Approach

We allocate a fresh block for every function call ...

Problem

When ordering the block before the call, we do not yet know the space consumption of the called function.

We order the new block after entering the function body!

Organisational cells as well as actual parameters must be allocated inside the old block …
When entering the new function, we now allocate the new block ...

In particular, the local variables reside in the new block ...

We address ...

- the formal parameters relatively to the frame-pointer;
- the local variables relatively to the stack-pointer.

We must re-organize the complete code generation ...

Alternative: Passing of parameters in registers ...

The values of the actual parameters are determined before allocation of the new stack frame.
The complete frame is allocated inside the new block – plus the space for the current parameters.

Inside the new block, though, we first store the old SP (possibly +1) in order to correctly return the result ...

3. Idea: Hybrid Solution

- For the first $k$ threads, we allocate a separate stack area.
- For all further threads, we successively use one of the existing ones !!!

- For few threads extremely simple and efficient;
- For many threads amortized storage usage.