

EPISTEMEUS

ESSAYS

VOLUME 1 ; ISSUE 2 01.08.2026

Adder 4 Bit

Hosea Pratama Sinaga

1 Hosea Pratama Sinaga

EPISTEMEUS E S SAYS

Volume 1 ; Issue 1

Adder 4 Bit Hosea Pratama Sinaga

ABSTRAK

omputer, kalkulator, telepon pintar, dan berbagai perangkat otomatis K mampu melakukan perhitungan karena tersusun atas rangkaian yang membaca dua keadaan, yaitu 0 dan 1. Salah satu rangkaian dasarnya adalah adder 4 bit, sebuah rangkaian yang menjumlahkan dua bilangan biner empat bit serta menghasilkan empat bit keluaran dan satu carry out. Esai ini membahas prinsip kerja adder 4 bit melalui konsep bilangan biner, gerbang logika, half adder, full adder, carry in, carry out, LSB, dan MSB. IC 74LS83 digunakan sebagai contoh komponen yang mengolah penjumlahan biner dalam satu rangkaian terpadu. Materi disusun untuk siswa SD, SMP, dan SMA dengan tingkat pembahasan yang disesuaikan. Siswa SD dikenalkan melalui pola lampu mati dan menyala, siswa SMP mempelajari penjumlahan biner, sedangkan siswa SMA mengkaji hubungan gerbang logika dengan keluaran rangkaian. Sakelar, LED, kartu biner, dan simulasi digital dapat mengubah konsep abstrak menjadi pengalaman belajar yang mudah diamati. Adder 4 bit menjadi pintu masuk untuk memahami cara komputer berhitung sekaligus menumbuhkan ketertarikan terhadap elektronika digital sejak usia sekolah. Materi praktikum sendiri menempatkan pemahaman operasi penjumlahan, bit carry, dan penggunaan rangkaian adder sebagai tujuan utama pembelajaran. Kata kunci: elektronika digital, adder 4 bit, bilangan biner, full adder, IC 74LS83, media pembelajaran. Pendahuluan Satu sentuhan pada kalkulator dapat menghasilkan jawaban sebelum manusia sempat menyelesaikan hitungan di atas kertas. Telepon pintar membaca sidik jari, kendaraan mengenali jarak, mesin permainan memproses gerak Hosea Pratama Sinaga 2 an pemain, dan komputer menjalankan jutaan

perintah dalam waktu sangat singkat. Apa yang sebenarnya terjadi di dalam perangkat tersebut saat sebuah angka dimasukkan? Seluruh proses itu berawal dari kemampuan rangkaian digital membedakan dua keadaan yang ditulis sebagai 0 dan 1. Kedua angka tersebut mewakili keadaan listrik yang berbeda, lalu diolah melalui gerbang logika untuk membentuk operasi yang lebih besar. Susunan gerbang dapat digunakan untuk memilih data, menyimpan informasi, membandingkan nilai, dan melakukan perhitungan. Cara kerja tersebut menjadi fondasi perangkat digital yang dipakai sehari-hari. Pengetahuan tentang elektronika digital memberi kesempatan kepada siswa untuk melihat teknologi sebagai sistem yang dapat dipelajari dari bagian paling kecil. Bilangan biner, gerbang logika, dan rangkaian aritmetika menjadi pintu awal untuk memahami hubungan antara sinyal listrik dan kemampuan komputer mengolah informasi (Harris & Harris, 2012; Mano & Ciletti, 2017). Penjumlahan merupakan operasi yang mudah dikenali karena sudah dipelajari sejak sekolah dasar. Tantangannya mulai terasa ketika angka yang digunakan hanya terdiri dari 0 dan 1. Apa yang terjadi saat komputer menghitung 1 ditambah 1? Dalam bilangan biner, hasilnya ditulis sebagai 10. Angka 0 ditempatkan sebagai hasil pada kolom yang sedang dihitung, sedangkan angka 1 dipindahkan ke kolom berikutnya sebagai carry atau angka simpanan. Proses sederhana tersebut menjadi dasar rangkaian adder. Adder 4 bit menerima dua bilangan biner yang masing-masing tersusun dari empat bit, lalu menghasilkan empat bit penjumlahan dan satu keluaran carry. Kapasitas empat bit dapat mewakili bilangan desimal dari 0 sampai 15. Hasil yang melewati batas tersebut diteruskan melalui carry out. Materi praktikum yang menjadi dasar esai ini memakai IC 74LS83 serta menempatkan pemahaman penjumlahan aritmetika, bit carry, dan penggunaan rangkaian adder sebagai sasaran pembelajaran. Keluarga IC tersebut dirancang untuk menjumlahkan dua kata biner empat bit dan menyediakan keluaran hasil pada setiap bit beserta carry dari bit terakhir (Modul Praktikum Elektronika Digital, n.d.; Texas Instruments, 2001). Adder 4 bit memperlihatkan jalur yang menghubungkan pelajaran matematika dengan kerja perangkat keras. Angka yang semula ditulis di buku berubah menjadi sinyal, sinyal memasuki gerbang logika, lalu keluar sebagai hasil perhitungan. Mengapa rangkaian yang terlihat sederhana dapat menjadi bagian dari komputer yang sangat rumit? Komputer dibangun melalui susunan bertingkat. Gerbang logika membentuk rangkaian aritmetika, rangkaian aritmetika bergabung menjadi unit pengolah, lalu unit tersebut bekerja bersama memori dan sistem kendali. Siswa tidak perlu membongkar seluruh prosesor untuk memahami gagasan tersebut. Pengamatan terhadap satu rangkaian penjumlah sudah cukup untuk memperlihatkan bagaimana masalah besar dipecah menjadi langkah-langkah kecil yang memiliki aturan jelas. Pola 3 Hosea Pratama Sinaga berpikir semacam ini dekat dengan computational thinking, yaitu cara menyusun persoalan agar dapat diproses oleh manu-

sia atau mesin. Pembelajaran yang bergerak dari gerbang dasar menuju sistem komputer juga memberi gambaran utuh mengenai hubungan antara perangkat keras, perangkat lunak, dan informasi yang diolahnya (Wing, 2006; Nisan & Schocken, 2005). Pengenalan elektronika digital dapat dimulai sejak sekolah dasar selama bahasa, media, dan tingkat kesulitannya disesuaikan. Siapa yang harus mempelajari adder 4 bit, apakah hanya siswa jurusan teknik? Anak SD dapat mengenal 0 dan 1 melalui lampu mati, lampu menyala, kartu angka, serta permainan sakelar. Siswa SMP dapat mulai membaca bilangan biner dan mencoba penjumlahan sederhana. Siswa SMA dapat menelusuri fungsi gerbang XOR, AND, half adder, full adder, carry in, carry out, LSB, dan MSB. Satu rangkaian yang sama dapat dibaca melalui beberapa tingkat pembahasan tanpa kehilangan konsep utamanya. Standar pembelajaran ilmu komputer K-12 juga menempatkan konsep komputasi sebagai materi yang dapat dikenalkan sejak tingkat dasar, lalu dikembangkan secara bertahap menuju pembuatan dan pengujian sistem. Pengalaman dari media pemrograman visual memperlihatkan bahwa ketertarikan anak terhadap teknologi tumbuh saat mereka diberi ruang untuk membuat, mencoba, mengubah, dan melihat hasil pekerjaannya sendiri. Pola tersebut dapat diterapkan pada pembelajaran adder melalui LED dan sakelar yang memberikan keluaran langsung (Computer Science Teachers Association, 2017; Resnick et al., 2009). Rangkaian yang hanya dijelaskan melalui simbol dapat terasa jauh dari pengalaman siswa. LED yang menyala setelah sakelar ditekan memberi bentuk nyata pada angka biner. Bagaimana cara membuat siswa tetap penasaran saat tabel kebenaran mulai dipelajari? Guru dapat meminta mereka menebak hasil keluaran sebelum rangkaian diaktifkan, menyusun kelompok sebagai gerbang logika manusia, atau berlomba mengubah bilangan desimal menjadi biner. Kesalahan tidak berhenti sebagai jawaban yang dicoret karena siswa dapat mengikuti jalur sinyal dan mencari bagian yang keliru. Kegiatan semacam ini mengajak siswa mengamati, memperkirakan, menguji, dan memperbaiki pemahamannya. Kajian pembelajaran aktif dalam bidang sains, teknologi, rekayasa, dan matematika mencatat peningkatan hasil belajar saat peserta didik terlibat dalam aktivitas dan diskusi selama proses pembelajaran. Tinjauan dalam pendidikan teknik juga menemukan dukungan yang kuat terhadap aktivitas yang menempatkan siswa sebagai pelaku dalam proses belajar. Kedua kajian tersebut berfokus terutama pada pendidikan tinggi. Esai ini mengambil gagasan keterlibatan aktifnya untuk merancang pengenalan adder yang sesuai bagi siswa sekolah (Freeman et al., 2014; Prince, 2004). Esai berjudul Adder 4 Bit membahas rangkaian penjumlah sebagai sarana untuk mengenalkan elektronika digital kepada siswa SD, SMP, dan SMA. Pembahasan dimulai dari konteks perangkat digital di sekitar siswa, dilanjutkan Hosea Pratama Sinaga 4 jutkan dengan bilangan biner, gerbang logika, half adder, full adder, dan kerja penjumlahan empat bit. Di mana siswa akan menjumpai konsep ini

setelah kegiatan belajar selesai? Prinsip adder digunakan dalam unit aritmetika komputer, kalkulator, mikrokontroler, sistem kendali, robot, serta berbagai alat yang memproses data numerik. Penyampaian materinya diarahkan pada pengalaman yang dapat dilihat dan diuji melalui kartu biner, sakelar, LED, simulasi, serta percobaan rangkaian. Teknologi pembelajaran memberi hasil yang lebih berguna saat dipilih sesuai tujuan, dilengkapi interaksi, dan menyediakan umpan balik yang membantu siswa membaca kesalahannya. Penelitian pada siswa sekolah menengah juga memperlihatkan bahwa kegiatan pemodelan komputasional dapat menghubungkan pengetahuan sains dengan proses komputasi. Arah tersebut menempatkan adder 4 bit sebagai materi awal untuk melatih rasa ingin tahu, penalaran berurutan, serta keberanian mempelajari sistem teknologi dari komponennya yang paling dasar (National Academies of Sciences, Engineering, and Medicine, 2018; Aiken et al., 2012).

Teori Dasar

a. Elektronika Digital dan Bit Sebelum memahami adder 4 bit, pembaca perlu mengenal cara rangkaian digital membaca informasi. Arus listrik di dalam perangkat elektronik memiliki nilai tegangan yang dapat berubah secara terus-menerus. Rangkaian digital menyederhanakan perubahan tersebut menjadi dua keadaan logika, yaitu rendah dan tinggi. Keadaan rendah ditulis sebagai 0, sedangkan keadaan tinggi ditulis sebagai 1. Satu angka biner disebut bit, singkatan dari binary digit. Mengapa komputer memilih dua keadaan saja? Dua keadaan lebih mudah dibedakan oleh rangkaian, seperti lampu yang mati dan menyala atau sakelar yang terbuka dan tertutup. Komponen elektronik tidak melihat tulisan angka 0 dan 1 secara langsung. Komponen membaca rentang tegangan yang sudah ditetapkan sebagai logika rendah atau logika tinggi. Cara tersebut disebut abstraksi digital karena rincian tegangan listrik diterjemahkan menjadi simbol yang lebih sederhana. Satu bit hanya membawa dua kemungkinan, tetapi kumpulan bit dapat membentuk angka, huruf, warna, suara, gambar, dan instruksi program. Empat bit menyediakan enam belas susunan berbeda, mulai dari 0000 sampai 1111. Susunan inilah yang nantinya dipakai sebagai masukan dan keluaran pada adder 4 bit. Pemahaman tentang bit memberi dasar untuk melihat komputer sebagai kumpulan rangkaian yang menerima pola biner, mengolahnya, lalu menghasilkan pola baru sesuai fungsi yang dirancang (NIST, n.d.; Ward, 2009).

5 Hosea Pratama Sinaga

b. Sistem Bilangan Biner Manusia terbiasa memakai sistem desimal yang memiliki sepuluh simbol, yaitu 0 sampai 9. Elektronika digital memakai sistem biner yang hanya memiliki dua simbol, yaitu 0 dan 1. Nilai sebuah angka biner bergantung pada letaknya. Empat posisi dari kanan ke kiri memiliki nilai tempat 1, 2, 4, dan 8. Mengapa nilainya terus berlipat dua? Sistem biner memakai basis dua, sehingga setiap perpindahan satu tempat ke kiri menaikkan nilai menurut pangkat dua. Bilangan 1011 dapat dibaca

sebagai satu kelompok nilai tempat 8, 4, 2, dan 1. Angka 1 pada posisi paling kiri memberi nilai 8, angka 0 tidak memberi nilai 4, angka 1 berikutnya memberi nilai 2, dan angka terakhir memberi nilai 1. Hasil akhirnya adalah $8 + 0 + 2 + 1 = 11$. Empat bit tanpa tanda dapat mewakili angka desimal 0 sampai 15. Pola 0000 bernilai 0, sedangkan 1111 bernilai 15. Batas tersebut berasal dari enam belas kemungkinan pola yang dapat dibentuk oleh empat bit. Secara umum, bilangan biner tanpa tanda yang terdiri dari n bit memiliki nilai terendah 0 dan nilai tertinggi $2^n - 1$. Untuk empat bit, nilai tertingginya adalah $2^4 - 1 = 15$. Konsep ini menjelaskan mengapa hasil penjumlahan yang lebih besar dari 15 membutuhkan satu bit tambahan di bagian kiri (Stanford University, 2024; Colorado School of Mines, n.d.).

Biner Nilai tempat Desimal

0001 1 1

0010 2 2

0100 4 4

1000 8 8

1011 $8+2+1$ 11

1111 $8+4+2+1$ 15

c. LSB dan MSB Setiap posisi pada bilangan biner mempunyai pengaruh yang berbeda terhadap nilai keseluruhan. Posisi paling kanan disebut Least Significant Bit atau LSB karena memiliki nilai tempat paling kecil. Pada bilangan empat bit, LSB bernilai $2^0 = 1$. Posisi paling kiri disebut Most Significant Bit atau MSB karena menyimpan nilai tempat terbesar. MSB pada bilangan empat bit bernilai $2^3 = 8$. Sebagai contoh, perubahan LSB pada 1010 menjadi 1011 hanya menambah nilai dari 10 menjadi 11. Perubahan MSB pada 0010 menjadi 1010 menambah nilai dari 2 menjadi 10. Apa pengaruh kedua posisi ini terhadap proses penjumlahan? Perhitungan dimulai dari LSB, sama seperti penjumlahan desimal yang dimulai dari kolom satuan. Carry yang muncul pada posisi tersebut diteruskan menuju posisi di sebelah kiri. Proses bergerak sampai mencapai MSB. Pada adder 4 bit, penamaan A1 dan B1 biasanya digunakan untuk pasangan bit paling rendah, sedangkan A4 dan B4 digunakan Hosea Pratama Sinaga 6

untuk pasangan bit paling tinggi. Keluaran S1 menampilkan hasil dari posisi paling rendah dan S4 menampilkan hasil dari posisi paling tinggi. Urutan ini membantu pembaca mengikuti arah perpindahan carry dan mencegah kesalahan saat memasang sakelar atau LED pada rangkaian percobaan (Cornell University, 2016; Brown University, 2025).

d. Penjumlahan Biner Penjumlahan biner memakai aturan yang lebih sedikit daripada penjumlahan desimal. Empat kemungkinan dasarnya adalah $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$, dan $1 + 1 = 10$. Penulisan $1 + 1 = 10$ sering membingungkan pembaca yang baru mengenal biner. Angka 10 tersebut dibaca sebagai satu nol dalam basis dua, bukan sepuluh dalam basis desimal. Digit 0 ditempatkan pada kolom hasil, sedangkan digit 1 dikirim

ke kolom berikutnya sebagai carry. Cara kerjanya serupa dengan penjumlahan $7 + 5$ pada sistem desimal. Hasil 12 membuat angka 2 ditulis pada kolom satuan dan angka 1 disimpan untuk kolom puluhan. Perbedaananya terletak pada batas setiap kolom. Sistem desimal membuat carry saat hasil mencapai 10, sedangkan sistem biner membuat carry saat hasil mencapai 2. Contoh $0101 + 0011$ dapat dibaca sebagai $5 + 3$. Perhitungan dimulai dari kanan. Pasangan pertama menghasilkan 0 dengan carry 1. Carry tersebut ikut dihitung pada posisi berikutnya sampai keluaran akhirnya menjadi 1000 atau 8 dalam desimal. Dari mana komputer mengetahui kapan carry perlu dikirim? Aturan tersebut diwujudkan melalui gerbang logika yang menghubungkan masukan, keluaran sum, dan keluaran carry (Carnegie Mellon University, n.d.; MIT OpenCourseWare, 2006).

$$0101 = 5 \quad 0011 = 3 \quad 1000 = 8$$

e. Gerbang Logika Gerbang logika merupakan rangkaian kecil yang menerima satu atau beberapa masukan biner lalu menghasilkan keluaran berdasarkan aturan tertentu. Gerbang AND menghasilkan 1 ketika seluruh masukan bernilai 1. Gerbang OR menghasilkan 1 ketika sedikitnya satu masukan bernilai 1. Gerbang XOR menghasilkan 1 ketika kedua masukan memiliki nilai yang berbeda. Gerbang NOT membalik nilai masukan, sehingga 0 berubah menjadi 1 dan 1 berubah menjadi 0. Apa hubungan aturan tersebut dengan penjumlahan? Ketika dua bit dijumlahkan, keluaran utama bernilai 1 saat hanya salah satu masukan bernilai 1. Pola tersebut sesuai dengan fungsi XOR. Carry bernilai 1 saat kedua masukan sama-sama bernilai 1, sesuai dengan fungsi AND. Gerbang OR dipakai untuk menggabungkan beberapa kemungkinan terbentuknya carry pada full adder. Tabel kebenaran menjadi alat untuk mencatat seluruh kombi7 Hosea Pratama Sinaga

nasi masukan dan keluaran secara teratur. Jika sebuah gerbang memiliki dua masukan, terdapat empat kemungkinan pasangan, yaitu 00, 01, 10, dan 11. Perancang rangkaian dapat membaca tabel tersebut, menuliskan persamaan Boolean, lalu menyusun koneksi gerbang yang menghasilkan perilaku sesuai kebutuhan. Alur tersebut mengubah aturan matematika menjadi jaringan komponen elektronik yang dapat diuji menggunakan sakelar dan LED (Duke University, 2006; University of Wisconsin-Madison, 2025).

$$A \ B \ \text{AND} \ \text{OR} \ \text{XOR} \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0$$

f. Half Adder Half adder adalah rangkaian yang menjumlahkan dua bit tunggal. Kedua masukannya diberi nama A dan B. Rangkaian ini menghasilkan dua keluaran, yaitu sum dan carry. Sum menyatakan angka hasil pada posisi yang sedang dihitung, sedangkan carry menyatakan angka simpanan yang bergerak menuju posisi berikutnya. Gerbang XOR dipakai untuk menghasilkan sum karena keluarannya bernilai 1 saat A dan B berbeda. Gerbang AND dipakai untuk menghasilkan carry karena keluarannya bernilai 1 saat A dan B sama-sama bernilai 1. Hubungan tersebut dapat ditulis sebagai $S = A \oplus B$ dan $C = A \cdot B$. Lambang XOR berarti XOR, sedangkan

tanda titik menyatakan AND. Coba masukkan $A = 1$ dan $B = 1$. Gerbang XOR memberi keluaran 0, sedangkan gerbang AND memberi keluaran 1. Hasilnya dibaca sebagai 10, tepat seperti aturan penjumlahan biner. Nama half adder muncul karena rangkaian tersebut belum menerima carry dari perhitungan sebelumnya. Keterbatasan ini tidak terasa pada posisi paling kanan karena belum ada kolom sebelumnya yang mengirim carry. Masalah muncul pada posisi kedua, ketiga, dan keempat. Setiap posisi tersebut membutuhkan rangkaian yang mampu menghitung A, B, serta carry yang datang dari sisi kanan. Kebutuhan itu membawa pembahasan menuju full adder (Redekopp, 2025; Bucknell University, 2020).

A B Sum Carry 0 0 0 0 0 1 1 0 1 0 1 0 1 1 0 1 Hosea Pratama Sinaga 8

g. Full Adder Full adder menyelesaikan kekurangan pada half adder melalui satu masukan tambahan bernama carry in atau Cin. Tiga masukannya terdiri dari A, B, dan Cin, sedangkan keluarannya terdiri dari sum dan carry out. Sum dapat ditulis sebagai $S = A \oplus B \oplus \text{Cin}$. Carry out dapat ditulis sebagai $\text{Cout} = AB + \text{Cin}(A \oplus B)$. Tanda tambah pada persamaan Boolean menyatakan operasi OR. Bagian AB menghasilkan carry saat A dan B sama-sama bernilai 1. Bagian Cin ($A \oplus B$) menghasilkan carry saat carry in bernilai 1 dan hanya salah satu dari A atau B bernilai 1. Full adder juga dapat dibangun dari dua half adder dan satu gerbang OR. Half adder pertama menjumlahkan A dengan B. Half adder kedua menjumlahkan hasil pertama dengan carry in. Dua keluaran carry digabungkan oleh gerbang OR untuk membentuk carry out. Bagaimana jika ketiga masukan bernilai 1? Jumlah desimalnya adalah 3, yang ditulis sebagai 11 dalam biner. Sum bernilai 1 dan carry out juga bernilai 1. Delapan kombinasi pada tabel kebenaran mencakup seluruh keadaan yang mungkin terjadi pada satu posisi bit (University of Toronto, n.d.; Wellesley College, 2025).

A B Carry in Sum Carry out 0 0 0 0 0 0 1 1 0 0 1 0 1 0 0 1 1 0 1 1 0 0 1 0 1 0 1 0 1 1 1 0 0 1 1 1 1 1 1

h. Susunan Adder 4 Bit Adder 4 bit terbentuk dari empat tahap penjumlahan satu bit. Setiap tahap menangani satu pasangan masukan A dan B. Tahap pertama memproses A1 dan B1 sebagai LSB. Carry out dari tahap pertama masuk ke tahap kedua sebagai carry in. Pola yang sama berlangsung sampai tahap keempat yang memproses A4 dan B4 sebagai MSB. Susunan tersebut menghasilkan empat keluaran sum, yaitu S1, S2, S3, dan S4, serta satu carry out akhir. Seluruh keluarannya membentuk hasil lima bit jika carry terakhir ikut ditulis. Sebagai contoh, $A = 0110$ bernilai 6 dan $B = 0011$ bernilai 3. Rangkaian menghasilkan $S = 1001$ dan carry out = 0, sehingga hasilnya bernilai 9. Contoh lain, $A = 1111$ bernilai 15 dan $B = 0001$ bernilai 1. Empat keluaran sum menjadi 0000, sedangkan carry out bernilai 1. Hasil lengkapnya ditulis sebagai 10000 atau 16. Mengapa satu adder dapat diperbesar menjadi 8 bit, 16 bit, atau 32 bit? 9 Hosea Pratama Sinaga

Full adder dirancang sebagai blok yang dapat dihubungkan berulang

kali. Penambahan jumlah tahap memperluas banyaknya bit yang dapat diproses, sementara aturan pada setiap tahap tetap sama (Cornell University, 2025; University of San Francisco, 2026).

i. IC 74LS83 Empat full adder dapat dirangkai menggunakan banyak gerbang satu per satu, tetapi pekerjaan tersebut membutuhkan lebih banyak komponen, kabel, dan ruang pada papan percobaan. IC 74LS83 menggabungkan fungsi penjumlahan empat bit ke dalam satu kemasan. Modul praktikum memakai delapan masukan utama yang terdiri dari A1 sampai A4 dan B1 sampai B4, satu carry in, empat keluaran sum, serta satu carry out. Sakelar dapat dipasang pada bagian masukan untuk menentukan nilai 0 atau 1, sedangkan LED dipasang pada keluaran agar hasil penjumlahan terlihat secara langsung. Saat A1 sampai A4 disetel menjadi 0101 dan B1 sampai B4 disetel menjadi 0011, LED keluaran membentuk pola 1000. Carry in juga dapat diaktifkan untuk menambahkan satu nilai ke hasil akhir. Materi menyatakan bahwa carry out bernilai 1 saat hasil penjumlahan melampaui 15. Dokumen Texas Instruments untuk seri 74LS283 menjelaskan bahwa LS283 setara secara fungsi dan listrik dengan LS83A, sedangkan susunan terminalnya berbeda. Informasi tersebut perlu diperhatikan ketika siswa membaca datasheet atau mengganti tipe IC karena fungsi yang sama tidak menjamin posisi pin yang sama. Pemasangan daya, ground, masukan, dan keluaran wajib mengikuti diagram terminal komponen yang benar (Modul Praktikum Elektronika Digital, n.d.; Texas Instruments, 2001). Hosea Pratama Sinaga 10

j. Perpindahan Carry dan Waktu Tunda Keluaran rangkaian digital tidak muncul tepat pada saat masukan berubah. Setiap gerbang membutuhkan waktu singkat untuk membaca masukan dan membentuk keluaran. Waktu tersebut disebut propagation delay atau waktu tunda perambatan. Pada susunan ripple carry, carry bergerak dari tahap LSB menuju tahap berikutnya secara berurutan. Tahap kedua menunggu carry dari tahap pertama, tahap ketiga menunggu tahap kedua, dan tahap terakhir menunggu tahap ketiga. Semakin panjang jumlah bit, semakin panjang jalur yang mungkin harus dilalui carry. Bayangkan empat siswa berdiri berjajar sambil menyampaikan pesan. Siswa terakhir baru memperoleh informasi setelah pesan melewati siswa pertama, kedua, dan ketiga. Prinsip yang sama terjadi pada ripple carry adder. Perancang rangkaian dapat mempercepat proses memakai carry look-ahead, yaitu logika yang memperkirakan carry berdasarkan pasangan bit yang menghasilkan atau meneruskan carry. Cara tersebut mengurangi ketergantungan pada perpindahan carry satu tahap demi satu tahap. Konsep waktu tunda menjelaskan mengapa bentuk rangkaian turut menentukan kecepatan perangkat. Dua rangkaian dapat menghasilkan jawaban matematika yang sama, tetapi membutuhkan waktu pemrosesan yang berbeda. Pemahaman ini membuka hubungan antara materi adder sederhana dan perancangan prosesor berkecepatan tinggi (University

of California San Diego, 2018; Cornell ECE Open Courseware, n.d.).

k. Carry Out dan Overflow Carry out muncul ketika hasil penjumlahan tanpa tanda membutuhkan bit tambahan di luar kapasitas rangkaian. Pada adder 4 bit, kapasitas keluaran utama berhenti pada pola 1111 atau 15. Operasi $1111 + 0001$ menghasilkan 10000. Empat keluaran sum menyimpan 0000 dan carry out menyimpan angka 1 di bagian paling kiri. Dalam penggunaan dasar adder 4 bit, carry out dapat dibaca sebagai tanda bahwa hasil lengkap membutuhkan lima bit. Istilah overflow perlu digunakan secara hati-hati karena artinya dapat berubah saat pola biner dipakai untuk bilangan bertanda. Carry berkaitan langsung dengan penjumlahan tanpa tanda, sedangkan signed overflow terjadi ketika hasil bilangan bertanda keluar dari rentang yang dapat diwakili. Sebagai contoh, carry dapat muncul tanpa signed overflow, dan signed overflow dapat terjadi tanpa carry. Pembahasan untuk siswa pemula dapat difokuskan pada bilangan tanpa tanda agar hubungan antara nilai 0 sampai 15 dan carry out tetap jelas. Penjelasan mengenai bilangan bertanda dapat ditempatkan pada tingkat lanjutan. Di dalam prosesor, informasi carry dan overflow disimpan sebagai penanda kondisi yang dapat dibaca oleh instruksi berikutnya. Penanda tersebut membantu komputer membandingkan nilai, mengambil keputusan, dan mengolah angka dengan ukuran lebih besar dari satu kelompok bit (Arm, 2013; Swarthmore College, n.d.). 11 Hosea Pratama Sinaga

l. Contoh Aliran Penjumlahan Empat Bit Cara termudah memahami adder 4 bit adalah mengikuti carry dari kanan ke kiri. Ambil $A = 1001$ atau 9, $B = 0110$ atau 6, dan carry in awal = 1. Posisi pertama menjumlahkan $A_1 = 1$, $B_1 = 0$, dan carry in = 1. Jumlahnya 2 atau 10 dalam biner, sehingga $S_1 = 0$ dan carry menuju posisi kedua bernilai 1. Posisi kedua menerima $A_2 = 0$, $B_2 = 1$, dan carry = 1. Hasilnya kembali 10, sehingga $S_2 = 0$ dan carry berikutnya tetap 1. Posisi ketiga memproses $A_3 = 0$, $B_3 = 1$, serta carry = 1. S_3 bernilai 0 dan carry tetap bergerak. Posisi keempat menerima $A_4 = 1$, $B_4 = 0$, dan carry = 1. Keluaran S_4 bernilai 0 dengan carry out akhir bernilai 1. Pola hasil lengkapnya adalah 10000 atau 16. Perhitungan desimalnya memberi jawaban yang sama, yaitu $9 + 6 + 1 = 16$. Apa yang sebenarnya dilakukan oleh empat tahap tersebut? Setiap tahap hanya menyelesaikan persoalan kecil yang terdiri dari tiga masukan. Hubungan antartahap mengubah empat pekerjaan sederhana menjadi satu penjumlahan yang lebih besar. Prinsip pemecahan seperti ini digunakan luas dalam perancangan sistem digital (University of Pittsburgh, n.d.; Illinois Institute of Technology, n.d.).

Adder 4 bit memberi gambaran langsung mengenai cara matematika berubah menjadi kerja rangkaian. Bit menyimpan dua keadaan, sistem biner memberi nilai pada susunan bit, gerbang logika menjalankan aturan, half adder menangani dua masukan, dan full adder memasukkan carry dari posisi sebelumnya. Empat full adder yang bekerja bersama mampu menjum-

lahkan dua angka empat bit dalam satu alur. Dari rangkaian kecil ini, siswa bisa mulai membaca dasar kerja kalkulator, mikrokontroler, unit aritmetika, dan prosesor komputer.

Teori Tambahan

a. Adder 4 Bit sebagai Gambaran Kecil Cara Kerja Komputer Adder 4 bit tidak berdiri sebagai rangkaian penjumlah yang terpisah dari teknologi sehari-hari. Rangkaian tersebut mewakili salah satu pekerjaan paHosea Pratama Sinaga 12

ling mendasar di dalam sistem digital, yaitu menerima data, mengolahnya berdasarkan aturan logika, lalu mengirim hasil ke bagian lain. Saat seseorang mengetik $6 + 3$ pada kalkulator, angka yang terlihat di layar lebih dahulu diterjemahkan menjadi pola biner. Pola pertama masuk melalui kelompok masukan A, pola kedua masuk melalui kelompok B, lalu carry bergerak dari bit paling rendah menuju bit paling tinggi. Hasil akhirnya dapat disimpan, ditampilkan, dibandingkan, atau dipakai untuk perhitungan berikutnya. Di mana proses seperti ini dipakai selain kalkulator? Penjumlahan biner muncul pada unit aritmetika dan logika, pencacah digital, pengaturan alamat memori, pengukuran waktu, pengolahan data sensor, serta sistem kendali. Prosesor modern memiliki susunan yang jauh lebih besar, walaupun gagasan dasarnya masih dapat ditelusuri melalui hubungan antara bit, gerbang logika, carry, dan keluaran sum. Adder 4 bit memberi ukuran yang cukup kecil untuk diamati siswa tanpa memutus hubungannya dengan komputer yang sesungguhnya (Harris & Harris, 2012; Mano & Ciletti, 2017).

b. Dari Konsep Abstrak Menuju Bentuk yang Dapat Dilihat Bilangan biner mudah dibaca setelah seseorang memahami nilai tempatnya, tetapi bentuk 0 dan 1 tetap terasa abstrak bagi anak yang baru mengenal elektronika. Tampilan berupa deretan angka belum menjelaskan bahwa setiap bit sebenarnya mewakili keadaan listrik. Sakelar dan LED dapat mengubah konsep tersebut menjadi pengalaman yang dapat diamati. Sakelar yang dilepas dibaca sebagai 0, sedangkan sakelar yang diaktifkan dibaca sebagai 1. LED keluaran memperlihatkan hasil tanpa menunggu penjelasan panjang. Apa yang terjadi saat dua sakelar bernilai 1 ditekan bersamaan? Siswa dapat menyaksikan LED sum padam dan LED carry menyala. Gerbang XOR dan AND yang semula berupa simbol di kertas mulai memiliki makna karena efeknya tampak pada rangkaian. Penyajian seperti ini membantu pikiran membagi informasi menjadi bagian yang lebih kecil. Teks menjelaskan aturan, gambar memperlihatkan hubungan komponen, sedangkan percobaan memberi bukti yang langsung terlihat. Terlalu banyak istilah dalam satu waktu dapat memenuhi kapasitas kerja ingatan dan membuat pembaca kehilangan alur. Materi yang disusun dari lampu, sakelar, bit, gerbang, lalu adder menjaga beban belajar tetap terarah (Mayer, 2009; Sweller, 1988).

c. Praktikum sebagai Pengalaman Belajar Elektronika sulit dipahami

jika pembelajaran berhenti pada kegiatan membaca tabel kebenaran. Praktikum memberi kesempatan kepada pelajar untuk mengubah masukan, melihat keluaran, mencatat pola, lalu menghubungkan hasilnya dengan teori. Siswa dapat memasukkan $A = 0011$ dan $B = 0101$, memperkirakan hasilnya, menyalakan rangkaian, lalu membandingkan pola LED dengan jawaban yang sudah dihitung. Jika keluaran berbeda, mereka perlu 13 Hosea Pratama Sinaga

memeriksa posisi bit, carry in, sambungan kabel, catu daya, atau susunan pin IC. Kesalahan berubah menjadi bagian dari pencarian, bukan sekadar nilai yang salah. Bagaimana sebuah konsep bertahan lebih lama dalam ingatan? Pengalaman langsung memberi jalur yang lebih kuat karena peserta didik melakukan tindakan, menerima hasil, memikirkan penyebabnya, lalu mencoba susunan baru. Proses tersebut dekat dengan pembelajaran eksperiensial yang bergerak dari pengalaman konkret menuju pengamatan dan pembentukan konsep. Papert juga melihat belajar teknologi sebagai kegiatan membuat sesuatu yang dapat diperiksa dan diperbaiki. Rangkaian adder cocok untuk pola tersebut karena setiap perubahan masukan menghasilkan respons yang segera terlihat (Kolb, 1984; Papert, 1980).

d. Tingkat Pembahasan untuk SD, SMP, dan SMA Satu rangkaian dapat diajarkan melalui kedalaman yang berbeda. Anak SD cukup mengenal dua keadaan melalui permainan lampu mati dan menyala. Carry dapat disebut angka titipan agar mereka memahami bahwa hasil yang tidak muat pada satu tempat perlu dipindahkan. Pelajar SMP mulai mempelajari nilai tempat biner, perubahan bilangan desimal ke biner, serta penjumlahan beberapa bit. Pelajar SMA dapat mengikuti susunan gerbang, persamaan Boolean, tabel kebenaran, propagation delay, dan kerja IC 74LS83. Kapan istilah teknis perlu diperkenalkan? Istilah baru sebaiknya muncul setelah anak memahami gejala yang dijelaskan oleh istilah tersebut. Seorang siswa lebih mudah menangkap makna carry out setelah melihat LED tambahan menyala pada operasi $1111 + 0001$. Penyusunan bertingkat tidak mengurangi isi ilmu. Cara ini mengatur urutan agar pembaca memiliki pijakan sebelum memasuki konsep yang lebih rumit. Bruner menyebut gagasan pembelajaran berulang sebagai kurikulum spiral, sedangkan Vygotsky menempatkan bantuan guru, teman, dan media sebagai penyangga bagi kemampuan yang sedang tumbuh. Adder 4 bit dapat berfungsi sebagai materi yang terus dikembangkan sejak pengenalan angka biner sampai analisis rangkaian digital (Bruner, 1960; Vygotsky, 1978).

e. Adder sebagai Latihan Berpikir Komputasional Pembelajaran adder tidak hanya berisi cara menjumlahkan angka biner. Setiap tahap mengajarkan cara memecah persoalan besar menjadi pekerjaan yang lebih kecil. Penjumlahan empat bit dipisahkan menjadi empat penjumlahan satu bit. Setiap full adder hanya perlu membaca A, B, dan carry in. Hasil dari satu tahap diteruskan sebagai masukan tahap berikutnya. Bagaimana komputer

menyelesaikan pekerjaan besar tanpa memahami persoalan seperti manusia? Sistem digital menjalankan rangkaian instruksi kecil dalam urutan yang jelas. Siswa yang mengikuti perjalanan carry sebenarnya sedang mempelajari dekomposisi, pengenalan pola, penyusunan algoritma, dan pemeriksaan Hosea Pratama Sinaga 14

hasil. Mereka dapat menuliskan urutan kerja mulai dari membaca LSB, menentukan sum, menentukan carry, lalu bergerak ke posisi berikutnya. Saat hasil rangkaian keliru, mereka mencari tahap yang menyebabkan kesalahan. Kegiatan tersebut dekat dengan debugging dalam pemrograman. Computational thinking tidak terbatas pada menulis kode. Cara berpikir ini juga muncul saat seseorang menyusun rangkaian, menelusuri sinyal, membandingkan keluaran dengan prediksi, dan memperbaiki prosedur yang gagal (Wing, 2006; Brennan & Resnick, 2012).

f. Pembelajaran Aktif melalui Tantangan dan Permainan Kegiatan adder dapat dibentuk sebagai tantangan singkat agar kelas tidak berubah menjadi sesi menghafal simbol. Guru dapat menampilkan dua bilangan desimal, lalu meminta kelompok siswa mengubahnya menjadi biner sebelum sakelar diatur. Kelompok lain menebak pola LED hasil dan carry out. Permainan manusia sebagai full adder juga dapat dilakukan. Tiga anak memegang kartu A, B, dan carry in, sedangkan dua anak lain menentukan sum dan carry out berdasarkan tabel kebenaran. Setelah satu tahap selesai, kartu carry diberikan kepada kelompok berikutnya. Siapa yang paling cepat menemukan kesalahan ketika hasil akhirnya tidak cocok? Pertanyaan semacam itu mendorong diskusi tanpa memaksa siswa menghafal jawaban. Pembelajaran aktif membuat peserta didik berbicara, menghitung, menguji, dan menjelaskan alasan di balik pilihannya. Penelitian pada pendidikan sains dan teknik menemukan bahwa keterlibatan langsung memberi hasil belajar yang lebih baik dibandingkan ceramah yang mendominasi seluruh waktu kelas. Adder 4 bit mendukung pendekatan tersebut karena materi dapat diubah menjadi prediksi, percobaan, perlombaan, atau pemecahan masalah kelompok (Freeman et al., 2014; Prince, 2004).

g. Simulasi Digital sebagai Pendamping Rangkaian Fisik Tidak setiap sekolah memiliki IC, breadboard, catu daya, dan alat ukur dalam jumlah cukup. Simulasi digital dapat membuka akses tanpa menghilangkan inti pembelajaran. Siswa dapat menempatkan gerbang XOR, AND, dan OR pada layar, menghubungkannya dengan kabel virtual, lalu mengubah masukan 0 dan 1. Kesalahan sambungan dapat diperbaiki tanpa risiko komponen rusak. Simulasi juga memperlihatkan bagian yang sulit diamati pada rangkaian fisik, seperti perubahan sinyal pada setiap tahap carry. Apa yang membedakan simulasi dari animasi biasa? Simulasi memberi ruang bagi pengguna untuk mengubah kondisi dan menerima keluaran sesuai aturan model. Pelajar tidak hanya menonton proses yang sudah jadi. Mereka dapat menguji $0010 + 0101$, mengaktifkan carry in, mengganti satu bit, lalu

melihat bagian mana yang berubah. Rangkaian fisik tetap berguna karena memperkenalkan kabel, pin, tegangan, dan kesalahan pemasangan nyata. Simulasi berperan sebagai ruang 15 Hosea Pratama Sinaga

latihan sebelum siswa berpindah ke perangkat keras. Kajian tentang *discovery learning* dan simulasi sains mencatat manfaat yang lebih kuat ketika eksplorasi disertai tujuan, pertanyaan penuntun, dan kesempatan menjelaskan hasil (de Jong & van Joolingen, 1998; Rutten et al., 2012).

h. Umpan Balik dan Pemeriksaan Kesalahan LED keluaran pada adder memberi umpan balik hampir seketika. Saat masukan berubah, siswa dapat langsung melihat apakah prediksinya sesuai. Respons cepat membantu mereka menghubungkan tindakan dengan akibat. Umpan balik yang berguna tidak berhenti pada pemberian jawaban benar atau salah. Guru dapat bertanya pada bit mana kesalahan pertama muncul, carry mana yang tidak diteruskan, atau apakah urutan A1 sampai A4 sudah dibaca dengan benar. Pertanyaan tersebut mengarahkan perhatian pada proses berpikir. Bagaimana cara mengetahui bahwa siswa benar-benar memahami rangkaian? Mereka perlu mampu menjelaskan alasan keluaran, memperkirakan hasil sebelum percobaan, dan menemukan letak kesalahan pada contoh yang sengaja dibuat keliru. Penilaian dapat berbentuk kartu prediksi, tabel pengamatan, penjelasan lisan, atau tugas memperbaiki rangkaian. Guru memperoleh gambaran tentang bagian yang sudah dipahami tanpa menunggu ujian akhir. Umpan balik yang jelas membantu pelajar melihat jarak antara pemahaman saat ini dan tujuan pembelajaran, sedangkan penilaian formatif memberi kesempatan memperbaiki konsep selama kegiatan masih berlangsung (Hattie & Timperley, 2007; Black & Wiliam, 1998).

i. Hubungan dengan Pembelajaran Sains, Teknologi, Rekayasa, dan Matematika Adder 4 bit berada pada pertemuan beberapa bidang sekaligus. Matematika muncul melalui sistem bilangan dan operasi penjumlahan. Fisika hadir saat siswa mempelajari tegangan, arus, kondisi logika, dan waktu tunda sinyal. Rekayasa terlihat pada kegiatan menyusun rangkaian, memilih komponen, membaca diagram pin, serta menguji keluaran. Ilmu komputer masuk melalui representasi data, algoritma, dan hubungan antara perangkat keras dengan program. Mengapa materi kecil seperti adder cocok dijadikan kegiatan lintas bidang? Satu percobaan dapat dimulai dari hitungan sederhana, diterjemahkan menjadi biner, diwujudkan dalam sakelar, diproses oleh IC, lalu dibaca kembali sebagai angka. Setiap langkah menghubungkan konsep yang biasanya dipelajari secara terpisah. Siswa juga dapat diberi proyek merancang kalkulator mini, pencacah nilai, atau permainan tebak angka biner. Standar ilmu komputer sekolah menempatkan sistem komputasi, data, algoritma, dan dampak teknologi sebagai bagian yang saling berhubungan. Kerangka pendidikan sains modern juga mendorong peserta didik untuk membangun model, melakukan penyelidikan, menganalisis data, dan merancang solusi Hosea Pratama Sinaga 16

(Computer Science Teachers Association, 2017; NGSS Lead States, 2013).

j. Menumbuhkan Ketertarikan terhadap Elektronika Digital Ketertarikan sering muncul saat pelajar merasa mampu membuat sesuatu yang bekerja. Sebuah rangkaian sederhana yang berhasil menyalakan LED sesuai hasil perhitungan dapat memberi kepuasan yang sulit diperoleh dari membaca rumus saja. Anak mulai melihat bahwa komputer bukan kotak misterius yang hanya dapat digunakan. Perangkat tersebut tersusun dari bagian-bagian yang dapat dibongkar secara konsep, dipelajari, lalu dirancang kembali. Apa yang dapat dibuat setelah memahami adder? Siswa dapat melanjutkan ke subtractor, comparator, counter, register, multiplexer, dan unit aritmetika sederhana. Jalur belajar tersebut memberi tujuan yang jelas karena setiap materi baru tumbuh dari konsep sebelumnya. Aktivitas kreatif juga memberi ruang untuk memilih bentuk proyek, warna LED, pola masukan, atau aturan permainan. Lingkungan belajar yang memberi kesempatan membuat, mencoba, berbagi, dan memperbaiki karya dapat mempertahankan rasa ingin tahu lebih lama. Pengalaman pengembangan Scratch memperlihatkan bahwa anak belajar konsep komputasi secara lebih alami ketika mereka menghasilkan proyek yang bermakna bagi dirinya. Gagasan serupa dapat diterapkan pada elektronika melalui rangkaian yang dapat dimainkan dan dikembangkan (Resnick et al., 2009; Kafai & Burke, 2014).

Penutup Rangkaian adder 4 bit membuka isi komputer dari bagian yang paling mudah diamati. Angka biner masuk melalui sakelar, gerbang logika mengolah setiap bit, carry berpindah dari satu tahap ke tahap berikutnya, lalu LED menampilkan hasil penjumlahan. Jalur tersebut mempertemukan matematika, fisika, ilmu komputer, dan rekayasa dalam satu percobaan. Siswa tidak hanya mengetahui bahwa komputer memakai angka 0 dan 1, mereka dapat mengikuti perjalanan kedua angka itu sampai berubah menjadi keluaran. Apa yang membuat percobaan sederhana ini layak dibawa ke ruang kelas? Setiap perubahan masukan menimbulkan respons yang bisa diperiksa langsung. Siswa belajar memperkirakan hasil, menemukan kesalahan sambungan, membaca pola, dan menjelaskan alasan sebuah LED menyala. Kegiatan lintas bidang yang disusun dalam persoalan nyata dapat membantu pelajar membangun hubungan antarkonsep, sedangkan pembelajaran yang memberi ruang bagi pemecahan masalah dan kemandirian menyiapkan mereka menghadapi persoalan teknologi yang terus berkembang (National Research Council, 2014; OECD, 2020). Penyampaian materi perlu mengikuti usia dan pengalaman siswa. Anak SD dapat memulai dari lampu mati, lampu menyala, serta angka titipan. Siswa SMP bergerak menuju nilai tempat biner dan penjumlahan beberapa bit. 17 Hosea Pratama Sinaga

Siswa SMA mempelajari tabel kebenaran, full adder, carry in, carry out, waktu tunda, serta penggunaan IC 74LS83. Media digital dapat dipakai saat perangkat fisik belum tersedia, sedangkan rangkaian nyata memberi

pengalaman memasang komponen dan menelusuri kesalahan secara langsung. Kapan teknologi benar-benar membantu pembelajaran? Manfaatnya muncul saat alat dipilih untuk memperjelas konsep, memberi kesempatan bereksperimen, dan tetap berada di bawah arahan tujuan belajar. Kerangka kompetensi digital juga menempatkan pemecahan masalah serta kemampuan menggunakan teknologi secara kritis sebagai bagian dari kecakapan yang perlu dikembangkan, bukan sekadar keterampilan mengoperasikan perangkat (UNESCO, 2023; European Commission, 2025). Pembelajaran adder dapat menjadi langkah pertama sebelum siswa mengenal subtractor, counter, register, multiplexer, mikrokontroler, dan unit aritmetika pada prosesor. Rasa penasaran yang muncul dari satu LED dapat berkembang menjadi keinginan merancang alat sendiri. Sekolah tidak perlu memulai dari mesin yang rumit. Empat bit, beberapa sakelar, dan sebuah rangkaian penjumlah sudah cukup untuk memperlihatkan bahwa teknologi dibangun melalui aturan yang teratur serta dapat dipelajari sedikit demi sedikit. Pengalaman STEM yang memberi ruang untuk mencoba, membuat, dan memperbaiki rancangan dapat memperluas keterlibatan siswa, sementara pelajar juga membutuhkan teknologi yang sesuai dengan kebutuhan, mudah diakses, dan memberi mereka peran aktif dalam proses belajar (National Academies of Sciences, Engineering, and Medicine, 2024; UNESCO, 2024).

Kesimpulan Adder 4 bit adalah rangkaian elektronika digital yang menjumlahkan dua bilangan biner empat bit bersama carry in, lalu menghasilkan empat bit sum dan satu carry out. Rangkaian ini tersusun dari empat full adder yang bekerja berurutan dari LSB menuju MSB. Carry dari setiap tahap diteruskan ke tahap berikutnya sampai hasil akhir terbentuk. Konsep tersebut dapat dikenalkan kepada siswa SD melalui lampu dan sakelar, kepada siswa SMP melalui penjumlahan biner, serta kepada siswa SMA melalui gerbang logika, tabel kebenaran, dan IC 74LS83. Percobaan, permainan, dan simulasi membuat proses yang berlangsung di dalam komputer dapat dilihat serta diuji. Adder 4 bit akhirnya tidak berhenti sebagai rangkaian penjumlah. Materi ini menjadi jalan awal bagi pelajar untuk memahami perangkat digital, melatih penalaran berurutan, dan mengenali bahwa teknologi dapat dipelajari dari susunan kecil yang bekerja bersama. Hosea Pratama Sinaga 18

Daftar Pustaka

- Aiken, J. M., Caballero, M. D., Douglas, S. S., Burk, J. B., Scanlon, E. M., Thoms, B. D., & Schatz, M. F. (2012). Understanding student computational thinking with computational modeling. arXiv. <https://doi.org/10.48550/arXiv.1207.1764>
- Arm Limited. (2013, September 11). Condition codes 1: Condition flags and codes. Arm Community. <https://developer.arm.com/community/arm-community->

- blogs/b/architectures-and-processors-blog/posts/condition-codes-1-condition-flags-and-codes
- Black, P., & Wiliam, D. (1998). Assessment and classroom learning. *Assessment in Education: Principles, Policy & Practice*, 5(1), 7-74. <https://doi.org/10.1080/0969595980050102>
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Paper presented at the Annual Meeting of the American Educational Research Association, Vancouver, Canada. https://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf
- Bruner, J. S. (1960). *The process of education*. Harvard University Press. <https://doi.org/10.4159/9780674028999>
- Computer Science Teachers Association. (2017). *CSTA K-12 computer science standards: Revised 2017*. <https://csteachers.org/k12standards/>
- de Jong, T., & van Joolingen, W. R. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of Educational Research*, 68(2), 179-202. <https://doi.org/10.3102/00346543068002179>
- Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23), 8410-8415. <https://doi.org/10.1073/pnas.1319030111>
- Global Education Monitoring Report Team. (2023). *Global education monitoring report 2023: Technology in education: A tool on whose terms?* UNESCO. <https://unesdoc.unesco.org/ark:/48223/pf0000385723>
- Global Education Monitoring Report Team, & Restless Development. (2024). *Technology in education: A tool on our terms! 2024 youth report*. UNESCO. <https://unesdoc.unesco.org/ark:/48223/pf0000391424>
- Harris, D. M., & Harris, S. L. (2012). *Digital design and computer architecture* (2nd ed.). Morgan Kaufmann.
- Hattie, J., & Timperley, H. (2007). The power of feedback. *Review of Educational Research*, 77(1), 81-112. <https://doi.org/10.3102/003465430298487>
- Kafai, Y. B., & Burke, Q. (2014). *Connected code: Why children need to learn programming*. MIT Press.
- Kolb, D. A. (1984). *Experiential learning: Experience as the source of learning and development*. Prentice Hall.
- Hosea Pratama Sinaga
- Mano, M. M., & Ciletti, M. D. (2018). *Digital design: With an introduction to the Verilog HDL, VHDL, and SystemVerilog* (6th ed.). Pearson.
- Mayer, R. E. (2009). *Multimedia learning* (2nd ed.). Cambridge University Press.
- National Academy of Engineering, & National Research Council. (2014). *STEM integration in K-12 education: Status, prospects, and an agenda for research*. National Academies Press. <https://doi.org/10.17226/18612>
- National Academies of Sciences, Engineering, and Medicine. (2018). *How people learn II: Learn-*

- ers, contexts, and cultures. National Academies Press. <https://doi.org/10.17226/24783> National Academies of Sciences, Engineering, and Medicine. (2025). Scaling and sustaining Pre-K-12 STEM education innovations: Systemic challenges, systemic responses. National Academies Press. <https://doi.org/10.17226/27950> National Institute of Standards and Technology. (n.d.). Bit. Computer Security Resource Center Glossary. Retrieved July 19, 2026, from <https://csrc.nist.gov/glossary/term/bit> NGSS Lead States. (2013). Next generation science standards: For states, by states. National Academies Press. <https://doi.org/10.17226/18290>
- Nisan, N., & Schocken, S. (2005). The elements of computing systems: Building a modern computer from first principles. MIT Press.
- OECD. (2020). Back to the future of education: Four OECD scenarios for schooling. OECD Publishing. <https://doi.org/10.1787/178ef527-en>
- Papert, S. (1980). Mindstorms: Children, computers, and powerful ideas. Basic Books.
- Percobaan 9: Full adder 4 bit with carry. (n.d.). [Modul praktikum elektronika digital].
- Prince, M. (2004). Does active learning work? A review of the research. *Journal of Engineering Education*, 93(3), 223-231. <https://doi.org/10.1002/j.2168-9830.2004.tb00809.x>
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. B. (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11), 60-67. <https://doi.org/10.1145/1592761.1592779>
- Rutten, N., van Joolingen, W. R., & van der Veen, J. T. (2012). The learning effects of computer simulations in science education. *Computers & Education*, 58(1), 136-153. <https://doi.org/10.1016/j.compedu.2011.07.017>
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2), 257-285. https://doi.org/10.1207/s15516709cog1202_4
- Texas Instruments. (2001). 4-bit binary full adders with fast carry (Rev. A) [Data sheet]. <https://www.ti.com/lit/ds/symlink/sn74ls283.pdf>
- Vuorikari, R., Kluzer, S., & Punie, Y. (2022). DigComp 2.2: The digital competence framework for citizens, with new examples of knowledge, skills and attitudes. Publications Office of the European Union. <https://doi.org/10.2760/115376>
- Vygotsky, L. S. (1978). Mind in society: The development of higher psychological processes (M. Cole, V. John-Steiner, S. Scribner, & E. Souberman, Eds.). Harvard University Press.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>