

```

1  atomic<uint64_t> lock(0);
2
3  struct Descriptor {
4      jmp_buf *checkpoint;
5      uint64_t start_time;
6      unordered_map<uintptr_t*, uintptr_t> writes;
7      vector<pair<uintptr_t*, uintptr_t>> reads;
8  };
9  void beginTx(jmp_buf *b) {
10     checkpoint = b;
11     start_time = lock;
12     if (start_time & 1)
13         start_time--;
14 }
15 void abortTx() {
16     writes.clear();
17     reads.clear();
18     longjmp(*checkpoint, 1);
19 }
20 int void write(uintptr_t *ptr, uintptr_t val) {
21     writes.insert_or_assign(addr, val);
22 }

```

FIGURE 20.13 Software transactions with value-based validation (1/2).