Energy Efficient Ethernet Encodings

Yanpei Chen, Tracy Xiaoxiao Wang, Randy H. Katz
Department of Electrical Engineering and Computer Science
University of California, Berkeley
Berkeley, CA, United States
ychen@berkeley.edu, tracyx@berkeley.edu, randy@eecs.berkeley.edu

Abstract— **The energy efficiency of network elements is becoming more prominent, with growing concern for Internet power consumption and heat dissipation in datacenters and communications closets. Previous work has looked at energy efficient wireless topologies, network nodes, routers, and protocols. In considering a fresh redesign of the Internet datacenter for energy efficiency, we believe that energy efficient encodings are worthy of study. In this work, we re-examine the choice of Ethernet encoding, develop an associated energy model, evaluate current encodings, and propose new encodings. We found that simpler encodings are more energy efficient, with power savings of around 20% for the best encoding. Our work represents a first step in re-examining the established assumptions and practices of the PHY level of the network stack with respect to energy.**

Keywords-Ethernet encoding; energy efficiency.

I. INTRODUCTION

Network energy efficiency is receiving increasing attention. Global warming, energy costs, and heat dissipation in datacenters and communication closets put power management at the forefront of network research. Studies in 2001 found that 2% of U.S. electricity consumption can be attributed to powering the entire Internet. This amounts to 74TWhr and $6 billion spent in 2001 [1, 2]. In contrast to the continuously increasing energy demand of the Internet, U.S. national electricity generation capacity has remained constant since 2005 [3]. Improving Internet energy efficiency will not only reduce the operating costs of Internet equipment, it will also bring tangible reductions to the national carbon footprint.

There is considerable work on Internet energy efficiency. The network stack has already been examined [4, 5, 6], except for link layer encodings. The dominant Internet link layer technology is Ethernet. Its encodings were traditionally considered difficult to change, and consequently, there is no systematic understanding of the problem space for energy efficiency in this area. Recent interest in "Greenfield" datacenter design means there is now an opportunity to consider innovative, energy efficient link layer encodings.

We suspected that there would be opportunities to save energy, since the widely implemented encodings were developed before energy concerns became important. It is now critical to quantify the possible energy savings through energy conscious encodings, compared with savings for other techniques. As a first step, any new encodings we propose must be compatible with existing technology, in that we should require no changes to higher layers of the network stack.

We focus on three encodings: 4B5B and MLT-3 for 100Mbps over UTP Ethernet cables, 8B10B for 1Gbps over optical fiber, and 4D-PAM5 for 1Gbps over UTP. These three encodings are the most widely deployed. Internet datacenters and high end networks are predominantly 1Gbps, and most residential networks remain 100Mbps. We do not consider 10Gbps because while it may present an even greater opportunity for energy savings, it is significantly less widely deployed. Nevertheless, we will show that the methodology and insights gained from a study at 100Mbps and 1Gbps become even more relevant for higher speed links.

To the best of our knowledge, our work is the first detailed study to re-consider the choice of Ethernet encoding with respect to energy efficiency. Our contribution is three-fold. First, we offer a new view of how to consider the impact of energy consumption on communications encodings. Second, we examine the energy consumption of Ethernet encodings for 100Mbps and 1Gbps technologies, and suggest an improved encoding for 100Mbps. Third, we offer a power model of Ethernet encodings to help guide further work in the area.

The paper is structured as follows. Section 2 quickly reviews existing Ethernet encodings and the large body of prior work in Internet power management and energy efficiency. Section 3 explains our view of the encoded communication problem and outlines our power model for Ethernet encodings. Section 4 describes our proposed encoding. Section 5 evaluates various encoding through Matlab and Verilog simulation, at the same time verifying our power model. Section 6 distills key insights from our work, and makes recommendations for future work in energy efficient Ethernet encodings and Internet energy efficiency in general. Our key result is that simpler encodings are more energy efficient, with savings of around 20% for the encoding.

II. BACKGROUND AND RELATED WORK

A. *Existing Ethernet Encodings*

We look at 4B5B and MLT-3 for 100Mbps over UTP cables, 8B10B for 1Gbps over optical fiber, and 4D-PAM5 for

This work is performed in the RadLab with funding from Google, Microsoft, and matching support from the UC Discovery program.

1Gbps over UTP. Full descriptions of the Ethernet encodings are found in the IEEE and ANSI standards [7, 8].

4B5B and MLT-3 encodings are used in 100BASE-TX for 100Mbps over UTP. Four-bit blocks of the input bit stream are mapped to five-bit output blocks to facilitate synchronization and other functions. The outgoing bit rate is 125Mbps, above the natural frequencies of the copper UTP cable. The MLT-3 encoding allows 125Mbps to be delivered at 31.25 MHz. MLT-3 output signals have three analog levels, with a peak-to-peak voltage of 2V. It delivers 125Mbps at 31.25 MHz because it cycles through +1, 0, and -1 logic levels, with a "1" bit in the input causing a logic level transition. There are no direct transitions between the +1 and -1 logic levels, allowing a low frequency signal to be used.

8B10B encoding is used in 1000BASE-LX/SX for 1Gbps over single/multi-mode optical fiber. Eight-bit blocks of the input bit stream are mapped to ten-bit blocks to facilitate clock synchronization. A running parity check ensures the output is DC balanced. The output bit stream gets sent over optical fiber using either on-off-keying or phase modulation.

The 4D-PAM5 (4-dimensions, 5 levels pulse amplitude modulation) encoding delivers 1Gbps over UTP. Eight-bit blocks of the incoming bit stream are converted to four PAM5 signals, with a peak-to-peak voltage of 2V and sent over four twisted pairs of the UTP cable. A complex scrambling scheme ensures output DC balance and facilitates full duplex on all four twisted pairs, with each pair being 250Mbps full duplex.

B. *Internet Energy Efficiency*

One of the earliest works in Internet energy efficiency is [9]. Many studies have followed. There is well established research in wireless energy efficiency, motivated by the limited power budgets for wireless devices, such as those in sensor nets and ad hoc networks [10, 11]. Studies have also looked at how channel conditions and wireless protocols affect power consumption [12]. In comparison, our work focuses on higher speed wired topologies.

Prior work has looked at power management at network nodes, such as network switches [4, 13], networked storage and disk drives [14], servers [15], and PCs [16]. Our work focuses on communication between network nodes.

Other power saving strategies focus on protocols at both the transport and network layers. There have been studies on the power consumptions of different flavors of TCP [5], TCP in wireless [17], sleep option for TCP [18], and using proxies to facilitate extensive sleep time [19]. Our work is focused on PHY and link encodings rather than new protocols.

Possible ways to save energy in the link layer includes reducing the link layer speed to facilitate energy savings during times of low traffic [6]. Current work in the IEEE 802.3 Energy Efficient Ethernet (EEE) Task Force includes diverse ideas on how to save energy in the link layer [20]. We complement the work there by offering energy efficient encodings that can be deployed in conjunction with other link layer power management techniques.

Past work in ADSL2+ power management hints at the idea of using line encodings to save energy [21]. There, it was suggested that different modulation schemes would deliver different transmission energy. We are not aware if the idea was pursued further.

Our work on energy efficient Ethernet encodings (EEEE) is an alternative approach to saving energy for the Internet. We differ from previous work in our focus to investigate and quantify the energy benefits of alternative PHY encodings. Our work complements existing research. One can envision a power efficient Internet in the future, with energy efficient protocols and energy efficient nodes, energy efficient wireless for wireless nodes, and optimized wired links with EEEE for encoding the data sent.

III. ENERGY CONSCIOUS ENCODINGS

A. *The Energy Conscious Communications*

The canonical digital communication problem is: given a certain bandwidth of the communication channel, an energy budget at the communication endpoints, and a target error rate, we try to maximize the data rate of communication. The bandwidth and energy budget are the resources available, the error rate is a performance bound, and the maximized data rate is the performance goal. Encoding is a tool to maximize performance using the resources available.

Traditionally, bandwidth and channel conditions have been the bottlenecks, affecting the data rate and the error rate respectively. Thus, the focus of encoding schemes has been to efficiently use the available bandwidth, and correct errors introduced by channel noise. Only in applications with limited power supply, such as mobile devices or sensor networks, has the energy budget been a concern. Now, with a rising focus on power consumption, we need an alternate view that highlights energy issues.

We formulate an alternative digital communication problem. Given a certain bandwidth, an error rate, and a data rate prescribed by Internet standards, we minimize the energy budget required. In this view, the bandwidth is the resource available, the data rate and error rate are bounds, and the minimized energy consumption is the goal.

A performance metric common to both views is the bandwidth-energy product, helpful for comparing different encodings, given the same data rate, error rate, and "all else equal". For an "efficient" encoding, the value of this product should be low. We could also look at the data rate to bandwidth-energy ratio, i.e., (data rate) / (bandwidth × energy), a re-cast of the "energy-per-bit" concept. This metric is not as helpful since the data rate is often prescribed, and different communications channels may preclude certain data rates.

For comparing the energy efficiency of two encodings, the only metric we need is the energy budget. The encodings need to have the same bandwidth, data rate, error rate, and other performance criteria. Otherwise, we cannot compare two encodings using energy efficiency alone.

B. Energy Model for Ethernet Encodings

The goal of an energy model is to understand how different parts of the communication system contribute to the energy consumed. For Ethernet encodings, we find it helpful to include only two sources of energy consumption – the encoding circuits, and the transmission energy put on the communication channel. The encoding circuits include digital encoding circuits, D/A converters, and pulse-shaping circuits. The transmission energy of the communication channel is either dissipated into the channel, or received at the destination.

There are, of course, other energy consumers in a communication system, including send/receive buffers, memory, and possibly the operating system. These sources are independent of the encodings used, and are left out of our energy model. A more fine-grained model should account for these energy consumers and quantify their contribution.

We propose that in wired Ethernet, the energy spent in encoding circuits is much larger than the transmission energy. This is a significant departure from wireless energy models, where the encoding circuit energy is routinely assumed to be negligible compared to the transmission energy, e.g., in [22, 23, 24]. The difference is due to two reasons. First, the wireless channel is noisy and three-dimensional. Therefore, large transmission energy is required for a useful signal to noise ratio (SNR). In contrast, wired Ethernet is less noisy, and the signal is concentrated along a one-dimensional cable. Therefore, the transmission energy need not be large. Second, the data rates prescribed for 100Mbps and 1Gbps greatly exceeds the natural frequencies for UTP cables, requiring non-trivial and sometimes very complex encodings to deliver the prescribed data rate. Complex encodings means larger circuits, and higher speeds means greater circuit energy consumption. Therefore, we believe the encoding energy to be larger than transmission energy in wired Ethernet, and the ratio would become even more acute as data rates increase to 10Gbps and beyond.

Verifying this energy model requires getting a quantitative measure of the encoding energy and transmission energy. The transmission energy is relatively easy to estimate. For 100Mbps and 1Gbps over UTP, we can estimate the transmission energy by the line voltage and the cable insertion loss. For 1Gbps over optical fiber, we can estimate the transmission energy by the optical output of laser diodes. This estimate would exclude the power consumed in the DC bias circuits necessary for operating the laser diode; the power for these circuits should also be excluded from the encoding energy.

Estimating the encoding energy is more difficult. The most direct verification would require probing the encoding circuits. We are not aware of commercial chip-sets that allow such probing. This approach may also run into difficulties in identifying and isolating the encoding circuits in a highly-optimized chip-set layout. Another approach would be to design and layout a communication chip-set from scratch, then simulate the power consumption using a CAD suite. This approach has the side benefit of generating a fine-grained power model for the chipset in a NIC card, but is probably best left to experts in circuit design. We use a third approach, which gives only a first-order estimate. We take the power consumption of typical NICs, and subtract from it the transmission power. The remainder would include the power consumption of encoding circuits and other circuits, such as buffers, or DC bias circuits in optical fiber NICs. In the absence of a more fine-grained method, this calculation gives a ball-park estimate of the encoding circuit power consumption.

In addition, we can compare the relative encoding circuit energy for different encodings by looking at the size of the different encoding circuits. Larger circuits generally mean greater energy consumption. In particular, we can build circuit simulations of different encodings using a uniform technology, and compare the size of the resulting circuits. Such simulations would not give us absolute values for energy consumption. They would, however, give us an idea on the relative energy consumptions for the circuits used in different encodings.

IV. An Alternate Encoding to MLT-3

We propose an alternative encoding to MLT-3 that has lower power consumption. Like MLT-3, it takes as input the result of 4B5B encoding, and outputs the voltages to be sent. It also delivers 125 Mbps at 31.25 MHz. Our encoding is a fully backwards compatible alternative to MLT-3, in that we require no changes to 4B5B and the Mac layer above, nor the physical medium below. In Section 5, we show that our encoding can save 18% of transmission energy, and 60% of encoding circuit energy.

Our encoding is inspired by the observation that while the 4B5B output bit stream is not DC-balanced, the output of 4B5B in conjunction with MLT-3 is. If we assign voltage levels based on sequences in the 4B5B output stream, we can get a larger fraction of symbols in a low energy state, at the cost of slight DC-imbalance. If we ensure our encoding conforms to the output level transition constraints of MLT-3 and signals at the same baud rate, then we can deliver 125 Mbps at 31.25 MHz for lower transmission energy.

The new encoding uses a state machine involving the past two bits in the 4B5B output stream. The state transition diagram is shown in Fig. 1. An incoming data bit causes a state transition. For example, if the past two bits are 01, the machine is in State 01. An incoming bit with value 0 will cause a transition to State 10. In State 10, the previous two bits are the 1 bit carried over from State 01, and the new 0 bit. Given a predetermined initial state, the output is uniquely decodable.

The states are mapped to three logical output levels, the same as the three logical levels in MLT-3. The mapping is shown in Fig. 1. Like MLT-3, our encoding has no direct transitions between +1 and -1 output levels. This allows 125 Mbps at 31.25 MHz. If we signal at the same baud rate and use the same peak-to-peak voltage, our encoding would be fully compatible with 4B5B and existing technology. We would require both endpoints on the link use either MLT-3 or our alternative, and endpoints could potentially support both kind of ports.

As we will detail later, our alternative is significantly simpler to implement compared with MLT-3, leading to lower encoding circuit energy.

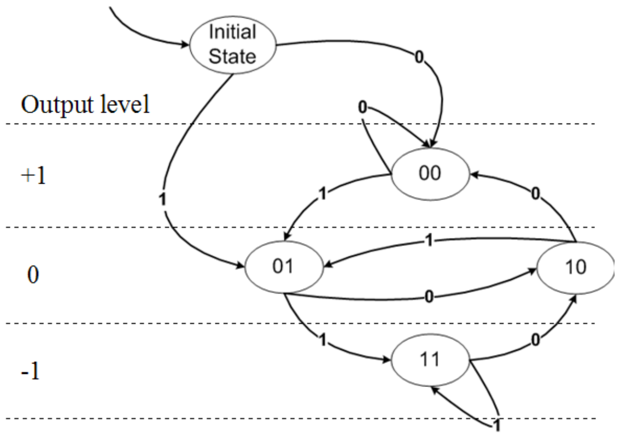

Figure 1. State transition diagram for our improved encoding.

A potential limitation is that our encoding is not DC balanced. This was formerly a significant issue. When encodings were first developed for the telephone network, the AC signal component would carry the voice, while the DC component would be used to power the telephone sets and the line repeaters. Any DC imbalance in the encoding would be treated as another DC power component, and would be unrecoverable at the receiver. Also, the DC component would not pass through any transformers used for impedance matching for the transmission line. Such concerns may still persist for DSL, but not for the vast majority of Ethernet links.

Another possible limitation is that our encoding is essentially amplitude modulation, where data is encoded in the voltage levels, in contrast to differential signaling in MLT-3, where data is encoded in the presence or absence of transitions. Differential signaling is preferred in noisy environments, because detecting a transition is easier than comparing against a threshold voltage. We believe this is not a critical issue, since wired Ethernet is a relatively non-noisy medium. Also, our encoding divides 2V peak-to-peak into three levels, with the difference between the voltage amplitude thresholds being smaller than that for 1000BASE-TX, with PAM5 dividing 2V peak-to-peak into five levels.

V. Evaluation

We built simulations of the encodings in Matlab and Verilog to evaluate their performance. The former allow us to analyze the statistical distribution of output voltage levels, and identify any obvious opportunities to save on transmission energy. The later give us an FPGA implementation of the encoding circuits, allowing us to compare the encoding circuit size and encoding circuit energy. The simulations take in a random bit stream supplied by the MAC layer and output a symbol stream sent to the media D/A converter.

For our Matlab simulations, we simulate for 100BASE-TX the Physical Coding Sublayer (PCS) and the Physical Medium Dependent (PMD) sublayer, containing the 4B5B and MLT-3 encodings. For 1000BASE-LX/SX and 1000BASE-T, our Matlab simulations include the PCS only, with the 8B10B encoding for 1000BASE-LX/SX, and the 4D-PAM5 encoding

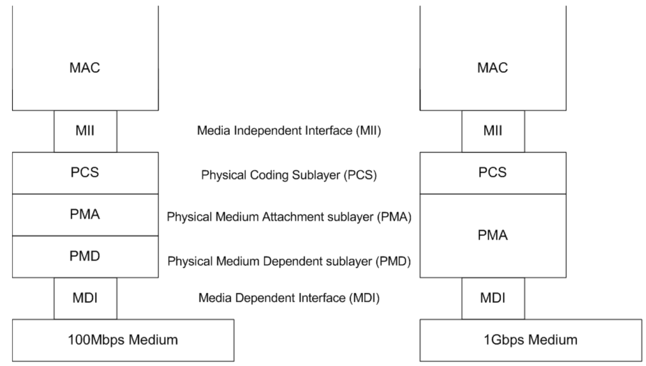

Figure 2. Simulation Layers.

for 1000BASE-T. The output voltages from our simulation correspond to the inputs to the Media Dependent Interface (MDI), which is immediately converted to an analog signal and sent on the physical medium.

For our Verilog simulations, we also include the Physical Medium Attachment sublayer (PMA), which serializes the output bit blocks from the PCS. The PMA also performs some other functions that were not simulated, such as generating control signals and performing synchronization. We do not count the PMA towards our encoding circuit size, since it is not a part of the encoding per se.

Fig. 2 shows the relationship between different sublayers included in our simulations.

For transmission energy, encodings with a greater fraction of symbols in lower voltage levels would consume less transmission energy. For encoding circuit energy, encodings with smaller circuits, indicated by fewer logical units required, would consume less circuit energy. Encodings with lower energy are more preferable.

A. Simulation Results – Transmission Energy

We used Matlab simulations to obtain the output characteristics of the encodings and analyze the output transmission energy. Table 1 shows the Matlab simulation results. For each encoding, we compute the fraction of output symbols at each logic voltage level. To simplify analysis, we have lumped together logic levels of the same magnitude, e.g. for 4B5B and MLT-3 encoding, the +1 and -1 logic levels together account for 50% of the output symbols. The peak-to-peak analog voltage is 2V for all encodings on UTP.

For 100Mbps, logic levels ±1 correspond to analog levels ±1V. For 1Gbps, logic levels ±1 are ±0.5V in analog, and logic levels ±2 are ±1V in analog. For 8B10B on optical fiber, the logic level is translated to either on-off-keying of the laser diode, or a phase modulated optical signal.

For 4D-PAM5, we see from Table 1 that it has a large fraction of symbols in logic level 1. This means the 4D-PAM5 is somewhat optimized, in that more symbols are in logic level 1 than in logic level 2. However, there is likely to be opportunities to further optimize the code, since there are more

TABLE I. MATLAB SIMULATION RESULTS

Encoding	Fraction of output symbols in each logical level		
	0	*±1*	*±2*
4D-PAM5	0.22	0.44	0.34
8B10B	0.5	0.5	-
4B5B MLT-3	0.5	0.5	-
4B5B and alternative to MLT3	0.59	0.41	-

symbols in logic level 1 than in logic level 0. Given the complexity and functional requirements of 4D-PAM5, it is not immediately obvious what are the possible optimizations. Any efforts must consider that we are optimizing the transmission energy. Since this is small compared with encoding circuit energy, "optimizations" that result in a more complex code is likely to lead to larger circuits and an undesired increase in energy consumption.

For 8B10B encoding, the output logic levels are mapped to on-off-keying or phase modulation optical signals. The transmission energy in optical fiber is given by the optical power used to drive the fiber, and not the logic levels sent. Hence 8B10B has no need to be further optimized for transmission on optical fiber. We will mention that for phase modulation, the optical signal is always "on", with logical high and low distinguished by a different optical phase. This means that for the same peak optical power, phase modulation spends two times the transmission energy of on-off-keying. However, the energy is not wasted, since phase modulation will result in a 3dB gain in the optical signal to noise ratio. From a circuit energy perspective, phase modulation circuitry is more complex than circuits for on-off-keying. Hence it is preferential to send 8B10B using on-off-keying, but at twice the transmitted optical power to achieve the same signal to optical noise ratio as phase modulation.

For 4B5B and MLT-3 encoding, we see that half the symbols are not energized. However, our improved alternative to MLT-3 allows over half of the symbols to be in a non-energized state. It is inspired by DC-imbalance in the 4B5B output. Our improved encoding takes advantage of this fact to assign over half of the symbols to the non-energized state. However 9% of the output symbols are assigned to logic level 1, while 22% of the symbols are assigned to logic level -1, leading to a slight DC-imbalance. On the other hand, based on simulation results, our improved encoding spends 9% less time in logic levels ±1 compared with MLT-3, leading to an 18% saving in transmission energy.

To repeat our transmission energy saving calculations, we use the results in Table 2. This shows the statistical characteristics of the 4B5B output bit stream, when a random bit stream is used as the input. The 4B5B output is then fed into either MLT-3 or our improved encoding. As shown in the table, when we encode a random bit stream into 4B5B, 61% of the output bits are "1". Also, 9% of the output two-bit sequences are "00" and 32% of the output two-bit sequences are "11". Thus, for our encoding, the state machine would spend 9% of the time in State "00" and 32% of the time in State "11". Fig. 1 shows that these two states respectively correspond to analog voltage levels +1 and -1. Thus, our improved encoding spends 9% + 32% = 41% of the time in an energized state, compared with 50% for MLT-3, leading to a (41% – 50%) / 50% = 18% saving in transmission energy.

TABLE II. STATISICAL ANALYSIS FOR 4B5B OUTPUT SEQUENCES

Output Sequence	Fraction of all sequences
0	0.3875
1	0.6125
Total	1.0
00	0.0937
01	0.2938
10	0.2938
11	0.3186
Total	1.0

As we will show below, our improved encoding is also significantly easier to implement, leading to a reduction in the circuit encoding energy also.

B. Simulation Complexity – Encoding Circuit Energy

We used Verilog simulations to get an idea of the circuit implementation complexity of the encodings. This would give us an indication of the relative encoding circuit energy consumption. Larger circuits and more complex code would lead to greater encoding circuit energy consumption.

We built our Verilog simulations with the Xilinx FPGA design suite. Our block designs follow the general block layout found in [25]. We used ModelSim to verify the correctness of our simulations. We ran the Xilinx synthesis tool to synthesize the design for FPGA implementation, and we looked at the synthesis report to extract the circuit size in terms registers and logical look-up tables (LUTs) used. Table 3 shows a summary of our simulated encoding circuit size.

We can break down each encoding circuit into two parts – an asynchronous translation table for mapping input symbols to output symbols, and a synchronous state machine for cycling through voltages or ensuring output DC balance.

Compared with the state machine for MLT-3, the state machine for our improved alternative encoding uses only a third of the registers and LUTs. This is because the optimized MLT-3 state machine requires four states, whereas our improved encoding requires only two states. Fig. 3 shows the optimized state machines. Our alternative to MLT-3 is the equivalent of the state machine in Fig. 1, with the state transitions given by *input* / *output*. The MLT-3 state machine has transitions driven by *input* and states marked with *state* / *output*. Our encoding is clearly superior in terms of simplicity. Using a good first-order approximation that circuit power is proportional to circuit area, we find that our encoding uses only a third of the encoding circuit power of MLT-3.

TABLE III. VERILOG SIMULATION ENCODING CIRCUIT SIZE

Simulation Block	Registers Used	LUTs Used
4B5B translation table	0	4
MLT-3 state machine	4	6
Alt. to MLT-3 state machine	1	2
4B5B & MLT-3 total	0 + 4 = 4	4 + 6 = 10
4B5B & alt. to MLT-3 total	0 + 1 = 1	4 + 2 = 6
8B10B translation table	0	10
8B10B state machine	1	3
8B10B total	0 + 1 = 1	10 + 3 = 13
4D-PAM5 translation table	0	46
4D-PAM5 state machine	38	28
4D-PAM5 total	0 + 38 = 38	46 + 28 = 74

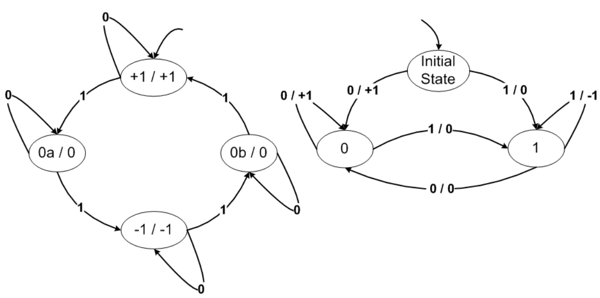

Figure 3. State machines for MLT-3 (left) and our alternative (right).

Also, the 8B10B state machine is as simple as that for our improved alternative to MLT-3, while the 4D-PAM5 state machine is an order of magnitude more complex. The 8B10B uses one memory variable of one bit to keep track of the running disparity in the output, similar to the one memory variable of one bit in our alternative to MLT-3. In comparison, the state machine for 4D-PAM5 has a 32 bit scrambler, with an exponentially larger state space.

Consequently, the complexity of encoding state machines is determined by the size of the state space, given by the number of memory variables required and the bits for each variable.

The translation table sizes offer another critical insight. For 8B10B, the translation table has 256 rows of 10 bits each compared with 16 rows of 5 bits for 4B5B. By counting bits, we would expect the 8B10B translation table to be 32 times larger than that for 4B5B, instead of 2.5 times as large. 8B10B breaks down its input-output translation into a 5B6B translation table and a 3B4B translation table. There are two output translations for each input, with the state machine keeping track of a running disparity that determines which of the two translations would be used. Thus the 8B10B translation tables would have 32 lines of 12 bits, and 8 lines of 8 bits, for 5.5 times the number of bits. Automatic optimization in Xilinx tools reduces the translation table even further, and it is 2.5 times as large as that for 4B5B.

For 4D-PAM5, the translation table has 256 rows of 12 bits each, where each row is a 4-tuple of 3-bit PAM5 symbols. Counting bits give a translation table 38.4 times larger than that for 4B5B. Instead, the table is 46 times as large, even with automatic optimization. We believe this is the overhead of having large tables, where additional circuitry is required to facilitate lookup under the same timing constraints.

Thus, we believe that encodings should avoid large translation tables whenever possible. One way to reduce the complexity of input-to-output bit translations is to adopt the strategy used by 8B10B, dividing the input bits into groups, and translating each group separately.

In short, our Verilog simulations show that to reduce encoding circuit complexity, we need to keep the state space small, avoid large translation tables, and have potentially many small translation tables instead.

C. *Verifying the Energy Model*

We seek to verify our proposed energy model for Ethernet encodings, i.e., that encoding circuit energy is much larger than transmission energy. We calculate the transmission power from line voltage and insertion loss, and compare it against the total power of NIC cards to get a rough estimate of encoding circuit power.

We calculate the total transmission power in several steps. We take the RMS transmission voltage and DC resistance to find the RMS current, assuming sinusoidal waveforms. The power loss is given by $I^2_{RMS}R_{DC}$. The insertion loss for the cable tells us what fraction of the total transmission power is lost. We divide the power loss by this fraction to get the total transmission power. Fig. 4 illustrates our calculations.

The peak voltage for 100Mbps and 1Gbps over UTP is 1V. We take from technical specs for off-the-shelf Cat5e UTP Ethernet cables typical values for DC resistance and insertion

$$V_{peak} = \text{Peak AC voltage on the cable}$$

$$V_{RMS} = \text{VDC equivalent of VAC} = \frac{V_{peak}}{\sqrt{2}} \text{ for sinusoids}$$

$$I_{RMS} = \text{DC equivalent current} = \frac{V_{RMS}}{R_{DC}}$$

$$R_{DC} = \text{DC resistance of the cable}$$

$$P_{lost} = \text{Transmission power lost in the cable} = I^2_{RMS}R_{DC}$$

$$P_{transmit} = \text{Total transmission power}$$

$$i = \text{Insertion loss in dB} = -10\log_{10}\left(\frac{P_{transmit} - P_{lost}}{P_{transmit}}\right)$$

$$P_{transmit} = \frac{P_{lost}}{1 - 10^{-\frac{1}{10}i}}$$

Figure 4. Calculating Transmission Power

loss. Substituting typical DC resistance of 9 Ohms at 100 meters and insertion loss of 10dB at 31.25 MHz for 100Mbps, and larger for 1Gbps, we find the transmission power to be approximately 0.062 W.

For encoding circuit power, we estimate by subtracting the transmission power from the total power for NICs. We obtain the power consumption of NICs from the technical specs for some off-the-shelf 1Gbps Ethernet NICs [26, 27, 28].

NIC power consumption ranges from 3.3W to 5W. Average power is typically lower than stated peak power rating. Even if we make a conservative assumption that average power is 10% of peak power, we find the transmission power is 0.062W / (10% × 3.3W) = 19% of the total power consumption, and encoding circuit power is the remaining 81%.

This calculation confirms our encoding energy model, that most of the energy is consumed in the encoding circuits rather than in data transmission.

Our energy model is indirectly verified by an independent study [29]. Recall the results from Section 5.B. indicate that 4D-PAM5 is almost an order of magnitude more complex than 4B5B and MLT-3. Assuming circuit power consumption is directly related to circuit size, our observation suggests that 4D-PAM5 consumes almost an order of magnitude more energy than 4B5B and MLT-3. In [29], using a completely different method, the NIC power is shown to indeed grow exponentially as link speed increases from 100Mbps to 1Gbps and to 10Gbps. Hence, we believe in our assumption that encoding circuit power is correlated with NIC power. At the same time, we are acutely aware that NIC power is more than encoding power, since it also contains contributions from buffers, OS interfaces, Wake-on-LAN, and other functions.

We should also mention that even though the available NIC specs offer the same data rate, their power consumption is considerably different. The NIC from 2001 consumed 5 W [27]. A 2004 model consumed 4.5 W [28]. A product from 2006 consumes only 3.3 W [26]. We believe that these numbers indicate a growing energy consciousness in the circuit design community, or a different, market-driven understanding of circuit energy consumption. Circuits performing the same encodings can save considerable power. We are encouraged by these results, because our energy model suggests that there is room for energy savings both in simpler code for simpler circuits, and more energy efficient circuits that consumes less power for a given encoding.

For Ethernet over optical fiber, the calculations are more straight-forward. The transmission power is simply the optical output power of the laser diode, typically 20-40mW, the same order of magnitude as the transmission power for Ethernet over UTP. The power consumption for optical fiber NICs is near identical to counterpart UTP NICs in the same product family, as suggested by [28]. Hence we can extend the above discussion to say that for optical fiber, the encoding circuit energy is also much larger than the transmission energy.

VI. Future Work and Conclusion

A. Key Insights

We developed several key insights regarding energy efficient Ethernet encodings.

We can look at the canonical encoded communication problem from a perspective that prioritizes energy. The goal of encoding is to deliver the prescribed data rate with the least energy, rather than to deliver the maximum data rate using a given energy budget. We believe this energy conscious perspective is helpful in designing future encodings.

For wired Ethernet, the encoding circuit energy is much larger than the transmission energy. This reverses the energy model for wireless encodings, where the encoding circuit energy is negligible, and the transmission energy dominates.

In the near future, the encoding circuit energy is likely to take an even larger share of the total energy for encodings, with lower transmission voltage over UTP and more complex encodings for 10Gbps and beyond.

Our proposed alternative to MLT-3 and our simulations shows that existing Ethernet encodings may not be energy efficient. In particular, we can reduce transmission energy by devising encodings for which a large fraction of encoded symbols have low energy.

Also, given that encoding circuit energy is much larger than transmission energy, we can get more energy savings by using simpler code. Simpler code means small translation tables, and state machines with a smaller state space. Our recommendation for simpler code is mirrored in suggestions for energy conscious wireless applications, where on-off-keying is also championed as the preferred encoding scheme [30].

B. Future Work

We make several recommendations for future work in energy efficient Ethernet encodings, in energy conscious encoding in general, and in energy efficient Internet.

For the first of these, a more rigorous method is required for verifying or measuring encoding circuit energy. Our approximations, though reasonable, is not sufficiently fine grained. Also, the energy efficiency evaluations here should be extended to new encodings for 10Gbps Ethernet and encodings being developed for even higher speeds. Furthermore, we should extend the simulations in Section 5.B to also include the decoder; for some encodings, the encoder and decoder may be highly asymmetric in hardware complexity. Last but not least, any new encodings should ideally be backwards compatible, although Greenfield datacenters give opportunities for the entire network stack to be redesigned.

For work on encodings in wired, energy conscious applications in general, we believe our alternative view on the encoded communication problem would be helpful. Our approach to understand the relative energy consumptions between encoding circuit power and transmission power should also be helpful.

Energy efficient Ethernet encodings is only one of many orthogonal methods to improve energy efficiency for the Internet. We hope our work invites a re-examination of the established assumptions and practices of the network stack with respect to energy.

C. Closing

The energy efficient Ethernet is a new study to reduce Internet energy consumption. We investigated the power consumption in link layer encodings. We evaluated existing encodings and proposed a new energy efficient encoding. Our study showed that simpler encoding is better, and encodings can be made more power efficient by being energy conscious.

ACKNOWLEDGMENT

We sincerely thank the following people for their help, suggestions, and pointers to related work: David Tse, Wael Diab, Jean Walrand, Bruce Nordman, and Martin Graham. We are especially grateful to Ken Christensen, Andrea Baldini, and Claudio DeSanti for their invaluable feedback on an early draft of this paper.

REFERENCES

[1] K. Kawamoto, J. Koomey, B. Nordman, R. Brown, M. Piette, M. Ting, A. Meier, "Electricity used by office equipment and network equipment in the US: Detailed report and appendices," Technical Report LBNL-45917, Lawrence Berkeley National Laboratory, February 2001.

[2] C. Gunaratne, K. Christensen, B. Nordman, "Managing Energy Consumption Costs in Desktop PCs and LAN Switches with Proxying, Split TCP Connections, and Scaling of Link Speed," Intl. J. of Network Management, vol. 15, pp. 297-310, September 2005.

[3] Energy Information Administration, "Annual Energy Review 2006," U.S. Department of Energy, June 2007.

[4] M. Gupta, S. Grover, and S. Singh, "A Feasibility Study for Power Management in LAN Switches," Proc. of the IEEE Intl. Conf. on Network Protocols, pp. 361-371, October 2004.

[5] H. Singh and S. Singh, "Energy consumption of TCP reno, newreno, and sack in multi-hop wireless networks," ACM SIGMETRICS Performance Evaluation Review, vol. 30, pp. 201-206, June 2002.

[6] C. Gunaratne and K. Christensen, "Ethernet Adaptive Link Rate: System Design and Performance Evaluation," Proc. of the 31st IEEE Conf. on Local Computer Networks, pp. 28-35, November 2006.

[7] IEEE LAN/MAN Standards Committee, "IEEE Std 802.3-2005 Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications," IEEE, December 2005.

[8] National Standards Institute, "Fibre Distributed Data Interface (FDDI) - Token Ring Twisted Pair Physical Layer Medium Dependent (TP-PMD) (formerly ANSI X3.263-1995 (R2000))," ANSI, 2000.

[9] M. Gupta, S. Sigh, "Greening of the Internet," Proc. Of ACM SIGCOMM, pp.19-26, August 2003.

[10] L. Feeny and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," Proc. of the 20th INFOCOM Annual Joint Conf. of the IEEE Computer and Communications Societies, vol. 3, pp.1548-1557, April 2001.

[11] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Mobile Networking for Smart Dust," Proc. Of 5th Intl. Conf. on Mobile Computing and Networking, pp.271-278, August 1999.

[12] M. Zorzi, M. Rossi, and G. Mazzini, "Throughput and energy performance of TCP on a wideband CDMA air interface," J. of Wireless Communications and Mobile Computing, vol. , pp. 71-84, December 2002.

[13] "Green Computing and D-Link," white paper, D-Link, Sepetember 2007, ftp://ftp10.dlink.com/pdfs/products/D-Link_Green_whitepaper.pdf, last accessed April 2008.

[14] S. Gurumurthi, A. Sivasubramaniam, and V. Natarajan, "Disk Drive Roadmap from the Thermal Perspective: A Case for Dynamic Thermal Management," Proc. of the 32nd Intl. Symp. on Computer Architecture, pp. 38-49, June 2005.

[15] P. Ranganathan, P. Leech, D. Irwin, and J. Chase, "Ensemble-level power management for dense blade servers," Proc. of the 33rd Intl. Symp. on Computer Architecture, vol. 34, pp. 66–77, October 2006.

[16] "Magic Packet Technology," White Paper, Advanced Micro Devices, November 1995, http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/20213.pdf, last accessed April 2008.

[17] B. Wang and S. Singh, "Computational Energy Cost of TCP," Proc. of the 23rd Annual Joint Conf. of the IEEE Computer and Communications Societies, vol. 2, pp. 785-795, March 2004.

[18] L. Irish, and K. Christensen, "A 'Green TCP/IP' to reduce electricity consumed by computers," Proc. of IEEE Southeastcon 1998, pp. 302–305, April 1998.

[19] K. Christensen, "The next frontier for communications networks: Power management," Proc. of SPIE – Performance and Control of Next-Generation Communications Networks, vol. 5244, pp. 1–4, August 2003.

[20] "802.3az Energy Efficient Ethernet Task Force," meeting materials, IEEE, January 2008, http://www.ieee802.org/3/az/public/jan08/, last accessed April 2008.

[21] "Lower-Power Modes for ADSL2 and ADSL2+," white paper, Texas Instruments, January 2005, http://focus.ti.com/pdfs/bcg/adsl2_low _power_wp.pdf, last accessed April 2008.

[22] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," IEEE Trans. Wireless Communications, vol. 4, pp. 2349-2360, September 2005.

[23] J. Kleinschmidt, W. Borelli, and M. Pellenz, "Power Efficient Error Control for Bluetooth-based Sensor Networks," Proc. of the 30th IEEE Conf. on Local Computer Networks, pp. 284-293, November 2005.

[24] J. Sanchez, P. Ruiz, "LEMA: Localized Energy-Efficient Multicast Algorithm based on Geographic Routing," Proc. of the 31st IEEE Conf. on Local Computer Networks, pp. 3-12, November 2006.

[25] "Implementing FDDI Over Copper, The ANSI X3T9.5 Standard," application note, Advanced Micro Devices, 1993, http://www.amd.com/files/connectivitysolutions/networking/archivednetworking/18258.pdf, last accessed April 2008.

[26] "Intel PRO/1000 PT Desktop Adapter," tech. spec., Intel Corporation, 2006, http://www.intel.com/network/connectivity/products/prodbrf/pro1000 _pt_desktop_adapter.pdf, last accessed April 2008.

[27] "Instant Gigabit Network Adapter," tech. spec., Linksys, 2001, http://www.kodeks.hr/linksys/Assets/Pdf/linksys_eg1032.pdf, last accessed April 2008.

[28] "HP 3X-DEGXA-XX PCI-X/PCI Gigabit Ethernet Network Interface Controllers," tech. spec. Hewlett-Packard Company, 2004, http://h18002.www1.hp.com/alphaserver/download/ek-degxa-in-b01-web.pdf, last accessed April 2008.

[29] Bruce Nordman, "Energy, Networks, Electronics, Data Centers," Technical Presentation, Lawrence Berkeley National Laboratory, October 2007, unpublished.

[30] J. Ammer and J. Rabaey, "The Energy-per-Useful-Bit Metric for Evaluating and Optimizing Sensor Network Physical Layers," Proc. of the IEEE Intl. Workshop on Wireless Ad Hoc & Sensor Networks, vol. 2, pp. 695-700, September 2006.