Refutation of Halting Problem Diagonalization Argument

I spent two years creating the x86utm operating system to concretely address the halting problem using a halt decider written in C. This partial halt decider invokes an x86 emulator to execute the input in debug step mode. The input is the machine address of the x86 function.

It examines the complete execution trace of this input immediately after each x86 instruction is simulated. As soon as the partial halt decider recognizes a non-terminating behavior pattern it aborts the simulation and reports not-halting.

```
typedef void (*machine_address)(void);


int D(machine_address P)
{
  if ( H(P, P) )
    return 0   // reject when H accepts
  return 1;    // accept when H rejects
}
```

When H is a simulating halt decider H(D,D) rejects its input as a halting computation on the basis that H(D,D) specifies infinitely nested simulation to H unless H aborts its simulation of D(D).

Table T (All Turing machines on each other as input)

	<M1>	<M2>	<M3>...	<D>...
M1	accept			reject
M2		reject		accept
M3			~halt	accept
...				
D	reject	accept	accept	accept
...				

Table TH is defined on the basis of Table T where:
(a) accept becomes accept (b) reject becomes reject (c) ~Halt becomes reject

Table TH (Turing machine H on all Turing Machine pairs as input)

	<M1>	<M2>	<M3>...	<D>...
M1	accept			reject
M2		reject		accept
M3			reject	accept
...				
D	reject	accept	accept	reject
...				

On the diagonal: a TM is executed with its own TM description as input. Table TD only has a single input that reverses the value of the diagonal of table TH for each TM description on the horizontal axis of table TH.

Table TD (reverses H decision along the diagonal of table TH)

<M1>	<M2>	<M3>...	<D>...
reject	accept	accept	accept

All of the table values are correct.
All of the values in TD must be the opposite of the values of the TH diagonal is satisfied:
The reject value of table TH at (D, <D>) corresponds to the actual behavior of H(D,<D>).
The accept value of table T at (D, <D>) corresponds to the actual behavior of D(<D>) also shown at element <D> in table TD.

Because the requirement that table TH have the same (accept / reject) value as table T directly contradicts the actual behavior of H(D,<D>) and D(<D>) we can toss out this requirement as erroneous.