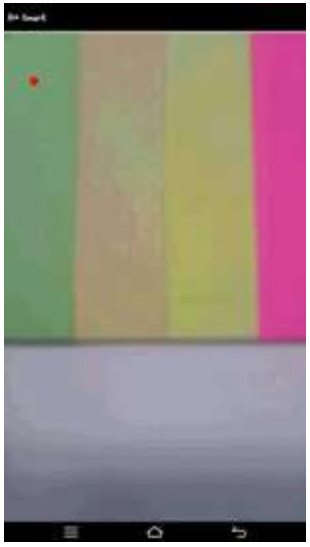
This parameter is used when setting the camera of the smart device to "Camera sensor".

.

- **Perceived colors** : If you are using the "Color Detection Mode" in the "camera sensor", is used to determine the colors that appear in the middle of the screen.
- Below is an example that uses detected color values.

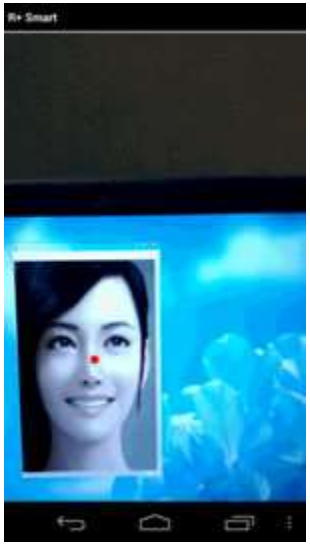
- **To detect the color line** : If you are using a "camera sensor," "Line Detection Mode" on, used to detect the color of the line.
- **Sensing area** : If you are using a "line detection mode" in the "camera sensor", is used to determine the position of the sensing line.
- Below is an example of displaying a red circle on a line when a green line is detected.

```
1  프로그램 시작
2  {
3      조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )
4
5      SMART: 카메라 선택 = 후면 카메라 (1)
6      SMART: 카메라 센서 = 라인 감지 모드 (4)
7      SMART: 감지할 라인 색상 = 녹색 라인 (4)
8
9      무조건 반복
10     {
11         라인감지위치 = SMART: 라인 감지 영역
12
13         // 화면에 표시한 빨간색 원을 모두 지움
14         SMART: 도형 표시 = 0
15         // 라인 감지 위치에 빨간색 원을 표시
16         SMART: 도형 표시 = [위치:라인감지위치],[아이템:1],[크기:10],[색상:빨간색]
17     }
18 }
```



- **Face Detection Area** : If you are using a "Face Detection Mode" in the "camera sensor", is used to determine the position of the detected face.
- Below is an example of displaying a red circle at the position when a face is detected.

```
1  프로그램 시작
2  {
3      조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )
4
5      SMART: 카메라 선택 = 후면 카메라 (1)
6      SMART: 카메라 센서 = 얼굴 감지 모드 (1)
7
8      무조건 반복
9      {
10         얼굴감지위치 = SMART: 얼굴 감지 영역
11
12         // 화면에 표시한 빨간색 원을 모두 지움
13         SMART: 도형 표시 = 0
14         // 얼굴 감지 위치에 빨간색 원을 표시
15         SMART: 도형 표시 = [위치:얼굴감지위치],[아이템:1],[크기:10],[색상:빨간색]
16     }
17 }
```

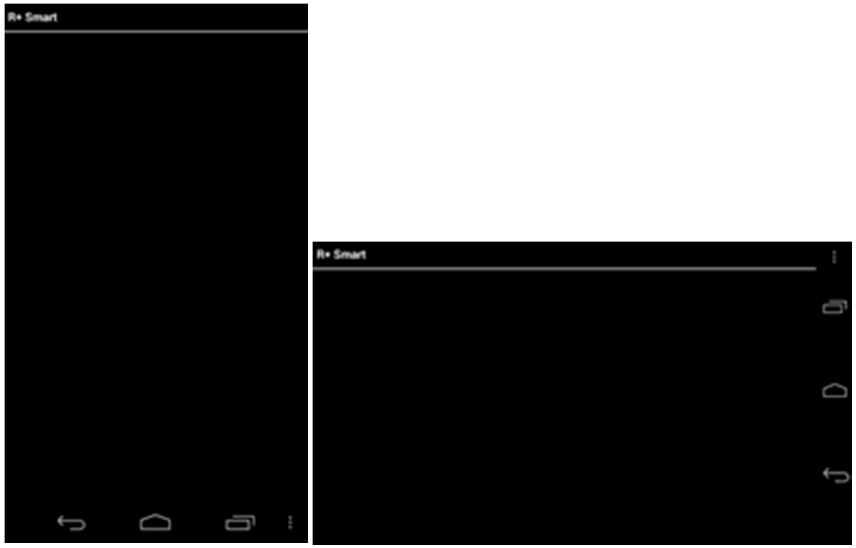


- **Motion detection area** : If you are using a "camera sensor," "motion detection mode" on and used to determine the location of the detected motion.
- Below is an example of displaying a red circle at the position when motion is detected.

4 Display 1 [Screen Rotation / Background Display / Picture Display / Display Detected Face Picture]

It is used to display background, picture, graphic, and text on the screen of smart device.

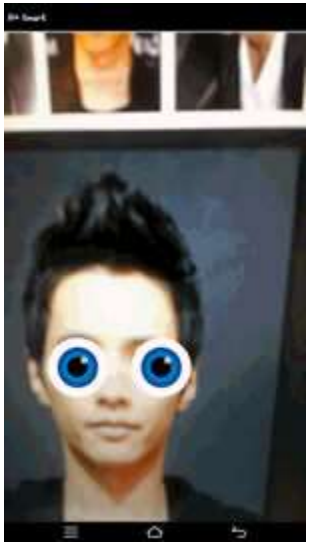
- **Rotate Screen** : The screen used to set the direction of smart devices.
- Below is an example of changing the screen orientation of the smart device every 1.024 seconds alternately.



- **Display Background** : This is used to set a background picture on the screen of your smart device.
  (You can only use the background that you have already registered with R + Smart app.)
- Below is an example of setting the picture background of smart device as item 1.

- **Picture Display** : Used to place a figure on the screen of your smart device.
  (You can only use images that you have already registered in R + Smart app.)
- Below is an example of displaying pictures at positions 2, 3 and 4,3 in the smart device.

- **The Face Detection Picture Display** : Use this if you use the "Face Detection Mode" in the "camera sensor," When you set the picture override the detected face.
  (You can only use images that you have already registered in R + Smart app.)
- Below is an example of detecting the face using the camera of the smart device and then displaying the figure on the detected face.



5 Display 2 [Shape display / Character display / Numerical display]

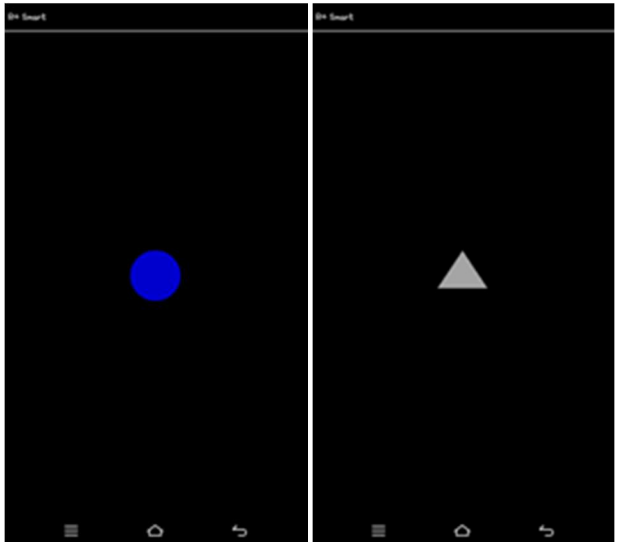
It is used to display background, picture, graphic, and text on the screen of smart device.

.

- **Graphics display** : Used to place the shapes on the screen of your smart device.

  (1: circle, 2: square, 3: triangle)

- Below is an example of alternating blue circles and gray triangles at position 3,3.

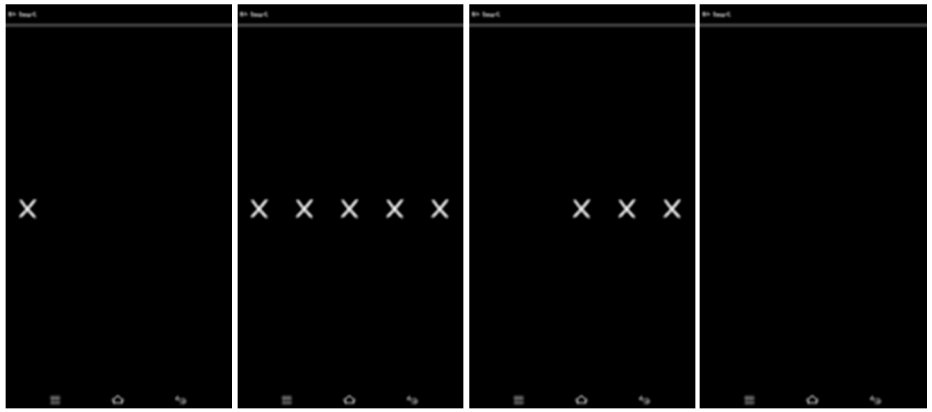


- **Character display** : Used to place the text on the screen of your smart device.

  (You can only use pre-registered characters in R + Smart app.)

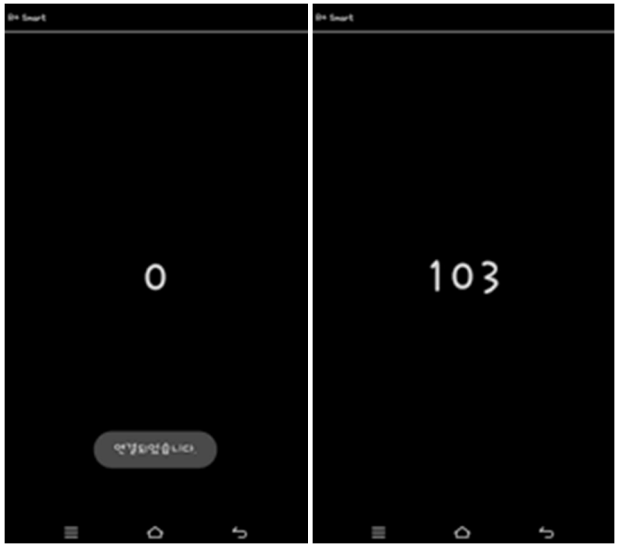
- Below is an example of displaying and deleting characters in positions 1, 3 ~ 5, 3 in order.

```
프로그램 시작
{
    조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )

    무조건 반복
    {
        횟수 반복 ( 위치 = 11 - 15 )
        {
            SMART: 문자 표시 = [위치:위치],[아이템:1],[크기:120],[색상:흰색]
            타이머 = 1.024초
            조건 대기 ( 타이머 > 0 )
        }

        횟수 반복 ( 위치 = 11 - 15 )
        {
            SMART: 문자 표시 = [위치:위치],[아이템:0],[크기:120],[색상:흰색]
            타이머 = 1.024초
            조건 대기 ( 타이머 > 0 )
        }
    }
}
```



- **Numeric display** : Used to place a number on the screen of your smart device.
  (A number between 0 and 255 can be used without registering a separate character.)
- Below is an example of increasing the number at position 3,3.

```
프로그램 시작
{
  조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )

  무조건 반복
  {
    // 0 ~ 255 까지 숫자 표시
    횟수 반복 ( 숫자 = 0 ~ 255 )
    {
      SMART: 숫자 표시 = [위치:(3,3)],[아이템:숫자],[크기:120],[색상:흰색]
      타이머 = 1.024초
      조건 대기 ( 타이머 > 0 )
    }
  }
}
```



6 Multimedia [Automatic TEXT (TTS) / Instrument Play / Audio / Volume / Movie Playback]

Use it to output images or output sounds using the screen and speakers of the smart device.

- **Text-to-speech conversion automatically (TTS)** : is used to take advantage of the services of smart devices automatically convert text to speech.

(You can only use pre-registered characters in R + Smart app.)

- Below is an example that converts character item 2 and character item 3 alternately to voice.

- **Playing a musical instrument** : It is used to make the instruments sound as smart devices.

- Below is an example of repeatedly outputting the notes, even on an acoustic piano.

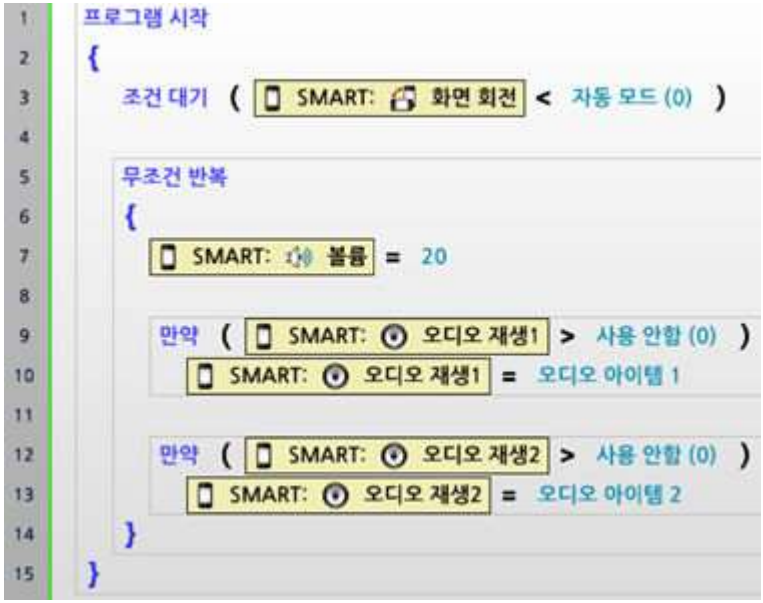
- Audio playback : This is used to play audio files from smart devices. Audio playback 1 and audio playback 2 operate independently.

(Only audio files pre-registered with the R + Smart app are available.)

- **Volume** : used to set the sound volume of smart devices.

The value ranges from 0 to 255. The larger the value, the larger the volume. The range of values may vary depending on the device.

- Below is an example of playing the sound source of the smart device using audio playback 1, audio playback 2, volume.



- **Video playback** : Use when playing a video file of smart devices.

(You can only use videos that have already been registered with the R + Smart app.)

- **Video Pause** : Used to pause when a video file is being played from a smart device.

- Below is an example of pausing a video while you are touching the screen using video playback and video pause.

7 Sensor 1 [Shake sensor / tilt sensor / illuminance sensor]

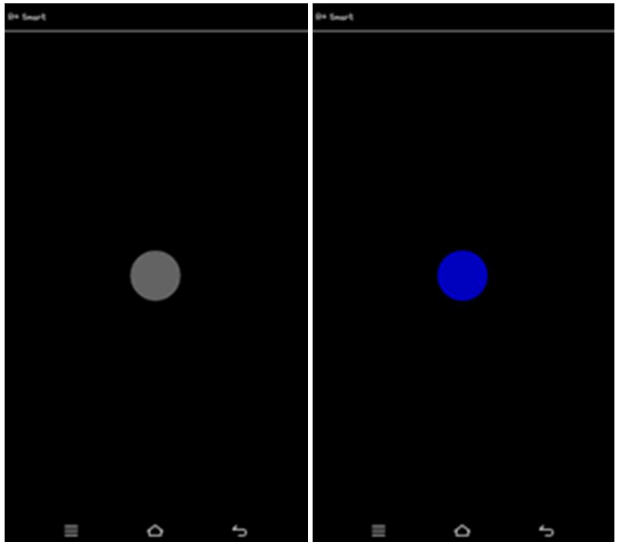
It is used to take advantage of various sensors built into the smart device.

- Shake sensor : Use a vibration sensor to take advantage of smart devices.

The value of 0 ~ 255 is output according to the shaking of smart device.

- Below is an example of changing the color of the figure displayed on the screen when the value of 80 is over by reading the shake of the smart device.

```
프로그램 시작
{
    조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )

    무조건 반복
    {
        흔들림 = SMART: 흔들림 센서

        만약 ( 흔들림 >= 80 )
        {
            2 임의의 숫자 = 9
            임의의색상 = 2 임의의 숫자

            SMART: 도형 표시 = [위치:(3,3)],[아이템:1],[크기:60],[색상:임의의색상]
        }
    }
}
```



- Tilt sensor : used to take advantage of the tilt sensor of smart devices.

(Left), (Right), (Up), (Down) , each 0-90 output is the slope of the road.

- Below is an example of displaying a circle in a tilted direction on the screen according to the tilt of the smart device.

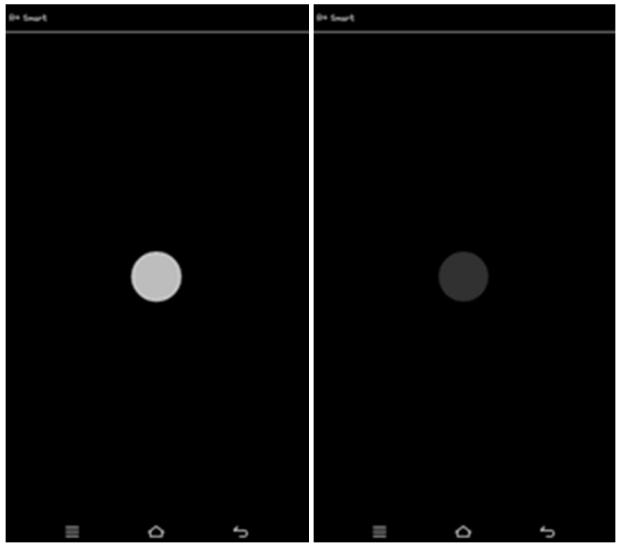
```
1  프로그램 시작
2  {
3      조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )
4      SMART: 화면 회전 = 가로 모드 (2)
5      무조건 반복
6      {
7          기울기각도왼쪽 = SMART: 기울기 센서 (왼쪽)
8          기울기각도오른쪽 = SMART: 기울기 센서 (오른쪽)
9          기울기각도위쪽 = SMART: 기울기 센서 (위쪽)
10         기울기각도아래쪽 = SMART: 기울기 센서 (아래쪽)
11         만약 ( 기울기각도왼쪽 > 30 )
12             SMART: 도형 표시 = [위치:(5,3)],[아이템:1],[크기:60],[색상:흰색]
13         아니면
14             SMART: 도형 표시 = [위치:(5,3)],[아이템:0],[크기:60],[색상:흰색]
15         만약 ( 기울기각도오른쪽 > 30 )
16             SMART: 도형 표시 = [위치:(1,3)],[아이템:1],[크기:60],[색상:흰색]
17         아니면
18             SMART: 도형 표시 = [위치:(1,3)],[아이템:0],[크기:60],[색상:흰색]
19         만약 ( 기울기각도위쪽 > 30 )
20             SMART: 도형 표시 = [위치:(3,5)],[아이템:1],[크기:60],[색상:흰색]
21         아니면
22             SMART: 도형 표시 = [위치:(3,5)],[아이템:0],[크기:60],[색상:흰색]
23         만약 ( 기울기각도아래쪽 > 30 )
24             SMART: 도형 표시 = [위치:(3,1)],[아이템:1],[크기:60],[색상:흰색]
25         아니면
26             SMART: 도형 표시 = [위치:(3,1)],[아이템:0],[크기:60],[색상:흰색]
27     }
28 }
```

- **Light sensor** : use the light sensor to take advantage of smart devices.

  0 ~ 65535 is output according to the ambient brightness. The range of values may vary depending on the device.

- Below is an example of measuring the illuminance and displaying a gray circle when the surroundings are dark and a white circle when the surroundings are bright.

```
프로그램 시작
{
    조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )

    무조건 반복
    {
        조도 = SMART: 조도 센서

        만약 ( 조도 == 0 )
        {
            SMART: 도형 표시 = [위치:(3,3)],[아이템:1],[크기:60],[색상:짙은 회색]
        }
        아니면
        {
            SMART: 도형 표시 = [위치:(3,3)],[아이템:1],[크기:60],[색상:흰색]
        }
    }
}
```



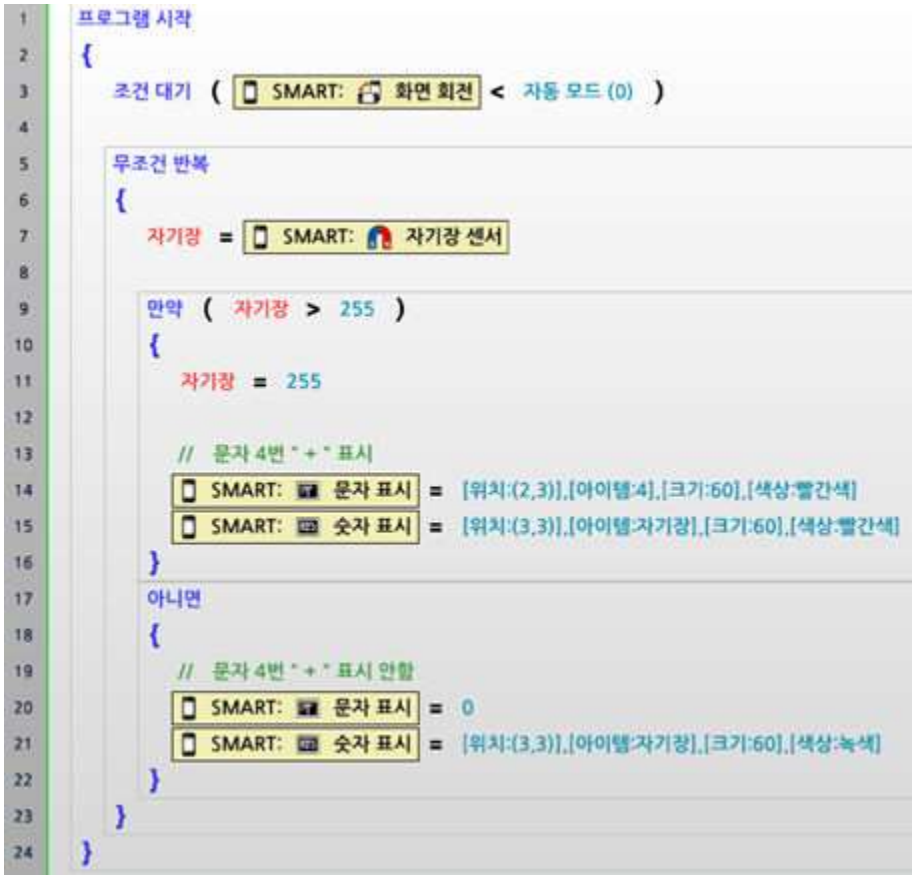
8 Sensor 2 [magnetic field sensor / direction sensor / noise sensor]

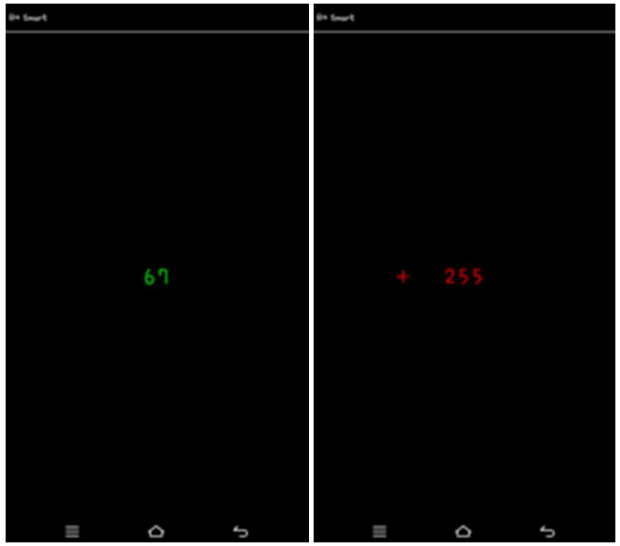
It is used to take advantage of various sensors built into the smart device.

- **Magnetic field sensor** : Use the magnetic field sensor to take advantage of smart devices.

  0 ~ 65535 is output according to the surrounding magnetic field.

- Below is an example of measuring the magnetic field around the smart device and displaying the value on the screen.

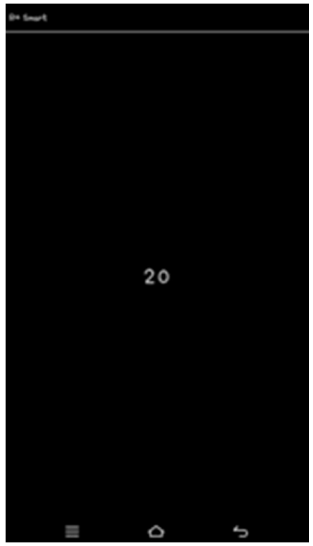




- Orientation Sensor: Use the orientation sensor to take advantage of smart devices.

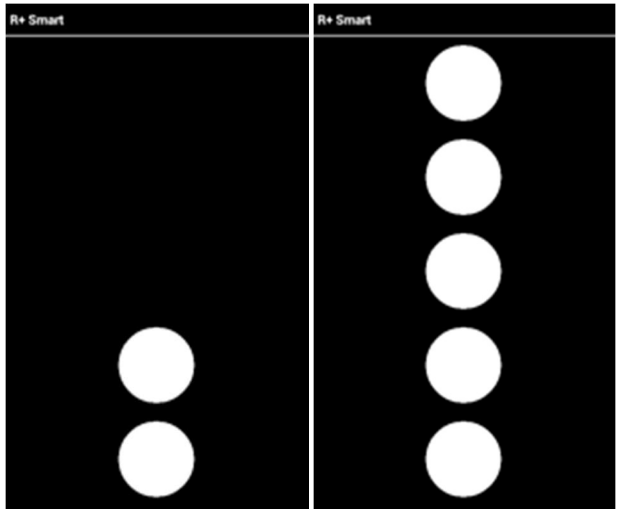
Outputs a value between 0 and 359 in degrees according to the direction. (0: North, 90: East, 180: South, 270: West)

- Below is an example of dividing the direction value of the smart device by 10 and displaying it on the screen.



- **Noise Sensor:** Use this to take advantage of the noise of the smart sensor device.
  Outputs a value between 0 and 255 in dB depending on the noise.
- Below is an example of displaying shapes according to the size of the noise.

```
프로그램 시작
{
    조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )
    무조건 반복
    {
        소음 = SMART: 소음 센서 (dB)
        만약 ( 소음 > 80 )
            SMART: 도형 표시 = [위치:(3,1)],[아이템:1],[크기:60],[색상:흰색]
        아니면
            SMART: 도형 표시 = [위치:(3,1)],[아이템:0],[크기:60],[색상:흰색]
        만약 ( 소음 > 70 )
            SMART: 도형 표시 = [위치:(3,2)],[아이템:1],[크기:60],[색상:흰색]
        아니면
            SMART: 도형 표시 = [위치:(3,2)],[아이템:0],[크기:60],[색상:흰색]
        만약 ( 소음 > 60 )
            SMART: 도형 표시 = [위치:(3,3)],[아이템:1],[크기:60],[색상:흰색]
        아니면
            SMART: 도형 표시 = [위치:(3,3)],[아이템:0],[크기:60],[색상:흰색]
        만약 ( 소음 > 50 )
            SMART: 도형 표시 = [위치:(3,4)],[아이템:1],[크기:60],[색상:흰색]
        아니면
            SMART: 도형 표시 = [위치:(3,4)],[아이템:0],[크기:60],[색상:흰색]
        만약 ( 소음 > 40 )
            SMART: 도형 표시 = [위치:(3,5)],[아이템:1],[크기:60],[색상:흰색]
        아니면
            SMART: 도형 표시 = [위치:(3,5)],[아이템:0],[크기:60],[색상:흰색]
    }
}
```

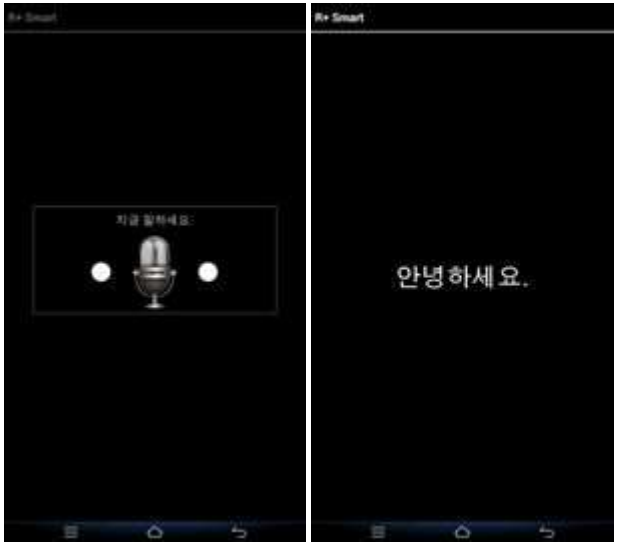


9 Sensor 3 [Speech Recognition / Touch Position / Gesture Recognition]

It is used to take advantage of various sensors built into the smart device.

- **Speech Recognition:** Use to turn the speech recognition functionality of smart devices on or off.
  When "True", "Speech Recognition starts" and when it is False, it means "Speech Recognition Stop".
- **Voice recognition results** : When using the "Speech Recognition" feature, used to check the recognition results.
  The recognized result is displayed as a number. "No result" when it is 0, 1 ~ 199 matches the corresponding character item.
- Below is an example of recognizing the voice when the screen is touched by using voice recognition and voice recognition result and displaying the recognized result in the center of the screen.

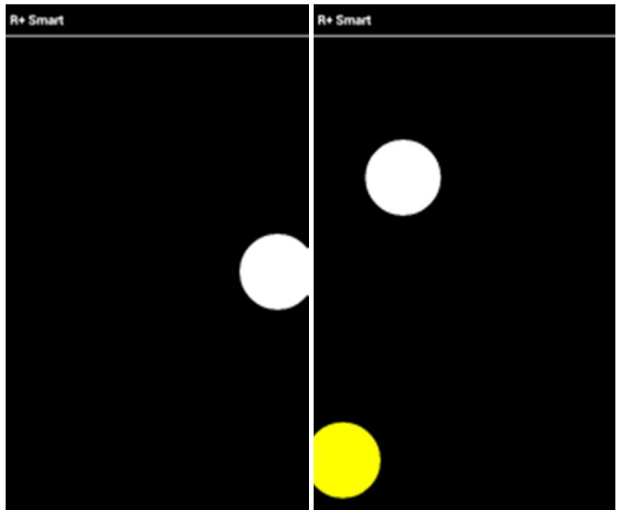

프로그램 시작
{
조건 대기 (SMART: 화면 회전 < 자동 모드 (0))
무조건 반복
{
터치 = SMART: 터치 위치1
만약 (터치 > 0)
{
만약 (SMART: 음성 인식 == 음성 인식 정지 (0))
SMART: 음성 인식 = 음성 인식 시작 (1)
}
음성인식결과 = SMART: 음성 인식 결과
만약 (음성인식결과 > 0)
{
SMART: 문자 표시 = [위치:(3,3)],[아이템:음성인식결과],[크기:60],[색상:흰색]
타이머 = 4.992초
조건 대기 (타이머 > 0)
SMART: 음성 인식 결과 = 사용 완함 (0)
음성인식결과 = 0
}
}
}



- Touch location : Use the touch screen to take advantage of the location of the smart machine. **Touch position 1** refers to the finger touch and the first **touch position 2** stands for a finger touch in the second.

- Below is an example of displaying the shape in the place you touched.

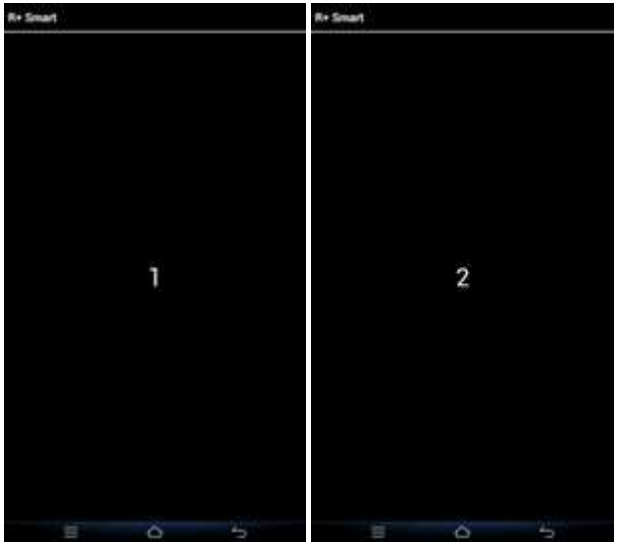
```
프로그램 시작
{
    조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )

    무조건 반복
    {
        터치1 = SMART: 터치 위치1
        터치2 = SMART: 터치 위치2

        // 모든 도형 지우기
        SMART: 도형 표시 = 0

        만약 ( 터치1 > 0 )
            SMART: 도형 표시 = [위치:터치1],[아이템:1],[크기:60],[색상:흰색]

        만약 ( 터치2 > 0 )
            SMART: 도형 표시 = [위치:터치2],[아이템:1],[크기:60],[색상:노란색]
    }
}
```



- Gesture Recognition : use to take advantage of the gesture recognition capabilities of smart devices.

- Below is an example of recognizing a gesture and outputting the number of the gesture on the screen.


프로그램 시작
{
조건 대기 (SMART: 화면 회전 < 자동 모드 (0))
무조건 반복
{
제스처 = SMART: 제스처 인식
만약 (제스처 == 1)
{
SMART: 숫자 표시 = [위치:(3,3)],[아이템:1],[크기:60],[색상:흰색]
타이머 = 5.120초
조건 대기 (타이머 > 0.000초)
SMART: 제스처 인식 = 0
SMART: 숫자 표시 = 0
}
아니면 만약 (제스처 == 2)
{
SMART: 숫자 표시 = [위치:(3,3)],[아이템:2],[크기:60],[색상:흰색]
타이머 = 5.120초
조건 대기 (타이머 > 0.000초)
SMART: 제스처 인식 = 0
SMART: 숫자 표시 = 0
}
}
}



10 Others 1 [Debug information display / Screen output / New line after screen output]

Used to take advantage of the additional features of the smart device.

- Debug information display: Used to display the value of major functions of the smart device on the screen.

- When the lower bit (right) is set to 1, the following information can be displayed on the screen of the smart device.
- 1st bit: Vision related position, color display (at constant input: 1)
- 2nd bit: Shaking value display (at constant input: 2)
- 3rd bit: Display the slope up / down / left / right value (at constant input: 4)
- 4th bit: Roughness value display (When inputting constant: 8)
- 5th bit: Displays the magnetic field value (at constant input: 16)
- 6th bit: direction value display (when inputting integer: 32)
- 7th bit: Noise value display (at constant input: 64)
- 8th bit: Touch position 1, 2 value display (for integer input: 128)
- 9th bit: Voice input result value display (when inputting constant: 256)
- 10th bit: SMS related phone number, contents (when inputting integer: 512)

- Below is an example using debug information display function.

- Output : Use when you want to check for a specific value in the task code with snow.
(Appears on the smart app screen.)
Screen output after line break : This is used when you want to check a particular value in the task code with snow. After output, it automatically changes to the next line. (Appears on the smart app screen.)

- Below is an example of screen output of smart device.

11 Others 2 [Smart Timer / Vibration / Current Time / Flash LED / Execute App]

Used to take advantage of the additional features of the smart device.

- Smart Timer: Use to set the timer for smart devices.

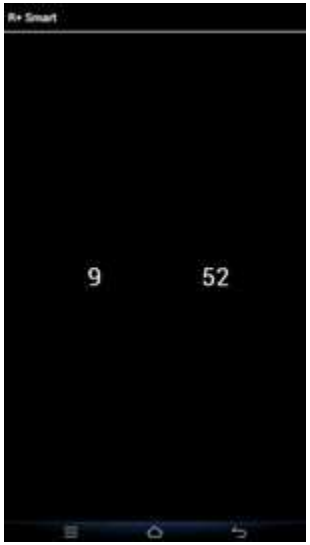
- Vibration time: Used to turn the vibration function of smart devices.

- Vibration condition: used smart devices to verify that the current vibration.

- Below is an example that uses a smart timer and vibration time to vibrate for 1 second every 10 seconds.

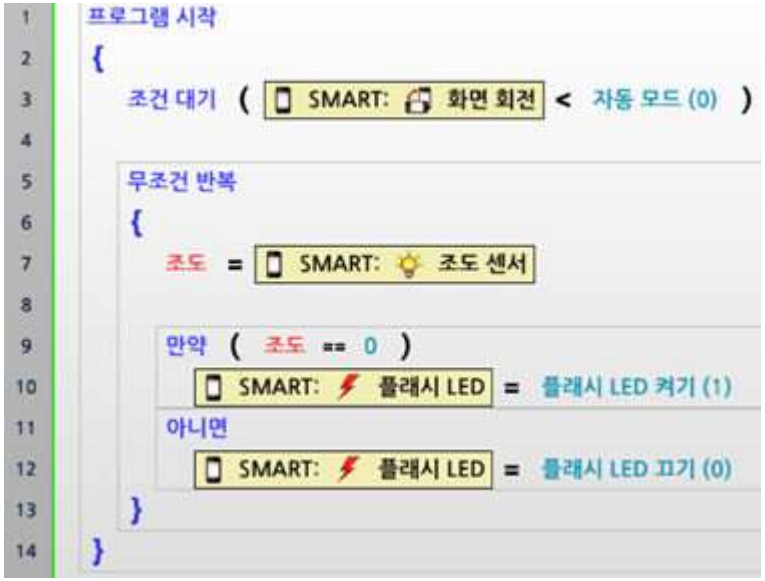
- Current Time: You get used to read the current time from the smart devices.

- Below is an example of displaying the current time on the screen.



- Flash LED: Used to turn on the camera flash LED on the smart device on or off.

- Below is an example of measuring the ambient light with the illuminance sensor and turning on the flash LED when it is dark.



- To run the app: The app is used to run the registration of smart devices automatically.

- Below is an example of running a registered app by touching the screen.

12 Others 3 [Send SMS / Notification Bar Event / E-Mail]

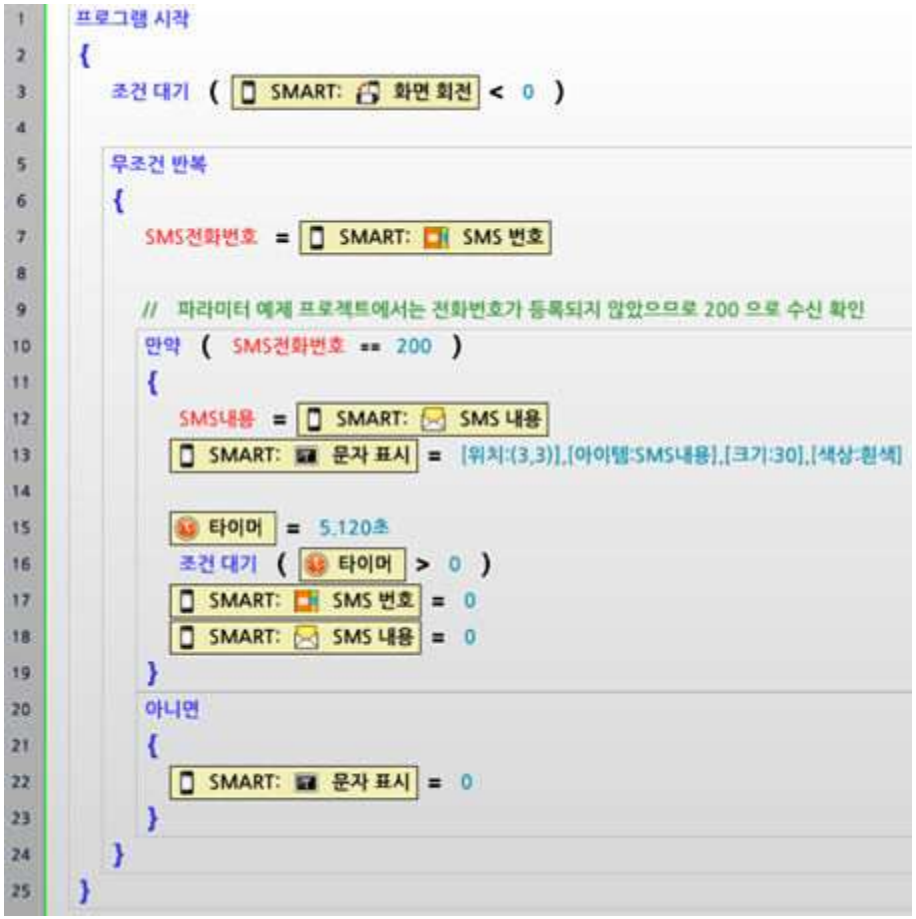
Used to take advantage of the additional features of the smart device.

.

- SMS number used to identify the number receiving SMS when you receive an SMS.

- SMS content: It is used to check the information received SMS When you receive an SMS.

- Below is an example to check the SMS number and display SMS contents on the screen.


프로그램 시작
{
조건 대기 (SMART: 화면 회전 < 0)
무조건 반복
{
SMS전화번호 = SMART: SMS 번호
// 파라미터 예제 프로젝트에서는 전화번호가 등록되지 않았으므로 200 으로 수신 확인
만약 (SMS전화번호 == 200)
{
SMS내용 = SMART: SMS 내용
SMART: 문자 표시 = [위치:(3,3)],[아이템:SMS내용],[크기:30],[색상:흰색]
타이머 = 5,120초
조건 대기 (타이머 > 0)
SMART: SMS 번호 = 0
SMART: SMS 내용 = 0
}
아니면
{
SMART: 문자 표시 = 0
}
}
}

- Notification Bar Event information: When you receive a message through the instant messaging such as KakaoTalk, Line, used to check the information received.

- Below is an example of displaying the message contents of a normal messenger on the screen.

```
1  프로그램 시작
2  {
3      조건 대기 ( SMART: 화면 회전 < 자동 모드 (0) )
4
5      무조건 반복
6      {
7          알림바이벤트 = SMART: 알림바 이벤트 내용
8
9          만약 ( 알림바이벤트 == 200 )
10         {
11             SMART: 문자 표시 = [위치:(3,3)],[아이템:200],[크기:30],[색상:흰색]
12
13             타이머 = 5.120초
14             조건 대기 ( 타이머 > 0 )
15             SMART: 알림바 이벤트 내용 = 0
16         }
17         아니면
18         {
19             SMART: 문자 표시 = 0
20         }
21     }
22 }
```

- **E-Mail Transport:** Used to send your pictures or videos to E-Mail.
- **E-Mail Transfer Status:** Currently used to verify that the E-Mail is sent.
- Below is an example of e-mail sending function and e-mail sending status.

13 Direct Input

- You can access the address of smart device and read and write.
- Read or write the address specified by the user in Byte, Word, or DWord units.
- Please refer to the control table of R + Smart. R + Smart Control Tables

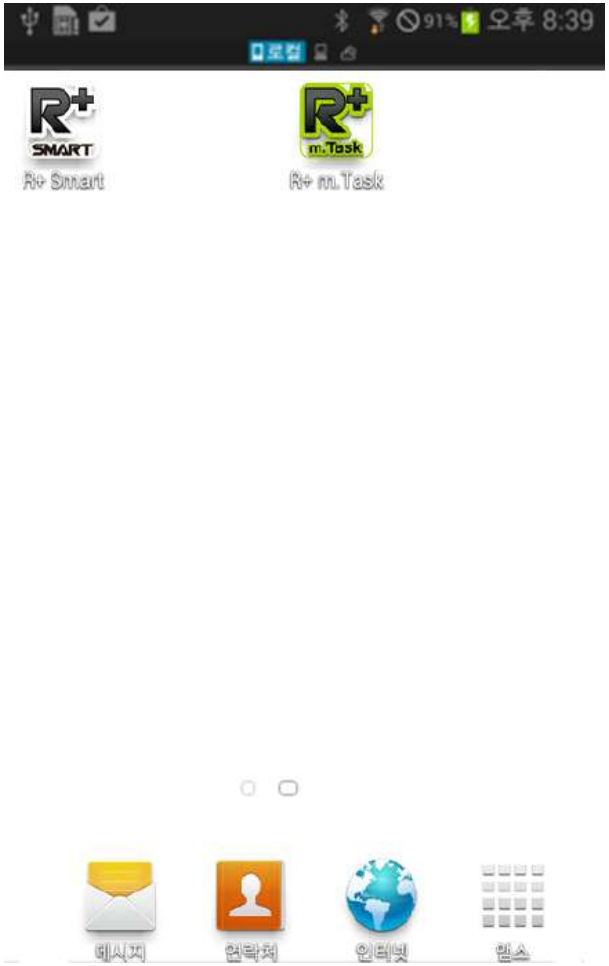
R + SMART Manual

Contents

1. Download and install R + Smart

A. Search and install Robotize or R + Smart on Google Play.

B. Please install R + m.Task as well.



C. After installation, the folder will be created in SD card \ Smart as below.

D. Custom records user projects, and system records example projects.

In E. System, each project has the following structure.

①Audio: Sound file used in project is saved.

②Capture: The photo file captured from Project t is saved.

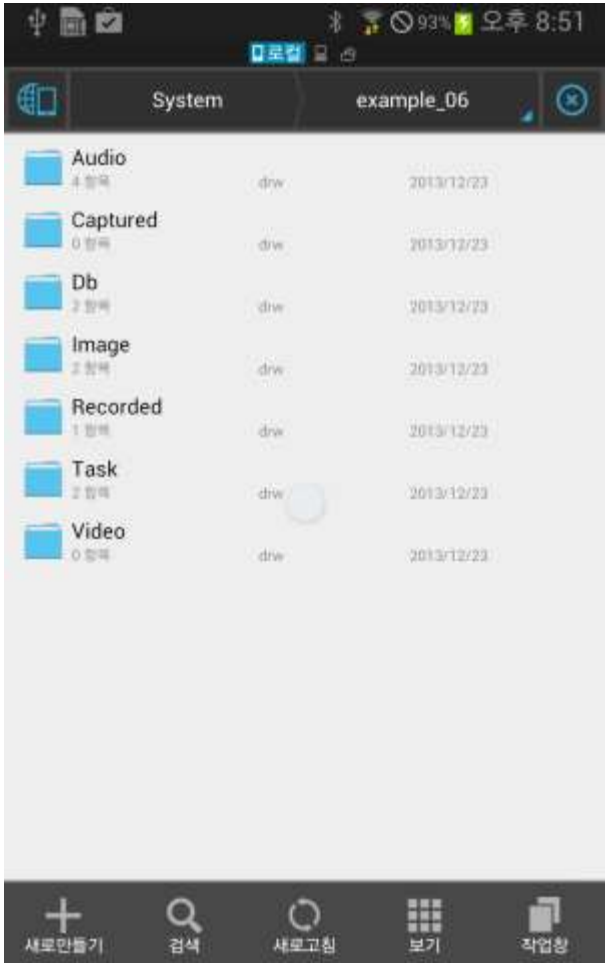
③Db: All file information and jezzy file used in project are saved.

④Image: Image file used in project is saved.

⑤Recorded: The video file recorded in the project is saved.

⑥Task: Project related task file is saved. If you create a task on a PC by connecting the smart device and the PC by copying the files created task in the Task folder of the project it can be used in conjunction with m.Task.

⑤Video: The video file used in the project is saved.



2. How to use R + Smart application

A. Preferences

① Press the setup button on the upper right to return to the configuration screen.

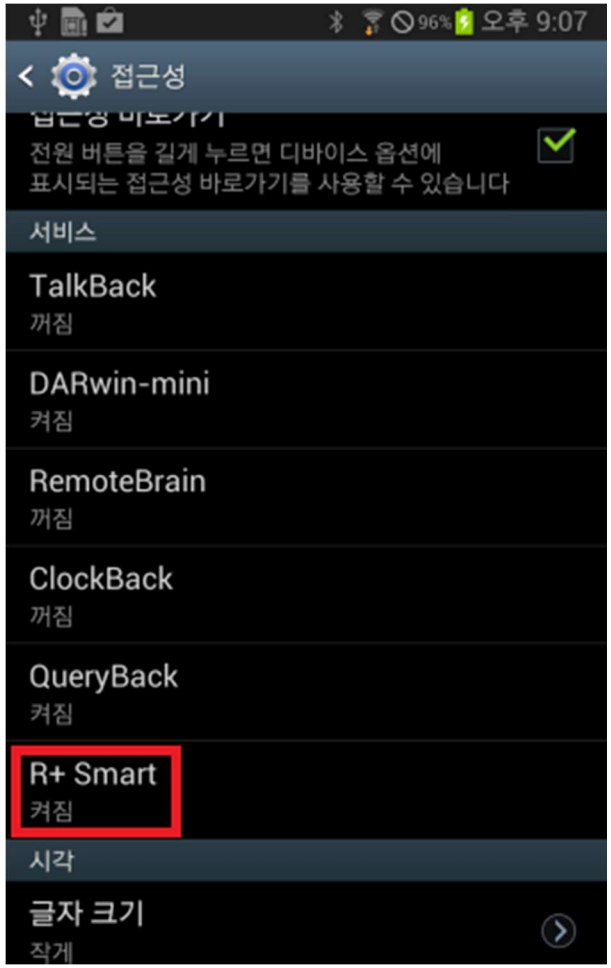
② On the setting screen, you can set the devices to be connected, accessibility service, and gesture error range.

③ Device to connect: If you select the address from Bluetooth module, it tries to connect to the selected device when connecting controller.

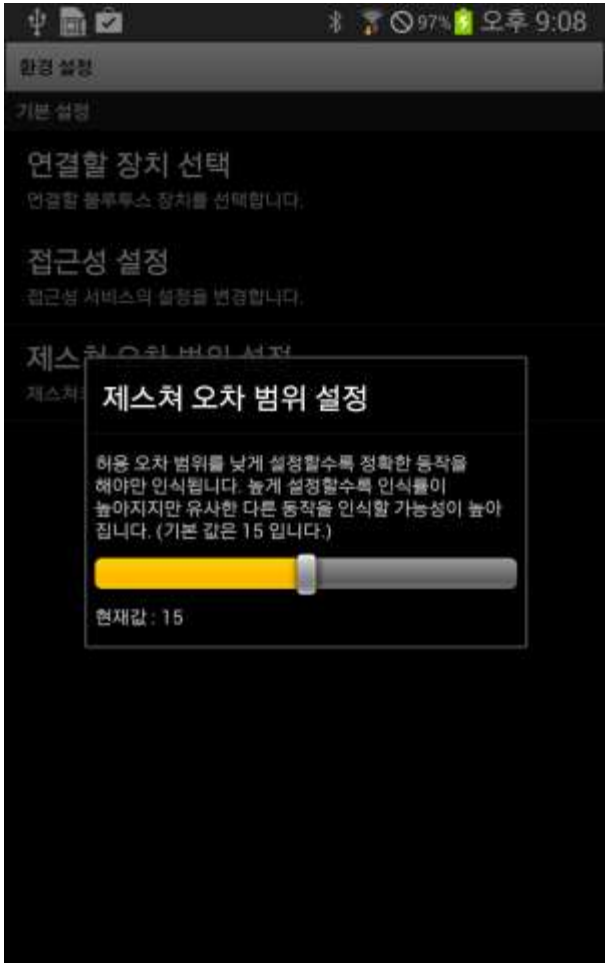
④ Example initialization: Initialize each example. When you accidentally delete an image or task used in the example,

You can return to the original state. After you select an example, all tasks, images, and so on that were modified by the user during initialization are deleted.

⑤ Accessibility setting: The function must be activated for the event function.



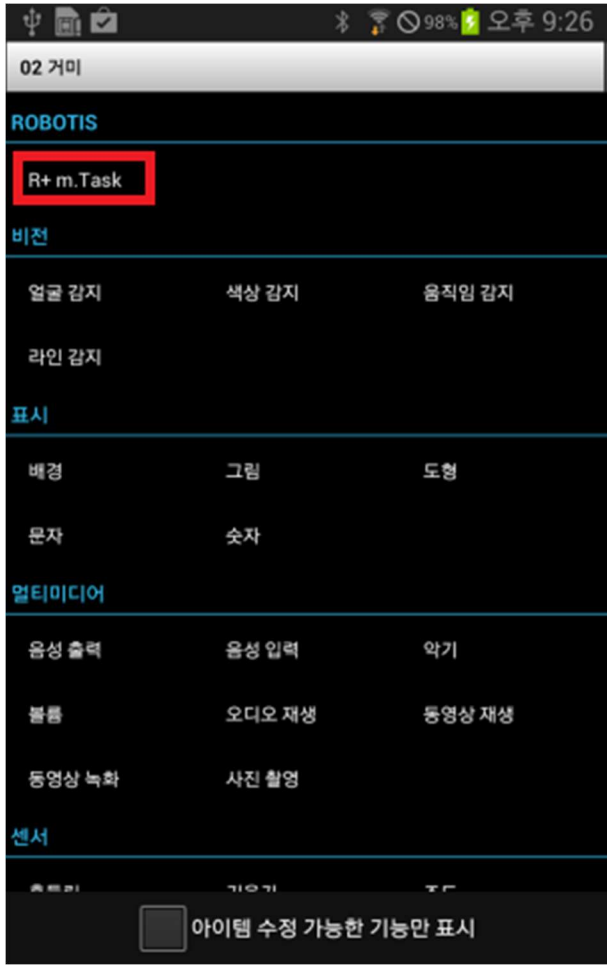
⑥ Gesture error range setting: Set the accuracy of gesture function.



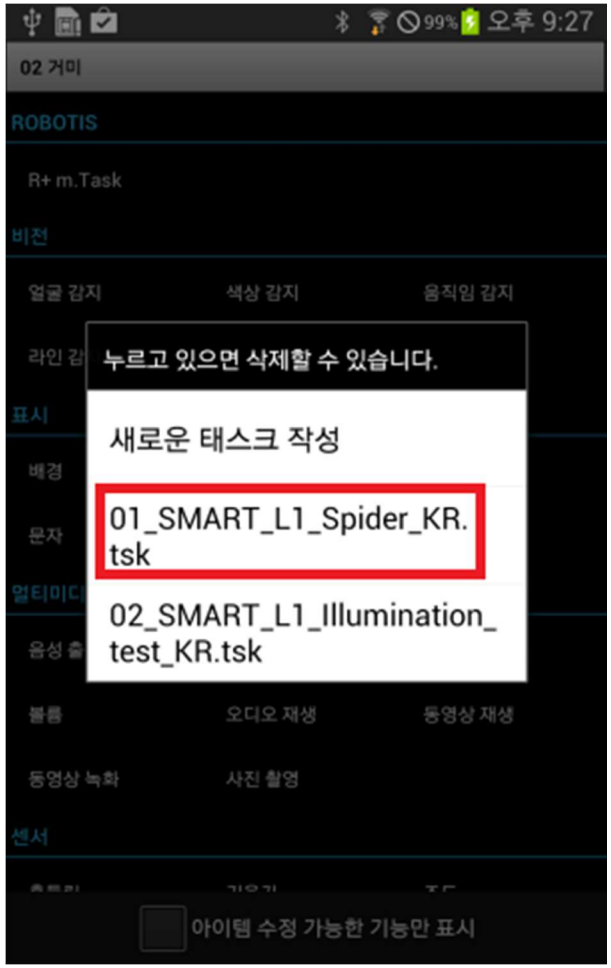
B. R + m.Task interlock

① Press and hold the project to display the function confirmation and editable screen.

② Click R + mTask on the displayed screen.



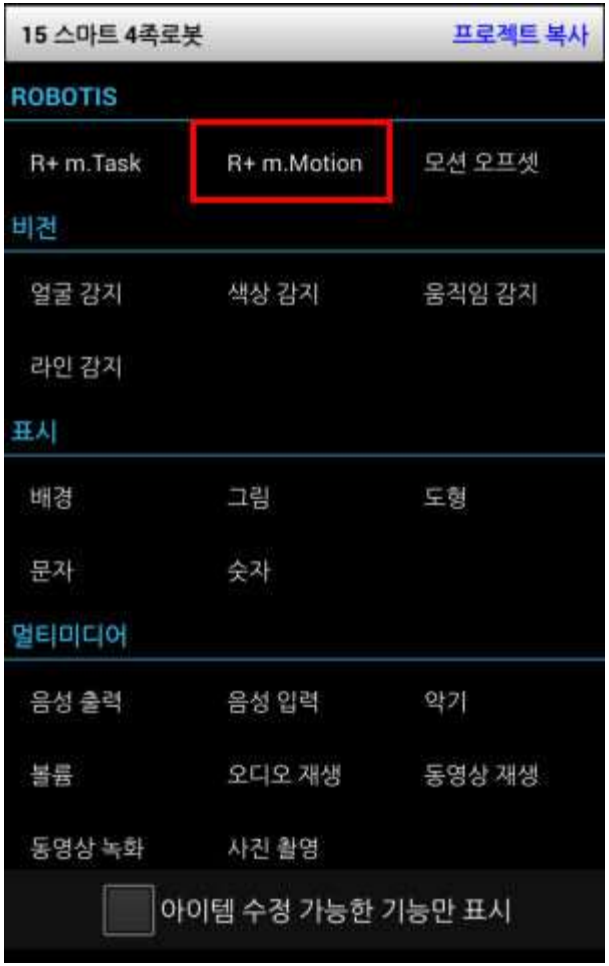
③ When the task is selected, R + m.Task is executed.



④ Click the menu button to download the task to the controller.

C. R + m.Motion interlock

① Press and hold the project and click R + m.Motion on the screen to check and modify the function of the indicator.

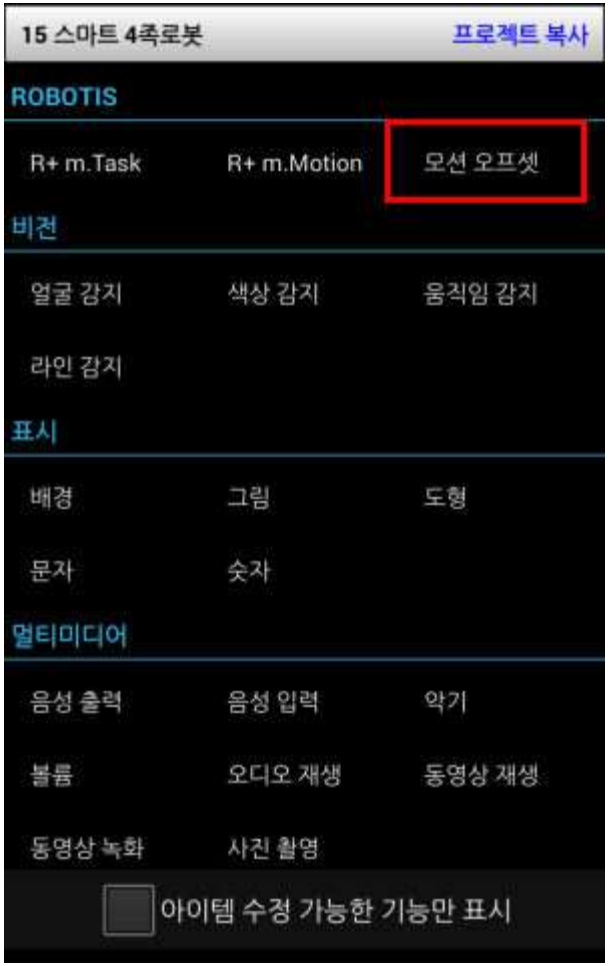


② When motion is selected, R + m.Motion is executed.

Information relating to the ③R + m.Motion Lobo Motion Plus 2.0, please see.

D. Motion Offset Settings

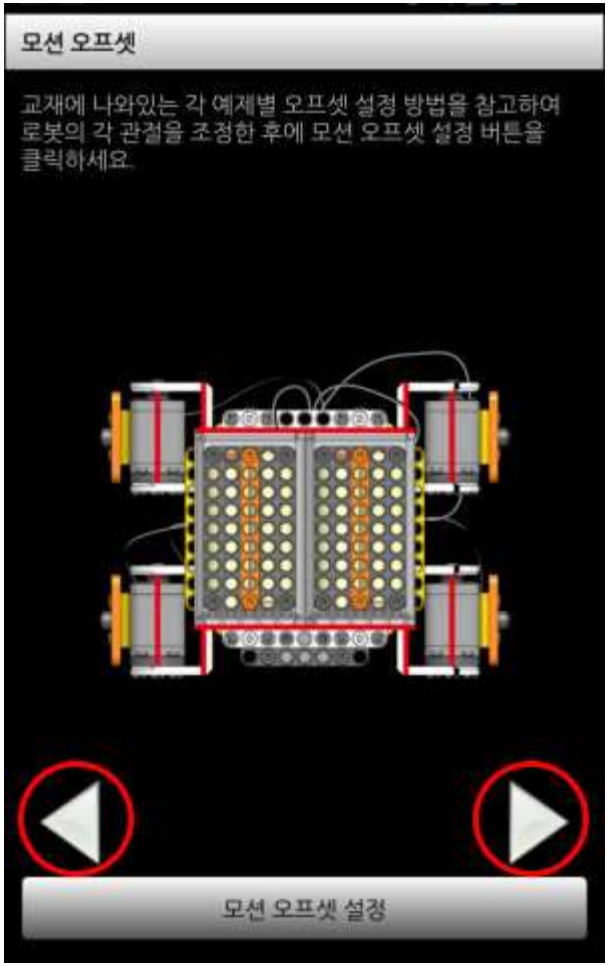
① Press and hold the project, and then click the motion offset on the screen to check and modify the displayed function.



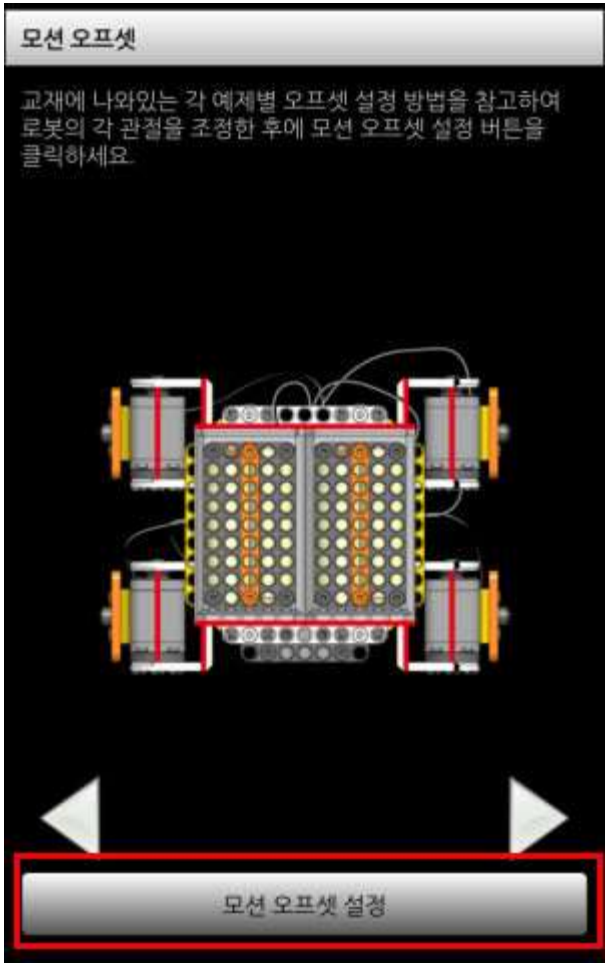
② If you can not connect, click "Retry" or click "Select Device" to select the correct Bluetooth.

③ In the case of using the motion (smart 4-legged robot, smart dog, transformation robot, smart droid)

Click the arrows to see the position of each joint in different directions.



④ Save each joint of the robot according to the offset setting method, and then click the motion offset setting button.



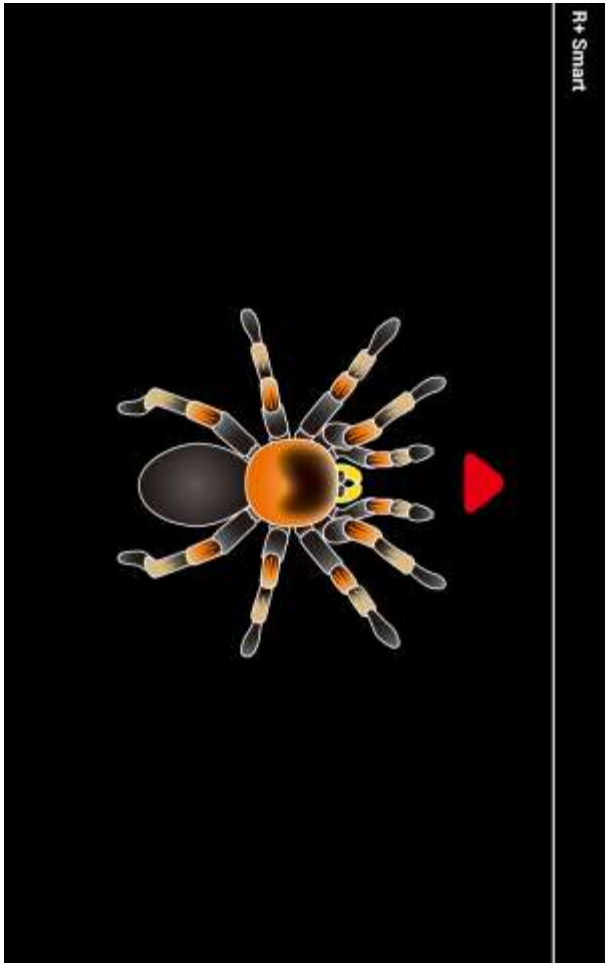
E. Run the project

① Press the start button of the robot, connect the Bluetooth, and then click the project briefly.

② If you can not connect, click "Retry" or click "Select Device" to select the correct Bluetooth.

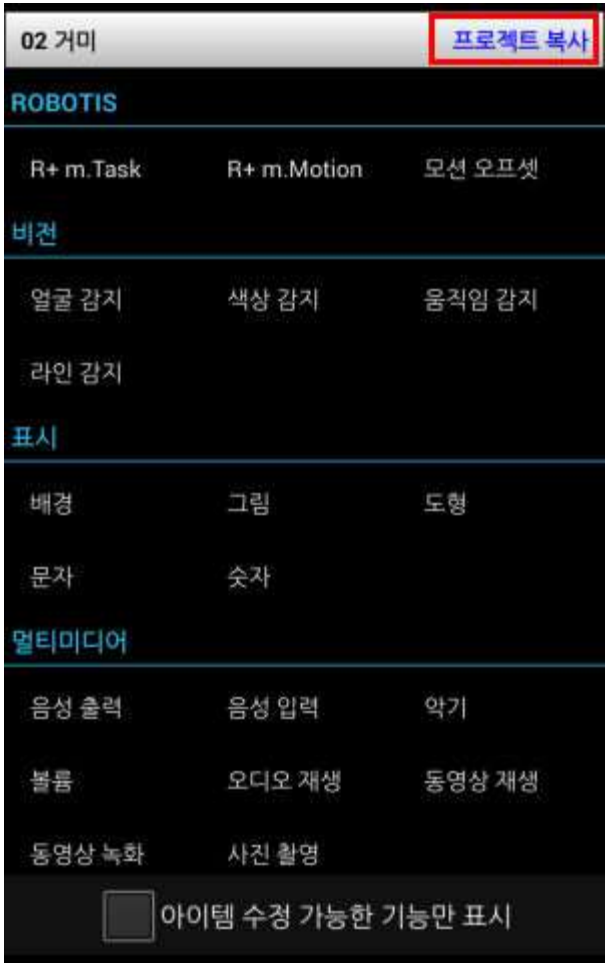
③ When connected, the screen created by the task is displayed.

(If the PIN number input screen appears, enter "0000".)



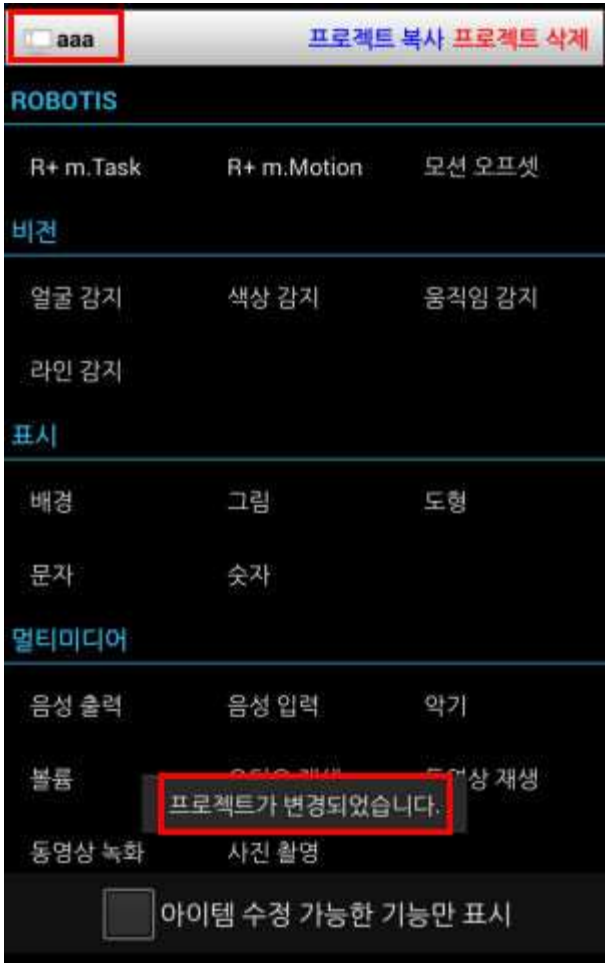
F. Copy Project

① Press and hold the project, and then click Copy Project on the screen that allows you to check and modify the displayed functions.



② Enter the name of the project to be copied and click the OK button.

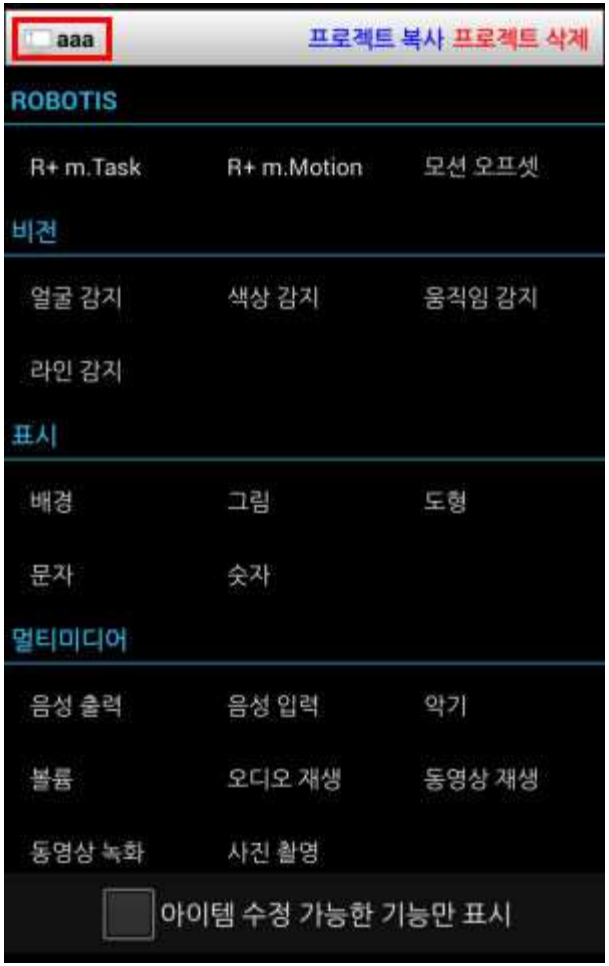
③ When copying is completed, the project name is changed.



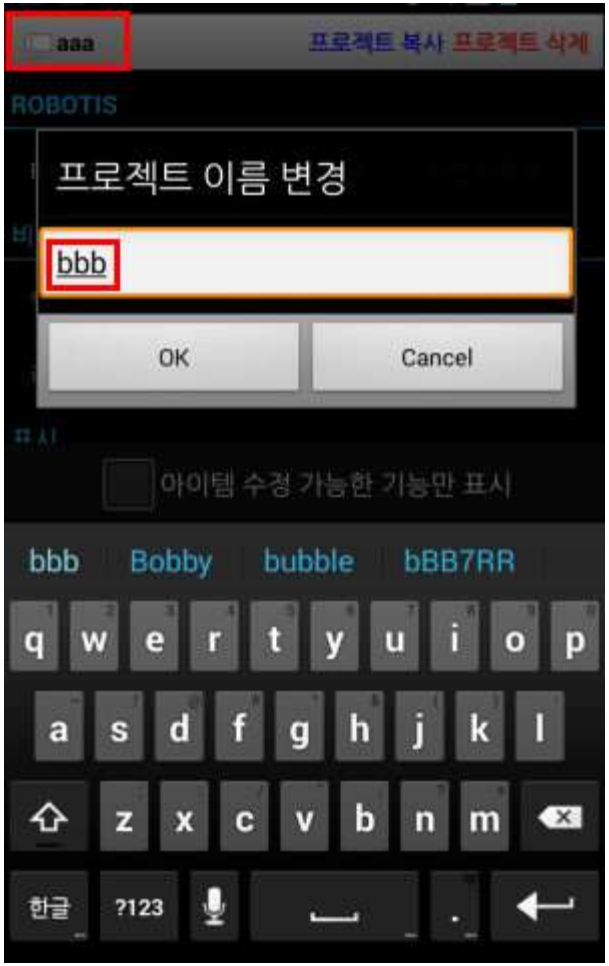
④ The copied project is added as a user project.

G. Rename the project

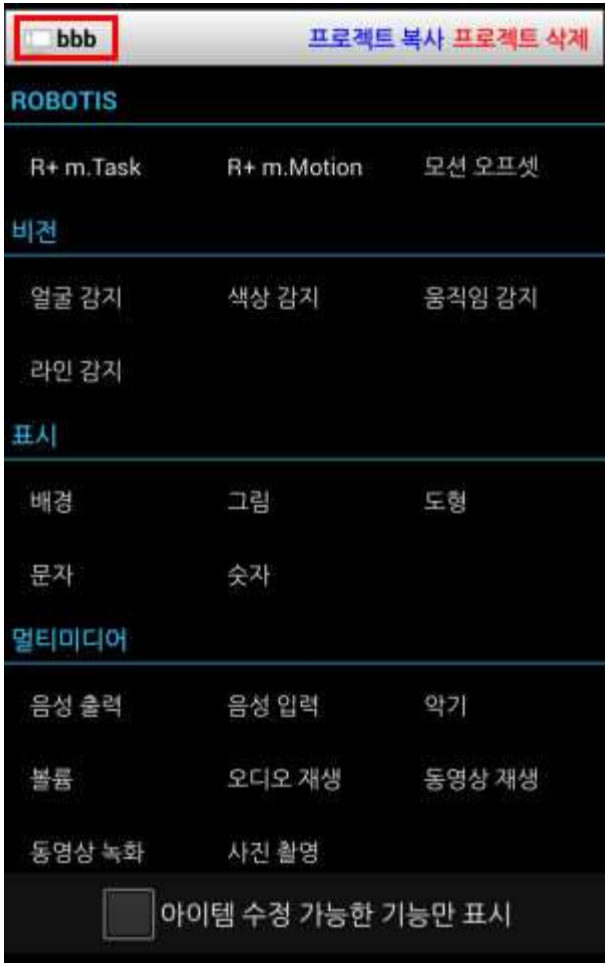
① Press and hold the project, and click the project name on the screen to check and modify the displayed function. (Only user projects can be renamed.)



② Enter the new project name and click the OK button.



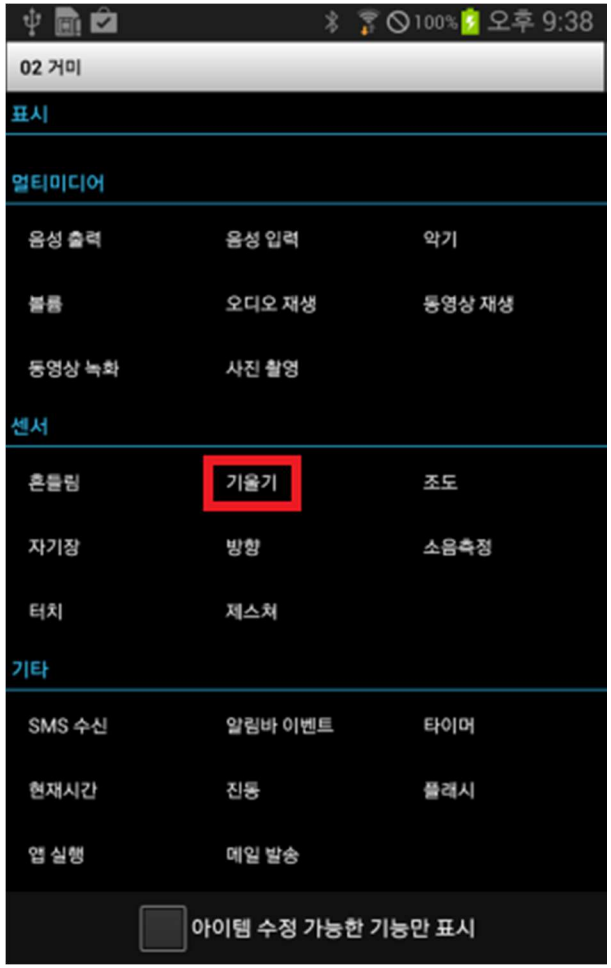
③ The project will be changed to the name you entered.



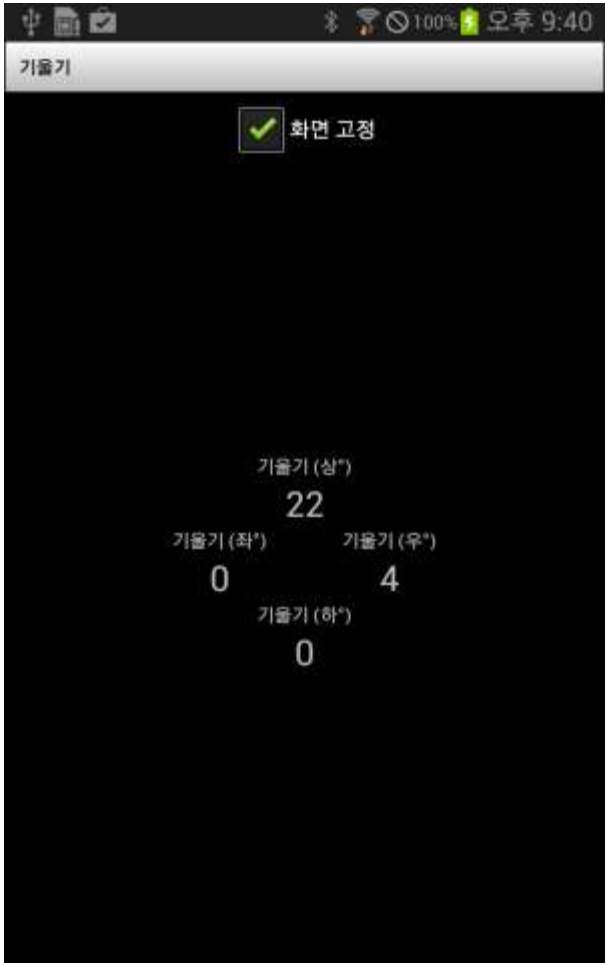
3. Check and modify R + Smart application function

A. Functional verification (eg tilt)

① Click the slope.



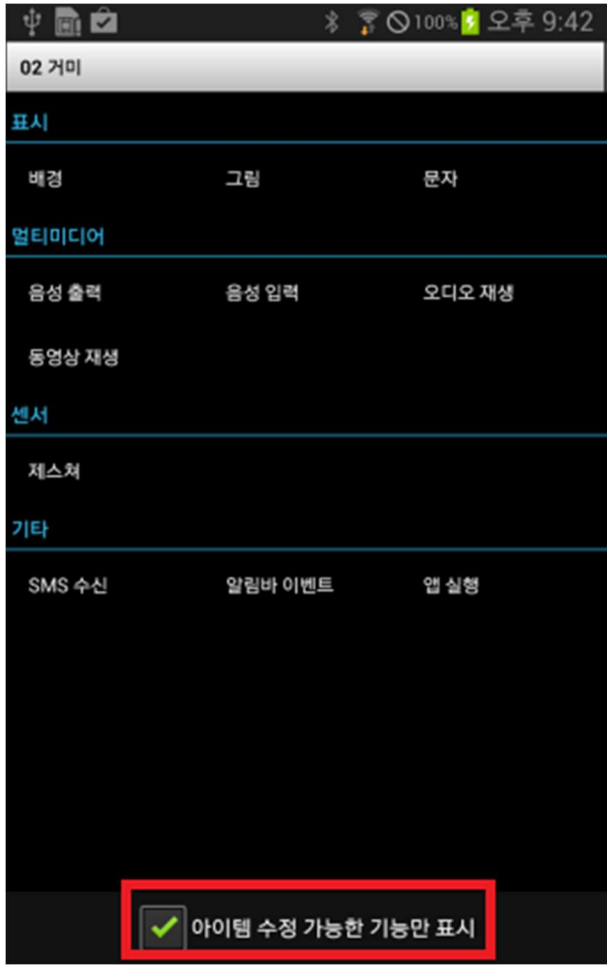
② Check the displayed slope value.



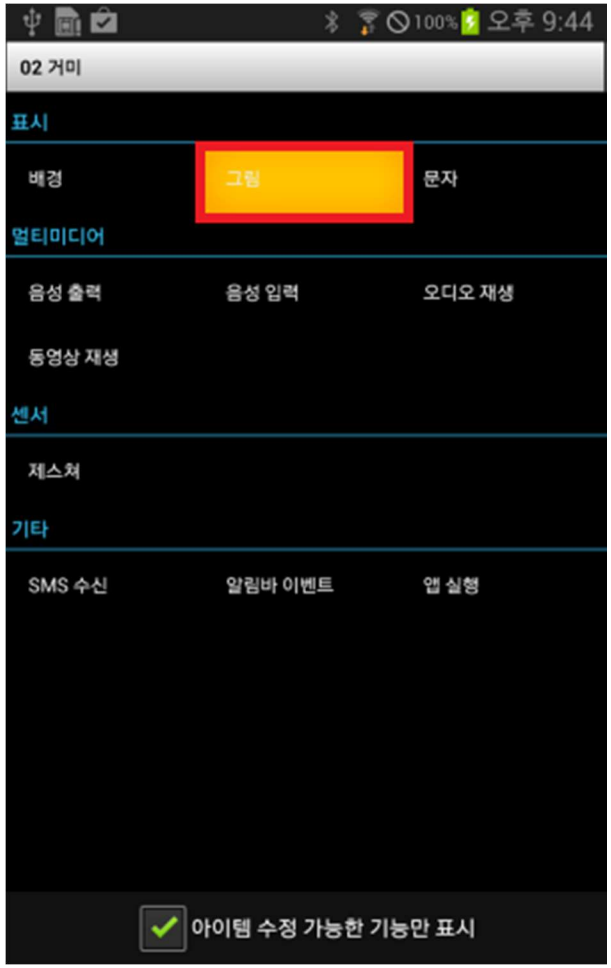
③ Start the task according to the checked value.

B. Modification of function (eg picture)

① Click the check box at the bottom of the screen to display only the functions that can be modified.

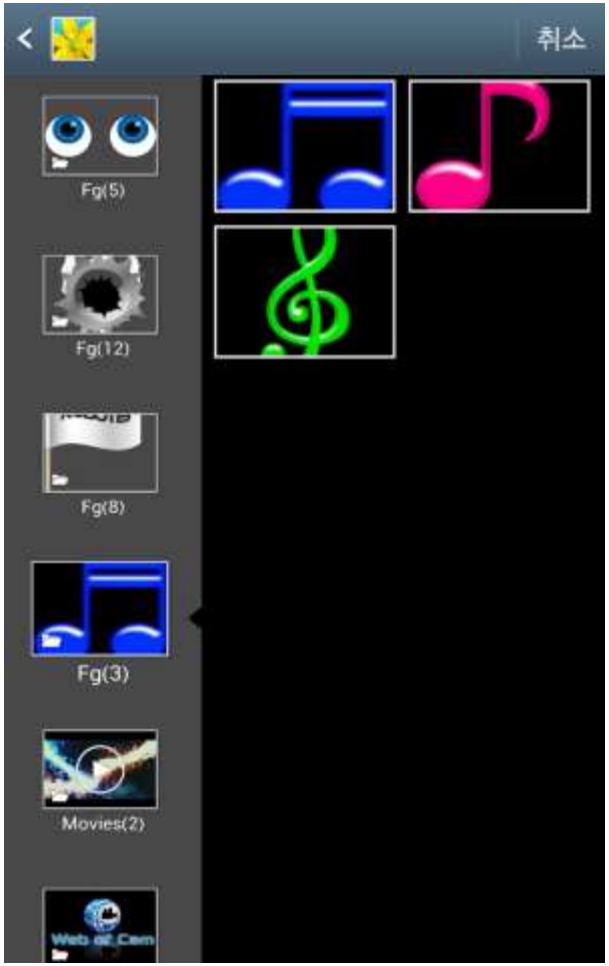


② Click "Picture".

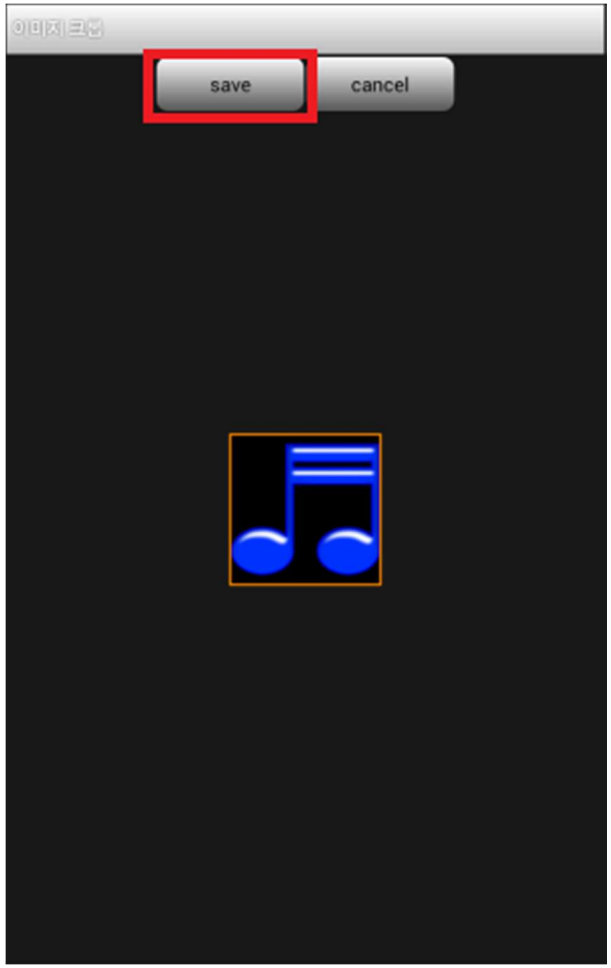


③ Click the "Top" button in the upper right corner, and click "Select from gallery".

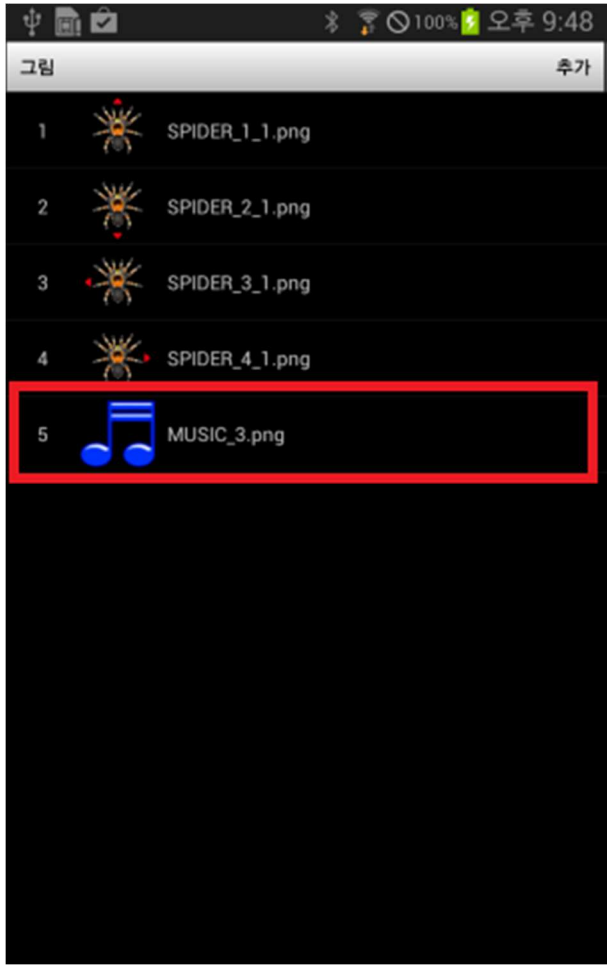
④ Select the image.



⑤ Touch the orange square to select the area to crop, then clip the save button.

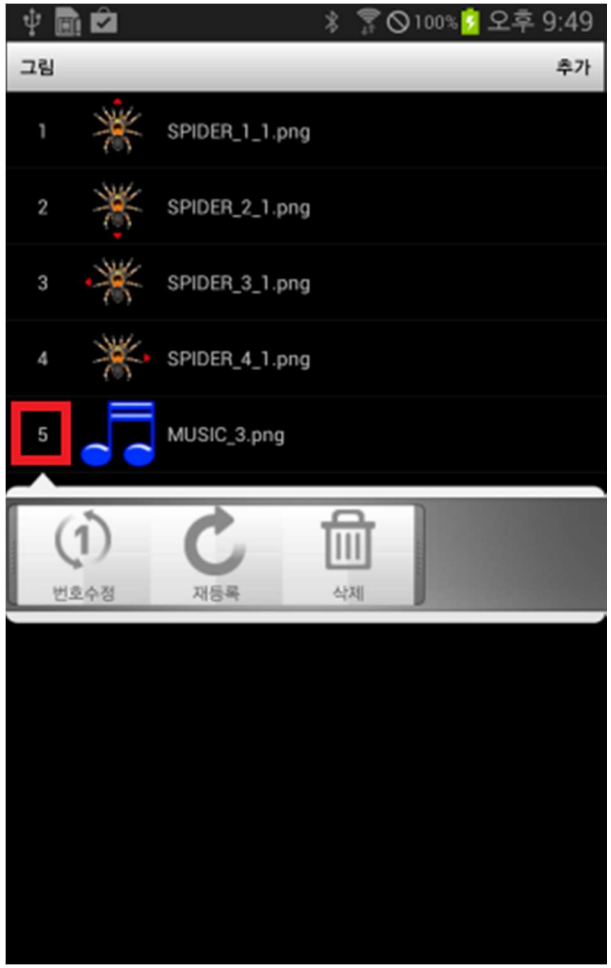


⑥ The image is added.



⑦ The figure can be displayed in the task using the number (IC code number) on the left.

⑧ Click the number on the left to edit, re-register, delete.



4. R + Smart ControlTable

address	name	property	Default	Minimum value	Maximum value	Length	Value Description
10000	Display debug information	Read / Write	0	0		4	Sensor , touch , outputs the information of the detection area on the screen, etc. 0: Off

							1: vision-related positions , color display 2: Sensor values are displayed shake 4 : display vertical and horizontal gradient value 8 : roughness values displayed 16: Display field value 32: Direction value display 64: display noise values 128: Touch Area 1, 2 value display 256: voice input result display 512: SMS -related Phone , Mark Content 1024: The status bar displays a value
10010	Rotate the screen	Read / Write	0	0	2	One	(Read) 0: Auto , 1: vertical , 2 horizontal (Write) 0: Auto , 1: vertical , 2 horizontal
10020	Select camera	Read / Write	0	0	2	One	0: Exit , 1 rear camera , 2: front camera
10030	Camera Zoom	Read / Write	0	0	255	One	0-255: magnification
10040	Select camera function	Read / Write	0	0	5	One	0: blank screen 1: sensing face 2 color paper 3: Detection motion 4: Sensing
10050	Color Detection (raingam terrain)	Read / Write	0	0	5	One	2 black , 3: enemy , 4: Rust , 5: Administration
10060	Record video	Read / Write	0	One	One	One	0: Stop recording , 1: Start Recording During recording, the screen rotation settings , camera selection , Not Noise function .
10070	Photo shoot	Read / Write	0	One	One	One	0: shooting stopped , 1: Recording start
10080	Face Detection Position	read	0	0	25	One	1-25 Display area between [i]

10090	Detection Color (for color detection)	read	0	0	5	One	0: Not detected . Or unknown color 2 black , 3: enemy , 4: Rust , 5: Administration
10100	Motion detection position	read	0	0	25	One	1-25 Display area between
10110	Line Detection Location	read	0	0	5	One	1-5 Display area between
10130	Show background	Read / Write	0	0	199	One	(Reading) last set number of images (Letter) image number to use as a background image 0: background erase (when the camera is used transparently . Other black).
10140	Show picture	Read / Write	0	0		3	Value : (magnification * 65536) + (Image number * 256) + (image position) Image Position 1-25: Screen area between Image number : the registration number of images in the app (Reading) last set number and location and scale images (Letter) image number to be used as a foreground image and the position and magnification Image Number = 0: Deleting images of that location Image Position = 0, the image number = 0: Delete all images * Scaling (1 ~ 10: x magnification , 11 ~ 20: x -fold reduction)
10145	Automatic picture display (Face Sensing)	Read / Write	0	0	199	One	(Read) the current foreground image item number (Letter) automatically displayed using the item numbers of the images in the foreground
10150	Show shape	Read / Write	0	0		4	Value : (shapes colors * 16777216) + (Shape Size * 65536) + (Figure position * 256) + (shape numbers) Shape Color : 0: Unknown 1: white , 2: black , 3: red , 4: green , 5: blue 6: yellow , 7: light gray , 8: Gray , 9: dark gray

							Shape size : 0-255 size values between Shape positions 1-25: Screen area between Shape numbers : 0: Shape removed 1: won , 2: square , 3: Triangle (Reading) the information of the last set shape (Write) information to be displayed on the screen of a shape Shapes position = 0, figure number = 0: remove all shapes
10160	Character display	Read / Write	0	0		4	Value : same as the geometry changes to the shape numbers only letters numbers Character number : the number of the character registered in the app Character number 200: SMS or information stored in message received
10170	Number display	Read / Write	0	0		4	Value : Shapes and replaced with the same number, but the number shapes Number : 0-255: the number is displayed (Reading) the information of the last set number (Write) information on the number to be displayed on the screen Number size = 0: the number of the deleted position Position number = 0, number size = 0: all numbers removed
10180	Sound output (TTS)	Read / Write	0	0	200	One	(Read) 0: Operation Disabled , 1 ~ 200: TTS converts character numbers being

							(Write) 0: Conversion stopped . 1 ~ 200: TTS text to number conversion
10190	Playing musical instruments	Read / Write	0	0		4	Value : (instrument type * 65536) + (octave * 256) + (scale) Instrument Type : see the type of instrument apps (1-128) Octave : 0-10 octave values between Scale : 1 to 12 scale values between (' also ' increased from one semitone)
10200	Audio playback 1	Read / Write	0	0	199	One	(Read), playing audio number (Write) 0: Stop playing . 1-199: play the audio number
10380	Audio playback 2	Read / Write	0	0	199	One	(Read), playing audio number (Write) 0: Stop playing . 1-199: play the audio number
10210	Play video	Read / Write	0	0	199	One	(Read), playing video number (Write) 0: Stop playing . 1-199: play video number
10410	Video suspension	Read / Write	0	0	One	One	(Read) 0: During playback , 1: During stop (Write) 0: Play , 1: stop
10220	Speech recognition	Read / Write	0	0	One	One	(Read) 0: No operation . 1 during operation (Write) 0: Operation stopped . 1: Start operation
10230	Speech recognition result	Read / Write	0	0	199	One	(Read) speech recognition result number of recognized characters (Letter) value initialization
10240	volume	Read / Write	0	0	255	One	Volume Size
10250	wave	read	0	0	255	One	The strength of smartphone shaking
10260	Tilt (left)	read	0	0	90	One	0 to 90
10261	Slope (right)	read	0	0	90	One	0 to 90
10262	Tilt (top)	read	0	0	90	One	0 to 90
10263	Tilt (bottom)	read	0	0	90	One	0 to 90

10270	Illuminance	read	0	0	65535	2	Illuminance sensor value
10280	magnetic field	read	0	0	65535	2	Magnetic field sensor value
10290	Direction (angle value)	read	0	0	359	2	0: north , 90: east , 180: south , 270: West 0 to 359 angle to FIG.
10300	Noise (db)	read	0	0	255	One	The level of noise db Ohm to read
10310	Touch Location 1	read	0	0	25	One	1-25 Display area between
10400	Touch Location 2	read	0	0	25	One	1-25 Display area between
10320	SMS phone number	Read / Write	0	0	200	One	(Read) 0: SMS Not received 1 ~ 199: SMS registration number of the telephone number sent by the 200: SMS phone number is sent to an unregistered numbers (Letter) value initialization
10321	SMS content	Read / Write	0	0	200	One	(Read) 0: SMS Not received 1 ~ 199: SMS registration number of the received text 200: SMS is received, but unregistered text (Letter) value initialization
10330	Notification bar event	Read / Write	0	0	200	One	(Read) 0: katok Not receiving such notification bar 1-199: Notifications of bar code received text 200: notifications are received, but unregistered text (Letter) value initialization
10340	timer	Read / Write	0	0	65535	2	(Reading) returns the remaining time

							(Letter) seconds, the timer value (up to 18 sigan)
10350	Vibration time	Read / Write	0	0	255	One	(Read) 0: Vibration behavior None , 1: vibration during operation (Write) to oscillating time (1/100 seconds)
10360	flash	Read / Write	0	0	One	One	0: Flash Off , 1: Flash On
10370	Launch your app	Read / Write	0	0	199	One	(Reading) Finally, the number of apps running (Letter) to your app code is running)
10390	current time	read	0	0		2	Value : minutes
10420	Send mail	Read / Write	0	0	2	One	(Read) 0: Draft 1: shipment of 2: normal shipping , 3: Error (Write) 0: non-operation 1: The captured image recently , 2: Recording an accompanying video Latest * No such file generated by successive * Recording while if you ship , you do not check the status after sending doemeuro state becomes inaccurate execution of a thread
10430	Screen output	Read / Write	0	0		4	Roboplus functions such as program output view
10440	Line break after screen output	Read / Write	0	0		4	Roboplus functions such as program output view
10450	Gesture	Read / Write	0	0	200	One	(Read) The number of Gesture Recognition (Letter) value initialization