STM32 and Jetson Nano Gyroscope Controller Project

This project establishes a communication bridge between a Jetson Nano and the STM32F3Discovery board. Utilizing Python, the goal is to gather data from the STM32F3Discovery's built-in gyroscope, process this data on the Jetson Nano, and then control the STM32F3Discovery board's LEDs to indicate the board's roll and pitch orientation. This documentation assumes the reader has basic familiarity with accessing the Jetson Nano, which in this scenario is through an SSH connection, although other interface methods are also acceptable. The project uses Visual Studio Code 1.762.2 for SSH access to the Jetson Nano and STM32CubeIDE 1.11.2 for programming the STM32 board.

Prerequisites

- STM32F3Discovery board
- Jetson Nano
- USB Mini B cable for STM32F3Discovery connection
- USB A to USB Mini B cable for Jetson Nano connection
- Visual Studio Code with SSH access setup
- STM32CubeIDE for STM32 programming

Environment Details

To ensure this project runs smoothly on your setup, here's a detailed overview of the required environment, including software and hardware specifications, library dependencies, and additional configuration details.

Hardware Requirements

1. STM32F3Discovery Board: The primary device for this project, used for demonstrating gyroscopic data collection and LED indication.
2. Jetson Nano: Acts as the computing platform for processing data received from the STM32F3Discovery board and sending back control commands.

Software Requirements

1. Operating System: Linux-based OS on the Jetson Nano, compatible with the majority of development and serial communication tools used in this project.
2. STM32CubeIDE: Version 1.11.2 or higher for programming and flashing the STM32F3Discovery board.
3. Visual Studio Code: Version 1.762.2 or higher, with SSH extension for remote development on the Jetson Nano.
4. Python: Version 3.x installed on the Jetson Nano, ensuring compatibility with the project's Python scripts.
5. PySerial: A Python library is required for serial communication between the Jetson Nano and the STM32 board. It can be installed using the following command:

```
pip install pyserial
```

Setup and Installation

STM32F3Discovery Board Setup

1. **Programming the Board:**
    - Open STM32CubeIDE and select or create a new workspace.
    - Go to File > Import > General > Existing Projects into Workspace.
    - Select the archive file containing the project code and import it.
    - Navigate to Project Explorer, find your project, and locate the main.c file under Core > Inc. Review or modify it as needed.
    - Ensure the STM32F3Discovery board is connected to your computer via the USB Mini B cable. The connection port is located centrally on the board.
    - Use the Run feature in STM32CubeIDE to compile and upload the code to the STM32F3Discovery. For first-time uploads, you might need to configure the debugger settings to recognize the ST-LINK. Select Run > Debug Configurations, choose the appropriate debugger, and press Scan to auto-fill the configuration.
2. **Connecting to Jetson Nano:**
    - After programming, disconnect the STM32F3Discovery from your computer and connect it to the Jetson Nano using the appropriate USB port.

Jetson Nano Setup

1. **Preparing the Environment:**
    - Open a terminal on the Jetson Nano and create a project directory: mkdir ~/Project_1 && cd ~/Project_1.
    - Ensure the three Python files (main.py,onboard_comm.py and stm32_conn.py) are placed in this directory. These can be transferred via SSH or any other file transfer method.
2. **Running the Project:**
    - Before executing the Python script, you may need to adjust permissions for the serial port: sudo chmod 666 /dev/ttyACM0.
    - Execute the main.py script: python main.py.
    - Follow the on-screen prompts to interact with the system. Options include running the system to monitor and indicate the board's orientation, calibrating the gyroscope for sensitivity adjustments, and exiting the application.

Usage

- Option 1 (Run): This mode uses the STM32F3Discovery's built-in LEDs to indicate the board's orientation based on the gyroscope data.
- Option 2 (Calibrate): Allows for calibration of the gyroscope's sensitivity settings. This is useful if the sensor readings are too rigid or overly sensitive.
- Option E (Exit): Allows to stop the program.

Calibration Process

1. Select from one of the four calibration options provided.
2. Position the STM32F3Discovery for the intended tilt direction and press enter.
3. Tilt the board slowly in the chosen direction for 3-5 seconds. The system will automatically save the new sensitivity settings.

Troubleshooting

Common Issues and Solutions

1. Serial Port Access Denied: If you receive an error indicating that access to the serial port is denied (e.g., /dev/ttyACM0), this is likely a permissions issue. Solve it by running sudo chmod 666 /dev/ttyACM0 in the terminal to grant read and write permissions.
2. STM32F3Discovery Not Recognized: Ensure the USB cable is properly connected to both the STM32 board and the Jetson Nano. If the board is still not recognized, try using a different USB port or cable.
3. Failure to Run the Python Script: If main.py fails to execute, confirm that Python 3 is installed and accessible from your terminal. You might need to use python3 main.py instead of python main.py depending on your system's configuration.
4. Error in LED or Gyroscope Functionality: Double-check the connections and ensure that the STM32F3Discovery board has been correctly programmed with the provided code. Errors in LED or gyroscope functionality can also stem from incorrect calibration, so revisiting the calibration steps might be necessary.
5. Problems with STM32CubeIDE: If you encounter issues importing or running the project in STM32CubeIDE, ensure that your IDE is up to date and that you've selected the correct target device (STM32F3Discovery). Review the import steps to make sure the project settings have been properly configured.
6. Jetson Nano Connectivity Issues: Ensure your SSH connection to the Jetson Nano is stable and that the Nano is correctly set up on your network. Connectivity issues can often be resolved by checking your network settings or consulting Jetson Nano's documentation for troubleshooting steps.

References

The project builds on concepts and resources provided by Mr. Maxwell Hogan and Dr. Zakaria Chekakta. This project was realized in the context of the module EPM104 by:

- Safinaz Khalil
- Sheshta Ramgoolam
- Paul Gin Hwan Cheung

Contact for Further Assistance

If you encounter issues not covered by this guide, consider reaching out to relevant online communities or forums. For hardware-specific issues, the STM32 and Jetson Nano official forums are valuable resources. For programming-related inquiries, Stack Overflow and the Python community can offer support. Or you can contact one of the following code creators:

Safinaz.khalil@city.ac.uk

Sheshta.Ramgoolam-Sobun@city.ac.uk

Paul-Gin-Hwan.Cheung@city.ac.uk