DESCRIPTION OF RADIO-BATON SIGNALS

Max Mathews, June 6, 2006

HOW IT WORKS

The radio-baton consists of an antenna board, an electronics board, and two batons. The antenna board contains four antennas, two complimentary wedge- shaped antennas to measure the x position of the batons and two complimentary bar-shaped antennas to measure the y position of the batons. Each of the two batons contains an oscillator transmitting a signal from the end of the baton. Baton #1 transmits at 38khz and baton #2 transmits at 50khz. The baton analogue electronics amplify, filter, and detect 8 signals, 4 signals from each of the 2 batons. These 8 signals are proportional to the signal strengths of the radio-waves received from the batons. The signal strengths at a given antenna and baton depend directly on the “area” of the antenna directly under the baton and inversely on the distance from the baton to the antenna.

The baton electronics also contains a small computer with eight 12 bit AD converters, which receive the 8 detected signals from the analogue electronics. The computer encodes these signals into 8 standard midi commands which are used in a “non-standard” midi way. The 8 midi commands are sent from a midi connector on the electronics board to any computer or other device that can receive midi signals. Each command encodes the 12-bit output of one AD converter. This number is proportional to one of the 8 signals detected by the analogue electronics.

In addition to the 8 AD inputs, the small computer receives 5 digital (on-off) signals from 5 buttons-switches on the analogue board and 7 digital signals from foot-switches which can be connected to the analogue board via a ribbon cable. These 12 bits are encoded onto a ninth midi command.

A midi command takes about 1ms to send so the entire cycle of information From the baton takes about 9ms to send the cycle is repeated about 110 times per second.

MIDI ENCODING OF DATA

A standard midi command contains three bytes byte0, byte1, and byte2. The byte0 is the status byte and its high order bit must be 1. The high order bit of byte1 and byte2 must be 0. The other 7 bits of these bytes can be anything.

In the standard midi command, the high order 4 bits of the status byte specify the op code and the low order 4 bits specify the midi channel number. We will not follow this standard. Our encoding will be as follows:

0xAn byte1 byte2

where for n=0 …7, n specifies the converter number

byte1 encodes the least significant 7 bits of the AD converter n

byte2 encodes the most significant 5 bits of the AD converter n

and for n=8

byte1 encodes the 5 bits of buttons 1 through 5

byte2 encodes the 7 bits of the 7 foot switches

CONVERTING AD DATA TO XYZ POSITIONS OF STICKS

The following is a slightly simplified C program for converting the AD outputs to xyz positions of the batons. The constants in the equations are chosen so x & y are in standard midi range 0 to 127. The estimate of z is simply the sum of the AD's for the four antennas for each stick.

```
//midiparser--for standard 3 byte midi messages
void  midiparser( byte0,byte1,  byte2)
{
    unsigned short xxx,arrayindex;
    int ii;
    arrayindex=byte0 & 0x0f;
    xxx=byte2<<7)+byte1;
    if(arrayindex!=8){array[arrayindex]=(float)xxx;}
    else
    {
      buttons=byte1;
      feet=byte1;
      xc1=(array[7]/(array[7]+array[1]))*128;
      yc1=(array[5]/(array[5]+array[3]))*128;
      zc1=array[7]+array[1]+array[5]+array[3];

      xc2=(array[6]/(array[6]+array[0]))*128;
      yc2=(array[4]/(array[4]+array[2]))*128;
      zc2=array[6]+array[0]+array[4]+array[2];
     }//end else
}//end processmidicommandforscore
```

RELATION OF AD CONVERTERS TO THE ANTENNAS & BATONS

AD	FREQ KHZ	ANTENNA
0	50	X LEFT
1	38	X LEFT
2	50	Y BOTTOM
3	38	Y BOTTOM
4	50	Y TOP
5	38	Y TOP
6	50	X RIGHT
7	38	X RIGHT

BATON #1--38KHZ
BATON #2--50KHZ