A UTMI-COMPATIBLE PHYSICAL-LAYER USB2.0 TRANSCEIVER CHIP

Jang-Jin Nam, Yong-Jun Kim, Kwang-Hee Choi, Hong-June Park
Dept. of Electrical Eng., Pohang University of Science and Technology(POSTECH), Korea
hjpark@postech.ac.kr

ABSTRACT
A physical layer(PHY) USB2.0 transceiver chip was designed and fabricated by using a 0.25um CMOS technology. The PHY chip is compatible to the USB2.0 Transceiver Macrocell Interface(UTMI) specification. It has a 16-bit parallel interface to its link layer. The electrical tests on the fabricated chip confirmed the successful operation of analog circuitry such as the clock data recovery(CDR) circuit, the output drivers, and the receiver circuits for both high-speed(480Mbps) and full-speed(12Mbps) data transmissions. The BERs were measured to be less than 1E-12 for both data transmissions. Functional verifications for USB2.0 protocol by using a logic analyzer with a pattern generator showed the successful operations of digital circuitry such as the bit stuffer/un-stuffer and the NRZI encoder/decoder. The chip area excluding the IO pads was 0.91 x 1.82mm². The power consumptions at the supply voltage of 2.5V were 225mW and 150mW for high-speed and full-speed operations, respectively.

Index Terms — USB2.0, CMOS transceivers, SERDES, clock recovery

I. INTRODUCTION
For the interface of multimedia data between PCs(personal computers) and peripheral components, the USB2.0 interface chip is being widely used. The USB2.0 interface chip supports the high-speed(HS, 480Mbps) and full-speed(FS, 12Mbps) data rates.

Fig. 1 illustrates a simplified block diagram of the USB2.0 interface chip for USB device applications. The chip is located between a PC and peripheral components such as hard disk, DVD driver and printers. It is connected to a PC or a USB2.0 hub via a USB cable, which is a twisted un-shielded cable with the maximum length of 5 meters. The USB2.0 interface chip consists of the USB 2.0 transceiver and the serial interface engine. The USB 2.0 transceiver is also called the USB 2.0 physical layer(PHY) and the serial interface engine is also called the USB link layer(LINK). The PHY is basically a serializer-deserializer(SERDES), which also handles the low level USB protocol and the signaling task. The interface between the PHY and the SIE was defined as UTMI (USB2.0 Transceiver Macrocell Interface) specification[2], which enables the easier design of peripheral devices including the SIE and device specific logics.

In the design of the USB1.1 transceiver(the predecessor of USB2.0 transceiver), the data rates were low enough(1.5Mbps and 12Mbps) to allow all designs to be synthesized by using HDL such as Verilog. However, in the USB2.0 transceiver design, high-speed analog blocks are widely required and it is difficult to synthesize the design with cell libraries since the maximum operating frequency is increased to 480Mbps.

In this work, the physical layer USB 2.0 transceiver chip was designed and fabricated by using a 0.25um 1-poly 5-metal CMOS process. The chip was laid out with a full-custom method. The behavior-level simulation was performed in advance for the functional verification of this chip by using a Verilog code.

Section II describes the transceiver architecture used in this work. Section III describes the brief circuit description of each block. Section IV shows the measured results, and section V concludes this work.

II. ARCHITECTURE
Fig. 2 shows the architecture of the physical layer USB2.0 transceiver chip for USB device applications.

Fig. 2. Architecture of the physical layer USB2.0 transceiver chip
III. CIRCUIT DESCRIPTION

A. Analog Front End Block

Compared to the USB1.1 transceiver, the USB2.0 transceiver uses three additional blocks in the analog front-end[3] block, which are a HS(high speed) driver, a HS receiver, and a transmission envelope detector[3]. Separate output drivers and input receivers are used for 480Mbps and 12Mbps data signaling, respectively, as shown in Fig.2. A bias generation block supplies reference voltages to the analog front-end block. The CDR block consists of a HSCDR(High Speed Clock Data Recovery: 480Mbps) followed by an elastic buffer and a FSCDR(Full Speed Clock Data Recovery: 12Mbps). The HSCDR circuit generates the 30MHz clock for the SIE interface and the 480MHz clock for the shared digital logic block from a 12MHz external crystal oscillator output. All the data transfers between the transceiver and SIE are synchronized with the 30MHz clock. The HSCDR and the FSCDR are used to extract the timing information from the incoming data stream which is synchronized to a transmitter(USB host) clock, and the following elastic buffer (EB) compensates for differences in frequency between the transmitter(USB host) clock and the receiver(USB device) clock.

The shared digital logic block consists of the transmitting path, and the receiving path and the control circuitry, which are shared both for HS and FS operations. The transmitting path consists of a serializer, a bit stuffer and an NRZI encoder. The receiving path consists of an NRZI decoder, a bit unstuffer and a parallelizer. The bit stuffer inserts a zero after every six consecutive ones in the data stream to ensure adequate signal transitions in the NRZI-encoded transmitted data. Fig.3 shows a timing diagram of signals generated by the transmitting path.

As shown in Fig.1, the transceiver is connected to its link layer(SIE) via a 16-bit parallel interface and communicates with a host(PC or Hub) through a USB differential cable. The proposed transceiver has three major blocks; an analog front-end block, a CDR block, and a shared digital logic block. An analog front-end block consists of output drivers and input receivers, two line state detectors, and a transmission envelope detector[3]. Separate output drivers and input receivers are used for 480Mbps and 12Mbps data signaling, respectively, as shown in Fig.2. A bias generation block supplies reference voltages to the analog front-end block. The CDR block consists of a HSCDR(High Speed Clock Data Recovery: 480Mbps) followed by an elastic buffer and a FSCDR(Full Speed Clock Data Recovery: 12Mbps). The HSCDR circuit generates the 30MHz clock for the SIE interface and the 480MHz clock for the shared digital logic block from a 12MHz external crystal oscillator output. All the data transfers between the transceiver and SIE are synchronized with the 30MHz clock. The HSCDR and the FSCDR are used to extract the timing information from the incoming data stream which is synchronized to a transmitter(USB host) clock, and the following elastic buffer (EB) compensates for differences in frequency between the transmitter(USB host) clock and the receiver(USB device) clock.

The shared digital logic block consists of the transmitting path, and the receiving path and the control circuitry, which are shared both for HS and FS operations. The transmitting path consists of a serializer, a bit stuffer and an NRZI encoder. The receiving path consists of an NRZI decoder, a bit unstuffer and a parallelizer. The bit stuffer inserts a zero after every six consecutive ones in the data stream to ensure adequate signal transitions in the NRZI-encoded transmitted data. Fig.3 shows a timing diagram of signals generated by the transmitting path.

III. CIRCUIT DESCRIPTION

A. Analog Front End Block

Compared to the USB1.1 transceiver, the USB2.0 transceiver uses three additional blocks in the analog front-end[3] block, which are a HS(high speed) driver, a HS receiver, and a transmission envelope detector.

Fig.4(a) shows the HS driver circuit. It consists of a PMOS input differential pair which acts as a high-speed current steering switch. PMOS transistors are used in the differential pair since the voltages of output nodes(outp, outn) have small values between 0 and 400mV. The series combination of RS1 and M1 (also RS2 in series with M2) forms the on-die termination resistor, which is adjusted to be the same as the characteristic impedance(45Ω) of USB cable by the replica bias circuit. An external resistor(Rext) was used in the replica bias circuit[4]. Vref was adjusted to be 400mV, which sets the maximum output voltage level to be 400mV. RS1 and RS2 are made of un-salicided poly-silicon. The driving current(Io) is set to be 17.78mA by the Vref bias voltage and the Rext external resistor. Thus, The differential output voltage between output nodes (outp and outn) ranges between -400mV and +400mV when outp and outn are terminated with precision 45Ω resistors to ground at the receiver side.

Fig.4(b) shows the simulated eye patterns at the transmitter output nodes(outp and outn) and the receiving ends after passing through a 5 meter USB cable. The data rate is 480Mbps. A transmission line W-model was derived as shown in Table I for a 5 meter USB cable by using a network analyzer.

<table>
<thead>
<tr>
<th>W-model RLGC parameter set for a USB cable</th>
</tr>
</thead>
<tbody>
<tr>
<td>L0</td>
</tr>
<tr>
<td>172.125nH</td>
</tr>
</tbody>
</table>

Fig.5(a) shows the schematic of the transmission envelope detector, which is also called the squelch detector. The transmission envelope detector detects the HS idle state, when the magnitude of the differential input voltage received through a USB cable falls below the squelch threshold (125mV). The transmission envelope detector consists of level-shifter, a current-mode Schmitt trigger, a one-shot pulse generator, a RC low-pass filter, and a voltage-mode Schmitt trigger. A current-mode Schmitt
trigger with NMOS input differential pairs is used to enhance the operating speed. Level shifters, which are PMOS source followers, are used to raise the input voltage (0 to 400mV) to the NMOS input voltage level. One shot pulse generator generates a 1-ns pulse only at the input transition time points. The NMOS diode and the RC low-pass filter form an envelope detector. The voltage-mode Schmitt trigger enhances the noise immunity in determining the idle state from the envelope detector output. The four input signals are applied at the input nodes of level shifters. They are differential input voltages (inp and inn), the squelch level bias voltage V_{sq}(125mV), and the ground level (GND).

Fig. 5. (a) Transmission envelope detector (b) current-mode Schmitt trigger with squelch level control

B. Clock Data Recovery Block

The clock data recovery block consists of a HSCDR, a FSCDR and an elastic buffer as shown in Fig. 2. To extract the clock signal from the NRZI encoded input data, a burst mode CDR[6][7][8] was used for both HSCDR and FSCDR in this work to meet the UTMI specification that the synchronized clock must be generated within 4 bit times from the 32-bit SYNC pattern input data. The over-sampling CDR or the tracking-style CDR could not be used in this work since their tracking times were longer than several tens of bit times at the data rates of 480Mbps.

The elastic buffer (EB) works as a clock domain buffer as well as for jitter reduction for HS operation. The local 480MHz clock signal generated from a local 12MHz crystal oscillator is used in the shared digital logic block. However the incoming data is synchronized to the remote 480MHz clock of the host transmitter which is located at the opposite end of the USB cable. The frequency of the remote clock is slightly different from that of the local clock due to the mismatch of crystal oscillators. The UTMI specification requires the maximum relative frequency difference between local and remote clocks to be less than +/-500ppm. The number of error bits can reach up to +/- 12 bits, when this frequency error is accumulated during the time interval of one maximum-size packet. Therefore, a twenty-four-bit FIFO was used as a clock domain buffer in this work, where a token-ring type synchronizing FIFO was employed in this design[9].

The elastic buffer is also used to reduce the jitter components generated from the incoming data. Two 480MHz clock signals are used in the elastic buffer. One is the dirty 480MHz clock extracted from the incoming data. This dirty clock is used to read the incoming data into the elastic buffer at the rising edge time points. The other is the clean local 480MHz clock. This clean clock is used to output data from the elastic buffer into the shared digital logic block. Since the outgoing data from the elastic buffer have low jitter components since they are synchronized to the clean local clock.

C. Shared Digital Logic Block

The shared digital logic block performs serialization, bit stuffing, NRZI encoding and insertion of SYNC and EOP fields, during data transmission. Likewise, it performs de-serialization, bit un-stuffing, NRZI decoding and stripping of SYNC and EOP fields while receiving data[2].

The behavioral-level Verilog simulation was done in advance for functional verification. Then, the Verilog code was converted to transistor level and it was laid-out in a full-custom way. 4200-transistors were used in the shared digital logic block, which consumes 30mW at the HS operation.
IV. TEST RESULTS

The test chip was implemented by using a 0.25um 1-poly 5-metal CMOS technology. Fig.6 shows the fabricated chip layout and the chip microphotograph. The core chip area excluding IO pads was 0.91 x 1.82mm² and the power consumption was 225mW at HS mode and 150mW at FS mode. Test features such as digital block loop-back test, CDR test, and driver/receiver electrical test mode were implemented inside the chip.

The test configuration is shown in Fig.7. The transmitter and the receiver correspond to a host PC and a device circuit board respectively. Two test chips were connected by a 5m USB cable. PRBS data were transmitted through the USB cable and measured at both sides using a differential probe for electrical characteristic test. The protocol level test was successfully performed using a pattern generator and a logic analyzer. Also, the BERs were measured to be less than 1E-12 for both HS and FS modes by using a 1.4Gbps BER tester.

Fig.8(a) and (b) show the measured eye patterns of HS mode at the transmitter and the receiver sides respectively. The shaded window indicates the requirement by UTM specifications. The eye openings at transmitter and receiver sides were 310mV and 225mV respectively, both of which satisfy the UTM electrical specification. However, the full chip test was not performed due to a minor problem at clock input pin. The overall performances of this work were tabulated in TABLE II along with those from other works.

V. CONCLUSIONS

A USB2.0 PHY transceiver chip compatible to UTM specification was designed, fabricated and tested by using a 0.25um 1P5M CMOS process. The functionalities of all the components of the fabricated chip were verified successfully by using several test modes for both HS and FS modes. They include the analog front-end block(output drivers, receivers, envelope detector), the CDR block, and the shared digital logic block. The electrical tests on the fabricated chip showed the eye patterns compatible to the UTM specifications, and the BER less than 1e-12, for both HS and FS modes. The chip area excluding IO pads was 0.91 x 1.82mm². The power consumptions at the supply voltage of 2.5V were 225mW and 150mW for HS and FS modes, respectively.

ACKNOWLEDGMENTS

This work was supported in part by the System IC 2010 project, the IDEC(Design Education Center) and the National Research Laboratory Program.

REFERENCES

[3] Universal serial bus specification, Revision 2.0, Apr.27,2000

TABLE II

<table>
<thead>
<tr>
<th>CY68000</th>
<th>KLSKUSB101</th>
<th>SMSC GT3100</th>
</tr>
</thead>
<tbody>
<tr>
<td>process</td>
<td>0.25um CMOS</td>
<td>0.1um CMOS</td>
</tr>
<tr>
<td>supply</td>
<td>3.3V</td>
<td>3.3V</td>
</tr>
<tr>
<td>power</td>
<td>600mW</td>
<td>600mW</td>
</tr>
<tr>
<td>core area</td>
<td>296x200</td>
<td>296x200</td>
</tr>
<tr>
<td>package</td>
<td>64pin</td>
<td>64pin</td>
</tr>
<tr>
<td>power</td>
<td>630mW</td>
<td>630mW</td>
</tr>
<tr>
<td>current</td>
<td>15uA</td>
<td>15uA</td>
</tr>
</tbody>
</table>

external Xstal | 24MHz | 48MHz | 12MHz | 12MHz |