Parallel Processing System Design with "Propeller" Processor

Yoichi Nagashima
Shizuoka University of Art and Culture
1-1, chuo-2, Naka-ku,
Hamamatsu, Shizuoka JAPAN
nagasm@suac.ac.jp

Abstract

This is a technical and experimental report of parallel processing, using the "Propeller" chip. Its eight 32 bits processors (cogs) can operate simultaneously, either independently or cooperatively, sharing common resources through a central hub. I introduce this unique processor and discuss about the possibility to develop interactive systems and smart interfaces in media arts, because we need many kinds of tasks at a same time with NIME-related systems and installations. I will report about (1) Propeller chip and its powerful IDE, (2) external interfaces for analog/digital inputs/outputs, (3) VGA/ NTSC/PAL video generation, (4) audio signal processing, and (5) originally-developed MIDI input/output method. I also introduce three experimental prototype systems.

Keywords: Propeller, parallel processing, MIDI, sensor, interfaces.

1.Introduction

Propeller [1] is supported by Parallax Inc. With its internal eight processors, we have full control over how and when each cog is employed; there is no compiler-driven or operating system-driven splitting of tasks among multiple cogs. A shared system clock keeps each cog on the same time reference, allowing for true deterministic timing and synchronization.

We can use two programming languages: the easy-to-learn high-level Spin, and Propeller Assembly which can execute at up to 160 MIPS (20 MIPS per cog). There is a popular technique "Interrupt" to realize multi-task with all CPU. However, Propeller doesn't have the "Interrupt" because parallel processing is controlled by its special hardware. Its resources - common memories (32KB RAM /ROM) and 32 external I/O pins - are automatically assigned to round-switched cogs.

We can easily make parallel processing software for Propeller by the smart and powerful IDE, without special consideration for synchronization. Because I have no enough space to introduce more both Propeller's languages and Propeller's IDE here, please refer my analyzing/ experiments report [2]. This website is currently only in Japanese, but the English version will be available.

2.Propeller interfaces

Propeller has 32 I/O pins that can be accessed by each cog with double or more monitoring and overwriting. Each cog has special timing circuits for counter/timer modes.

2.1MIDI input / output

Propeller can deal serial communications like MIDI only by software, without special hardware like UART. There is a sample MIDI-in object in the Parallax web page [3], but I arranged and developed the universal MIDI-in/MIDI-out module [2]. This module deals MIDI information with deep Rx/Tx FIFO buffers in common memory in the chip, so it is easy to make intercommunication of each cog. Figure 1 shows the original circuits with MIDI I/O, audio D/A and NTSC video output (described later).

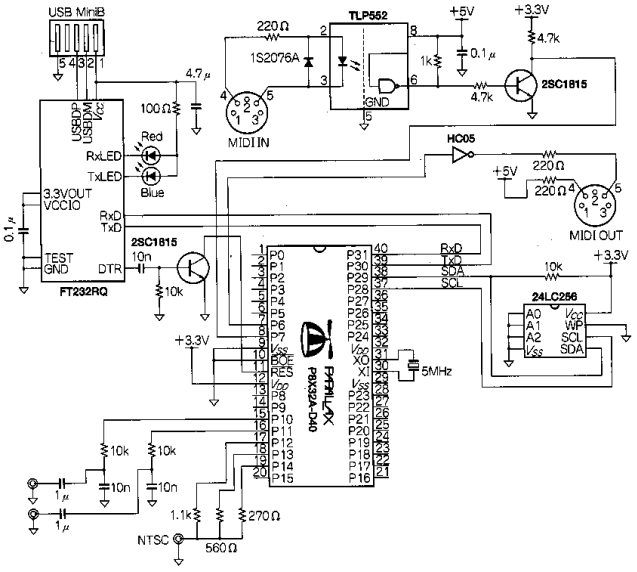

Figure 1. Prototype System for Propeller processor.

2.2D/A output and A/D input

Propeller can generate an analog output signal by PWM converter with external 2 capacitors and 1 resistor. Thus Propeller can generate easily 44.1 KHz sampling, 16 bits stereo audio signal with 2 external pins and 1 internal cog.

We can also get A/D input with reference D/A output by 1 internal cog and 2 pins, using Parallax's sample program. The Propeller's cog is fast enough for CD quality sampling conversion. I discovered and confirmed that the assembly language of Propeller was specially designed to achieve audio signal processing compactly and efficiently.

2.3Analog (RC) input

Because each cog of Propeller has an individual timing circuit in it, RC-type (time constant circuit) A/D conversion is simpler to construct. We can easily achieve high accuracy and high speed A/D by the specification of Propeller (80MHz, 32 bits). We can use many types of sensors: CdS, strain-gauge, piezo, static-electricity, carbon-rubber, electric-capacitor, etc.

2.4NTSC/PAL/VGA video output

Surprisingly, Propeller can generate the video signal of NTSC/PAL/VGA in the background, with 1 cog for graphic driver and 1 cog for video signal D/A with only 3 external resistors. So, we can produce many multimedia systems with using only 2 cogs of Propeller, and we can use remaining cogs for 6 individual parallel tasks.

3.Experimental Applications

After this research, I produced three experimental systems, and the third one is a media-installation work.

3.1Double NTSC output

Using 4 cogs for double NTSC video outputs, I developed a multi monitor MIDI-CG system. However, by the limitation of Propeller's internal memory, the double CG drawing-mode were only "storage-display like" without double-buffer computing.

3.2Propeller Compact Display Module

Next, I produced a compact display module for CG generator and MIDI monitor. The small module is supported by Little PCB Solutions [4], and I developed the prototype Compact Display Module. The system generates 3 patterns of realtime CG and MIDI display by Hexadecimal format (see Figure 2).

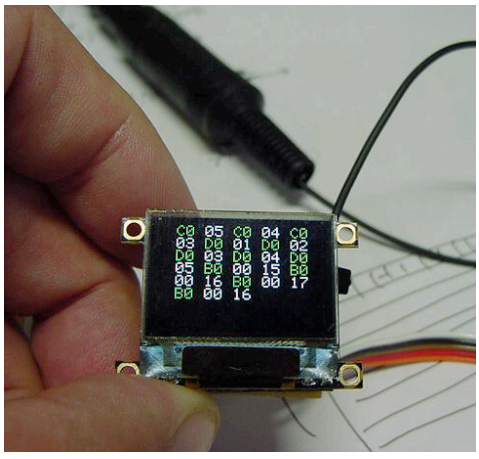

Figure 2. Prototype Compact Display Module.

3.3"Dodeca Propeller"

The newest work with Propeller is the installation work called "Dodeca Propeller". This was presented at Media Art Festival 2008 at SUAC in Japan in December 2008.

Figure 3 shows the system board and there are 13 Propeller chips on it. We have two huge display systems that arrange 12 large-scale video monitors in SUAC. "Dodeca Propeller" was designed for this display system, so the output is 12 NTSC video lines.

12 display Propeller run the real-time CG generating program. The "controller" Propeller works as: MIDI control (foot switch sensors) receiver, individual real-time On/Off switcher for each 12 screen, and the sound sensor in the gallery hall to control display patterns. As an important point, PC doesn't exist in the system that achieves this complex real-time generation of 12 screens CG and the interaction with the sensors. I developed this system with 2 undergraduate students. It was very easy to make CG programs of Propeller for the students, and I think that Propeller is a very good platform for education of students' programming.

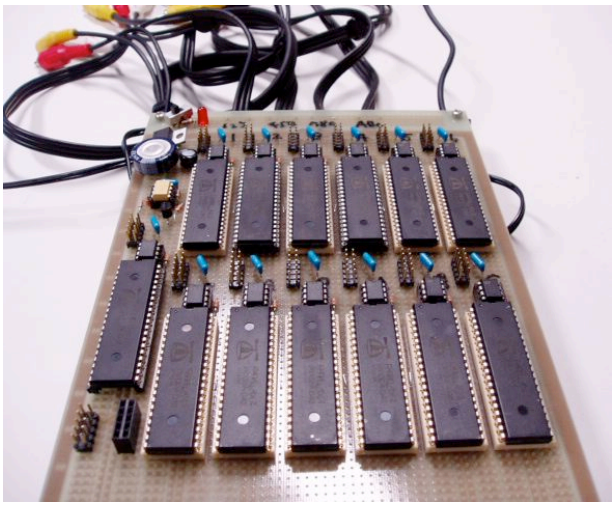

Figure 3. System Board of "Dodeca Propeller".

4.Conclusions

This is my first report of the "Propeller" chip. I think (1) it has deep possibility to design interactive systems, and (2) it is good for education in computer programming. I will report next step in the future with some projects in media arts.

References

[1] Propeller General Information, Parallax Inc., http://www.parallax.com/tabid/407/Default.aspx

[2] Yoichi Nagashima, Propeller Diary, http://nagasm.suac.net/ASL/Propeller/index.html

[3] Tom Dimock, Propeller MIDI in objects, http://obex.parallax.com/objects/229/

[4] µOLED-96-PROP, Little PCB Solutions, https://www.littlepcbsolutions.com/uOLED-96-PROP.html