```
Langkah-langkah Kontrol Dynamixel AX-18A dengan Publish-Subscribe:
1. Komponen yang Dibutuhkan:
Raspberry Pi dengan ROS terinstal.
Dynamixel AX-18A servo.
Modul U2D2 (untuk komunikasi USB-TTL).
Kabel TTL dan power supply (12V untuk Dynamixel).
dynamixel_sdk ROS package.
2. Instalasi ROS dan Dynamixel SDK di Raspberry Pi:
Update sistem:
bash
Copy code
sudo apt update && sudo apt upgrade
Instal ROS Noetic:
bash
Copy code
sudo apt install ros-noetic-ros-base
Buat workspace ROS dan instal dynamixel_sdk:
bash
Copy code
mkdir -p ~/catkin_ws/src
cd ~/catkin ws
catkin_make
sudo apt install ros-noetic-dynamixel-sdk
3. Koneksi Hardware:
Hubungkan Dynamixel AX-18A ke U2D2 menggunakan kabel TTL.
Sambungkan U2D2 ke Raspberry Pi via USB dan colokkan power supply ke servo.
4. Konfigurasi Servo:
Gunakan ROS untuk mengecek koneksi servo:
bash
Copy code
rosrun dynamixel_workbench_controllers find_dynamixel /dev/ttyUSB0
5. Setup Publish-Subscribe di ROS:
Buat publisher untuk mengirim perintah posisi servo:
python
Copy code
import rospy
from std_msgs.msg import Int32
def servo command():
  pub = rospy.Publisher('servo_angle', Int32, queue_size=10)
  rospy.init_node('servo_publisher', anonymous=True)
  while not rospy.is shutdown():
    pub.publish(512) # posisi default
    rospy.Rate(10).sleep()
if __name__ == '__main__':
  servo command()
```

Buat subscriber untuk menerima perintah dan menggerakkan servo:

```
python
Copy code
import rospy
from std msgs.msg import Int32
from dynamixel_sdk import *
def callback(data):
  portHandler = PortHandler('/dev/ttyUSB0')
  packetHandler = PacketHandler(1.0)
  portHandler.openPort()
  portHandler.setBaudRate(1000000)
  packetHandler.write2ByteTxRx(portHandler, 1, 30, data.data)
  portHandler.closePort()
def servo_listener():
  rospy.init node('servo subscriber', anonymous=True)
  rospy.Subscriber('servo_angle', Int32, callback)
  rospy.spin()
if __name__ == '__main__':
  servo listener()
6. Jalankan Sistem:
Jalankan roscore:
bash
Copy code
roscore
Di terminal lain, jalankan publisher dan subscriber:
bash
Copy code
rosrun your_package_name servo_publisher.py
rosrun your package name servo subscriber.py
7. Uji Pergerakan:
Servo akan bergerak sesuai perintah dari publisher. Sistem ini sangat berguna untuk
mengontrol pergerakan servo melalui komunikasi jarak jauh menggunakan
publish-subscribe dalam ROS.
```

Dengan langkah ini, Anda bisa mengontrol Dynamixel AX-18A secara dinamis dan fleksibel untuk berbagai proyek, mulai dari robotika hingga otomasi.