

ROBOT PIONEER

Première séance (23/10/17) :

Mr Marmoiton étant absent, j'effectue des recherches sur le Pioneer 3AT sur le WEB

Pioneer est une gamme de robot développée par la société MobileRobots Inc (OMRON Adept ?)

On peut utiliser ce robot avec le logiciel open source ROS ou ARIA

Fiche de présentation: <http://www.mobilerobots.com/Libraries/Downloads/Pioneer3AT-P3AT-RevA.sflb.ashx>

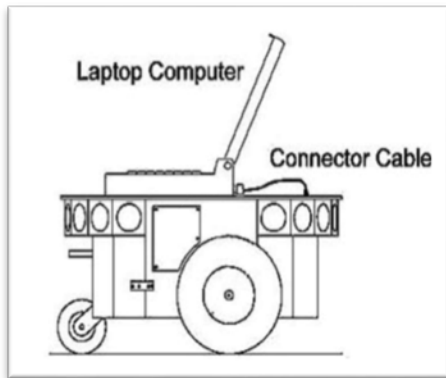
Datasheet: http://www.inf.ufrgs.br/~prestes/Courses/Robotics/manual_pioneer.pdf

Datasheet – Chapitre 5: “Quick Start” p.21

“The procedures we describe herein are for control of the basic robot”

Je vais donc faire un résumé du chapitre 5

Port série :



Tout d'abord, disposer le PC sur le Pioneer (voir ci-contre) et s'assurer que câble (connecteur 9-pin DSUB & protocole RS-232) est bien connecté au port du panneau de contrôle utilisateur

NB : D'autres liaisons sont possibles (Ethernet, sans fil, ...)

Installation des batteries :

Les batteries sont livrées complètement chargées

Le cas échéant, recharger les à l'aide du système fourni

Faites glisser au moins une et jusqu'à trois batteries dans le compartiment prévu par la porte arrière.

Disposez-les astucieusement pour équilibrer la surcharge de poids.

Logiciel ARIA :

Il y a beaucoup de logiciels pour utiliser le robot. Le meilleur d'entre eux est ARIA qui propose de nombreuses démo afin de tester tous les accessoires.

Les utilisateurs Windows peuvent sélectionner la démo ARIA dans le menu Démarrer du groupe de programmes MobileRobots.

Les utilisateurs de Linux trouveront la démo compilée dans « `/usr/local/Aria/bin/` » ou dans « `examples/` ». Commencer avec : « `%.demo` »

Par défaut, le programme connecte le pc au robot via le port « COM1 ». Pour paramétrer la connection du port, il faut le modifier dans le code source d'ARIA (*examples/demo.cpp* et voir les fichiers dans « `src/` ») et recompiler.

Connexion réussie :

Le microcontrôleur émet un signal sonore audible.

Vous devriez aussi entendre un clic distinctif et répétitif du au sonar.

Enfin, le voyant STATUS du panneau de contrôle utilisateur doit clignoter très rapidement (il clignotait lentement en attendant la connexion).

Vous pouvez maintenant prendre le contrôle du robot à l'aide du clavier :

Table 2. Keyboard teleoperation

KEY	ACTION
↑	forward
↓	reverse
←	turn left
→	turn right
space	all stop

Table 2. ARIA demo operation modes

MODE	HOT KEY	DESCRIPTION
laser	l	Displays the closest and furthest readings from the laser range finder
io	i	Displays the digital and analog-to-digital I/O ports
position	p	Displays the coordinates of the robot's position relative to its starting location
bumps	b	Displays bumpers status
sonar	s	Displays the sonar readings
camera	c	Controls and exercises the pan-tilt-zoom robotic camera
gripper	g	Controls, exercises and displays status of the Gripper accessory
wander	w	Sends the robot to move around at its own whim while avoiding obstacles
teleop	t	Drive and steer the robot via the keyboard or a joystick; avoids collisions
unguarded	u	Same as teleop, except no collision avoidance
direct	d	Direct command mode

A défaut de trouver un programme de test, j'ai listé les fonctions et exemples intéressants présents dans la librairie que vous m'avez envoyée.

Tests:

connectTest – « Connects to the robot, disconnects, and tries to break the connection state. This was designed to test the connection sequence. This uses `ArRobot::blockingConnect`. »

→ Test de la connexion

encoderCorrectionTest - Connects to a robot with a joystick, pressing button two will set the encoder correction callback, just run it to see the args

→ Test du robot grâce aux joysticks

Exemples:

simpleConnect – « The smallest example program. It just connects to the robot »

→ Connexion au robot

lasers – « Demonstrates how to connect to one or more laser rangefinder devices as defined in robot and program parameters, and obtain data from them »

→ Test de la connexion avec les lasers et renvoie les données obtenues

simpleMotionCommands – « Drives the robot around using the basic direct motion commands (no `ArActions` or obstacle avoidance) »

→ Contrôler le robot grâce aux joysticks

gripperDemo – « Program that moves the robot and controls the gripper with the joystick, note this doesn't do obstacle avoidance »

→ Contrôler le robot grâce aux joysticks sans éviter les obstacles

joydriveUserTask – « Program to drive the robot with a joystick. This one uses a user task to drive the joystick handler, does not do obstacle avoidance »

→ Contrôler le robot grâce aux joysticks

teleopActionsExample – « This uses ARIA's powerful Actions system to drive the robot around using input from the keyboard or a joystick, but does obstacle avoidance so the robot won't run into things (if it can sense them with sonar or laser). »

→ Contrôler le robot grâce aux joysticks en évitant les obstacles

Ces programmes sont disponibles dans les dossiers « tests » et « exemples »

Après avoir suivi de nombreux tutoriels pour enfin tester le Pioneer 3-AT

Je n'ai pas réussi à compiler un programme adéquat

J'ai décidé d'explorer le dossier ARIA présent sur le bureau après installation

Il y figure un fichier DEMO.exe

Ce fichier se connecte au robot via le port USB et permet de tester certaines fonctionnalités du robot.

On peut le piloter manuellement à travers les touches directionnelles du clavier.

A l'approche d'un obstacle, le robot s'arrête, puis reprend sa trajectoire lorsque l'on enlève l'obstacle.

On peut aussi obtenir plus d'infos sur les équipements en tapant quelque commande.

Je pense qu'on peut appliquer d'autres tests via la fenêtre de commande prompt mais je n'ai pas eu le temps d'en faire le test