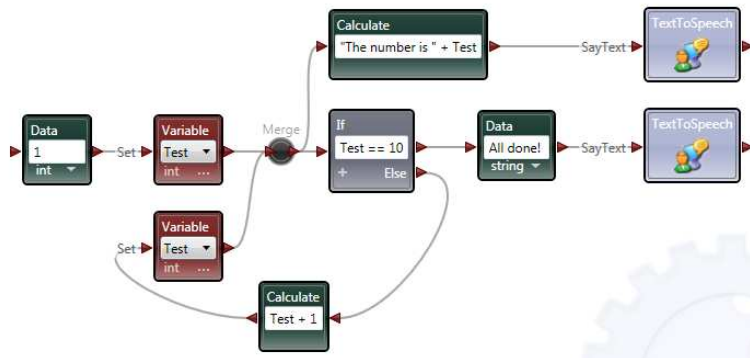


Introduction to Game Programming and Robotics

Unit # 2

VPL Tutorial # 2



Differential Drive

- Many robots use what is called a "Differential Drive" which has two wheels that can be driven independently.
- Although most Differential Drives have two driven wheels, there is often a third passive wheel, called a *castor* or *jockey wheel*, which is just for balance.
- The reason this configuration is popular is that it allows the robot to rotate on the spot. It can therefore drive in any direction after making a tight turn that takes a space no larger than the robot.

Differential Drive (Cont'd)

- RDS defines a *generic contract* for a Differential Drive that specifies the programming interface for controlling a drive regardless of the type of robot that you are using (which is why it is called *generic*).
- The **SetDrivePower** operation sets the power to each of the drive's wheels.

VPL Tutorial # 4

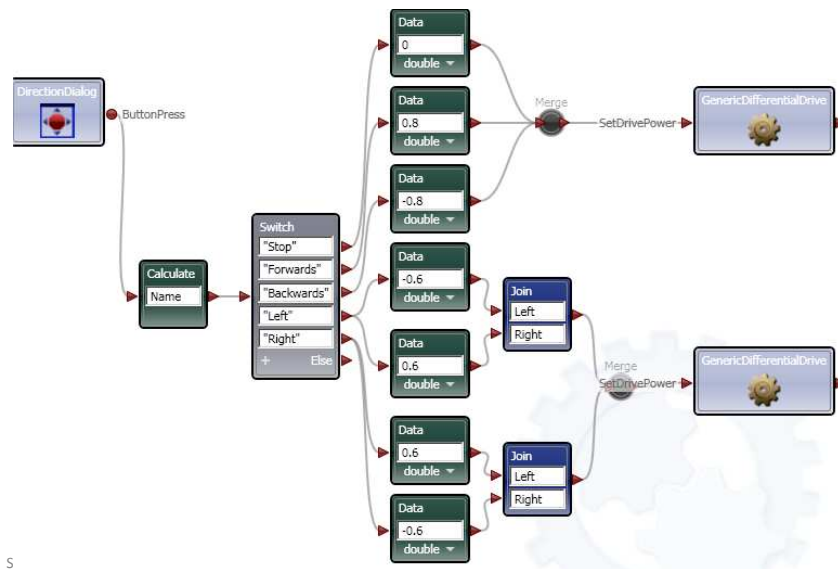
- This tutorial shows how to control a simulated or actual robot through DirectionDialog service.
- Controlling Robot through Direction Dialog (Instructions)
 - If “Stop” is pressed then Set Drive to 0.
 - If “Forward” is pressed then Set Drive to 0.8.
 - If “Backward” is pressed then Set Drive to -0.8.
 - If “Left” is pressed then Set Drive to (0.8, -0.8).
 - If “Right” is pressed then Set Drive to (-0.8, 0.8).

Sajjad Haider

Fall 2009

5

VPL Tutorial # 4

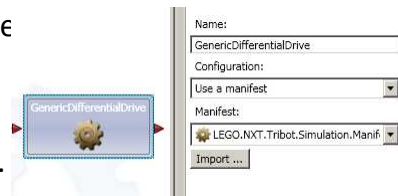


5

6

Simulated vs Actual Robot

- To use an actual Lego robot, change the configuration of "GenericDifferentialDrive" to "Use a manifest" from "none".
- Click on "Import" button and then select "LEGO.NXT.Tribot.Manifest.xml".
- Before you actually use a Lego Robot you have to create a bluetooth connection between your PC and the robot.
- Based on the Com port settings, you have to modify the xml file.



VPL Hands On Lab # 1

- We will write a simple program in VPL to control a robot using a joystick.
 - Actual Joystick
 - Simulated DesktopJoyStick Service

Desktop Joystick Service

- The **Desktop Joystick** service displays a window that exposes the basic capabilities of a joystick which can be driven with a mouse or keyboard.
- As with most activities, it has an input connection *pin* (or *port*) on the left, an output connection pin on the right and a notification pin also on the right.
- We send *requests* to the service via the input pin. These are also called *actions* or *operations*. The *result* or *response* is sent on the output pin.
- Notifications can be sent at any time by a service - we do not have to send a request. These are important for a service like the **Desktop Joystick** because it sends messages constantly when you are moving the joystick. Your VPL program can process these messages and use them to control a robot.

Sajjad Haider

Fall 2009

9

Publish Subscribe Model

- Robotics Developer Studio supports a *Publish-Subscribe model*.
- A service, or in this case your diagram, can *subscribe* to the *notifications* that another service publishes.
- In this example, when the position of the joystick changes the **Desktop Joystick** service will issue a notification.
- The diagram can receive that notification by connecting to the notification pin on the **Desktop Joystick** service.

Sajjad Haider

Fall 2009

10

Connection for

- DesktopJoyStick → GenericDifferentialDrive
 - From: UpdateAxes
 - To: SetDrivePower

Using a Real JoyStick

- In order to use a physical Joystick with this diagram, drag a **Game Controller** service from the **Services** toolbox to the diagram. Then drag from the notification pin on the right of the **Game Controller** service to the middle of the wire connecting the **Desktop Joystick** and **Generic Differential Drive** services.
- When we release the end of the new connection onto the middle of the existing connection a **Data Connection** dialog will pop up.
- On the left side in From, choose **UpdateAxes** as before, on the right side in To choose **MergeConnections**. A **Merge** activity will be added at that point.
- Now our diagram will work with both the **Desktop Joystick** and a physical Joystick at the same time. This is possible because both of the services support the same interface (or *contract* in DSS terminology).

VPL Hands On Lab # 2

- We will learn about driving a differential drive robot in arcs, and splicing those arcs together to make more complicated trajectories.

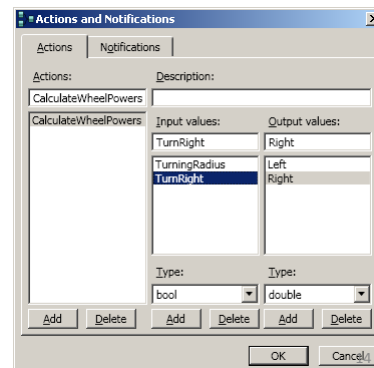
Sajjad Haider

Fall 2009

13

Inside TurningRadiusToWheelPowers

- Drag an **Activity** block from the toolbox onto the diagram and name it to **TurningRadiusToWheelPowers**.

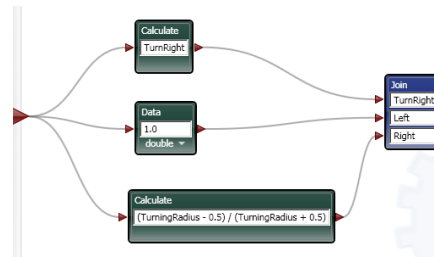


Sajjad Haider

Fall 2009

Inside TurningRadiusToWheelPowers

- The first **Calculate** simply extracts the **TurnRight** value from the incoming message so that it can be added to the **Join**.
- The **Data** block sets the power for the left wheel to 1.0.
- The second **Calculate** does the real calculations which are intended to figure out an appropriate value for the right wheel power.



Inside TurningRadiusToWheelPowers

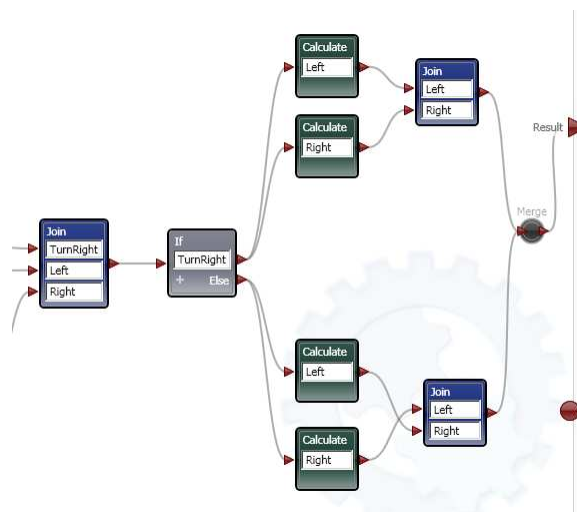
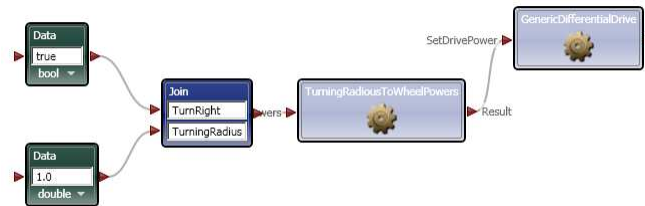


Diagram on the Main Page



- Services only allow one incoming message type per connection, since a connection corresponds to an execution path.
- However, a service allows multiple outgoing connections, because a single path might **spawn** multiple concurrent tasks.
- To aggregate the two pieces of data, we must add a **Join** block. This will create a compound message out of the two values, allowing us to access the values by the names we specify.

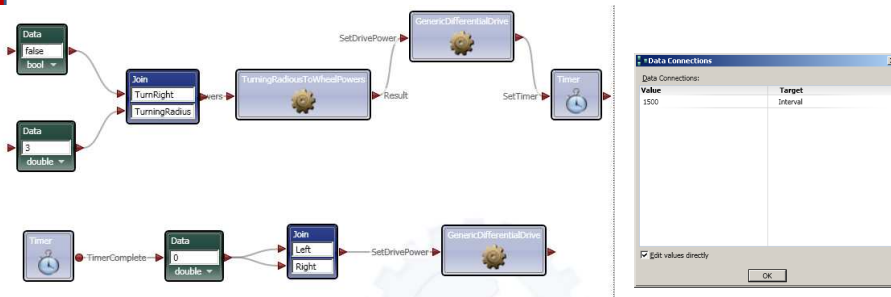
Precaution

- Before we run this program to drive the robot, we should consider the safety aspects.
- If we simply run the program, then the robot will start up immediately when the program starts.
- This might take somebody by surprise.
- Therefore we should have a "start button" so that we can tell it when to begin moving.
- If we are using a simulated robot then this is not a safety issue, but it is still good practise because the simulator takes a little while to start up.

Step 2

- In Step 1, the robot continues driving in circles because its motor powers are still set.
- To drive along an arc, we must stop the motors after some period of time, i.e. set the powers to zero.

Step 2 (Cont'd)



Step 3 and 4

- Homework Assignment