Overview

Nowadays, robot cannot response immediately or ignores some signal when handling multitasks since the computational power of Robotic MCU is not sufficient to deal with these tasks. Take Robocon as a case study, using ATmega128 as MCU which cannot handle the whole input data from sensor, therefore, we would like to develop robotic Hardware Architecture which is replaced ATmega128 by ARM9 to solve these existing problems.

Aim & Objective

According to the background of Robocon, the current core board does not enough performance to maintain the robots. The aim of designing the low-voltage and multi-function ARM9 core circuit board is to optimize its performance including highly computing power, real-time system and multiple algorithms. To deal with multi-tasks, ARM 9 is computing a large amount of input data within a very short period.
System Block Diagram

Hardware Part:

- User Control Interface
- Protocol
- AVR GCC (ATmega128)
- Develop the Control Programs
- • Servo Signal Generation
  • Motor Signal Generation
  • PWM Single Generation
  • Low speed sensor response
  • SPI Communication
  • IIC Communication
- ARM920T (SX2410)
- Install the Linux Kernel
- Develop the Driver Programs
  • Webcam
  • USB
  • sound card
  • LAN629
- Wireless LAN
  • Bluetooth
  • Color LCD

Software Part:

- User Control Interface
- Protocol
- ARM920T (SX2410)
- Install the Linux Kernel
- Develop the Driver Programs
  • Webcam
  • USB
  • sound card
  • LAN629
- Wireless LAN
  • Bluetooth
  • Color LCD

The System Hardware:

Whole System features:

- Fully support the old system of Robocon
- Developed new Linux Kernel for the board
- Bluetooth communication between ARM boards
- Supported Web cam plug in
- Wireless & Wire line LAN communication to PC
- Achieved well Multithread performance
- Shorten the duration of find the shortest path of the destination
- Color LCD display
Testing and Performance

Find the shortest path

For testing the real application for Robocon game, we aim to detect the duration which the robot finished for the task, which found the shortest path in the map.

<table>
<thead>
<tr>
<th>MAP Size (mm)^2</th>
<th>ARM920T</th>
<th>ATmega128</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Map1 = 10000x10000</td>
<td>0</td>
<td>500</td>
<td>ms</td>
</tr>
<tr>
<td>Map2 = 20000x20000</td>
<td>0</td>
<td>1093</td>
<td>ms</td>
</tr>
<tr>
<td>Map3 = 50000x50000</td>
<td>552</td>
<td>9170</td>
<td>ms</td>
</tr>
<tr>
<td>Map4 = 100000x100000</td>
<td>2460</td>
<td>40900</td>
<td>ms</td>
</tr>
<tr>
<td>Map5 = 500000x500000</td>
<td>14200</td>
<td>237000</td>
<td>ms</td>
</tr>
</tbody>
</table>

Result

Obviously, the required time of ARM for finding the shortest path is much less than that of ATmega. As map size increase, the time difference between ARM9 and AT mega becomes more significant and dominant.

Conclusion

The embedded system was used as the operating system on the ARM9 core board to develop a main application, communicate with the ATmega128 boards. We built a stability and economical platform for our designed system specification. However, what we did is that building up a high performance computational in Hardware system for supporting the programmers to develop the complex algorithms such as vision recognition, motion planning in artificial intelligence vision recognition is feasible to develop because our PCB support USB Web Cam and can upload the Real Time situation to the Websites. The goal of the project has been achieved.